A Location-Inventory Problem for Customers with Time Constraints

A LOCATION-INVENTORY PROBLEM FOR CUSTOMERS WITH TIME CONSTRAINTS

BY

FAN E, B.Sc.

A THESIS

SUBMITTED TO THE SCHOOL OF COMPUTATIONAL SCIENCE AND ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

Master of Applied Science (2016)	McMaster University
(Computational Science and Engineering)	Hamilton, Ontario, Canada

TITLE:	A Location-Inventory Problem for Customers with Time Constraints
AUTHOR:	Fan E B.Sc., (Applied Mathematics) Sun Yat-sen University, Guangzhou, China
SUPERVISOR:	Dr. Kai Huang
NUMBER OF PAGES:	ix, 49

Abstract

In this paper, a two-stage stochastic facility location problem integrated with inventory and recourse decisions is studied and solved. This problem is inspired by an industrial supply chain design problem of a large retail chain with slow-moving products. Uncertainty is expressed by a discrete and finite set of scenarios. Recourse actions can be taken after the realization of random demands. Location, inventory, transportation, and recourse decisions are integrated into a mixed-integer program with an objective minimizing the expected total cost. A dual heuristic procedure is studied and embedded into the sample average approximation (SAA) method. The computation experiments demonstrate that our combined SAA with dual heuristic algorithm has a similar performance on solution quality and a much shorter computational time.

To my supervisor and friends

Acknowledgements

I would first like to express my sincere gratitude to my supervisor Dr. Kai Huang of the DeGroote School of Business at McMaster University. We met almost every week to talk about my research and study. Dr. Huang not only consistently offered brilliant insights into my research or writing, but also encouraged me to think and solve research problems independently. He also provided me priceless advice on graduate life in general, which helped me enjoy a fulfilling and wonderful time on the path of learning and discovery. With his encouragement, I accepted an offer to persure a very interesting Ph.D. program.

I would also like to thank the rest of my thesis committee: Dr. B. Protas, Dr. C. Swartz, and Dr. R. Zheng. Without their time and participation, this defence would not be possible. I want to especially thank Dr. B. Protas for organizing all the events, seminars, and symposiums for the School of Computational Science and Engineering. They are an important part of all the memorable times during my graduate study.

Thanks also go to my fellow graduate students and my friends, especially Xiaozhou Zhang, A. S. M. Sohidull Islam, Micheal Xu and Yuanhua Li. Lastly, I want to thank my lovely roommate and closest friend, Kelsey Lee Reid, for her support even at my lowest times.

Contents

Abstract **iii**

Acknowledgements **v**

1 Introduction **1**

2 Literature review **7**

2.1 Risk-pooling effects 7

2.2 Location-inventory problem 9

2.3 Stochastic discrete facility location problem 11

3 The model **16**

3.1 Notations 16

3.2 Formulation 18

4 Sample Average Approximation algorithm **19**

5 A primal-dual algorithm **24**

5.1 Dual procedures 24

5.1.1 The v-ascent procedure 26

5.1.2 The u-ascent procedure 27
5.1.3 The u-v ascent procedure 29
5.2 Construction of primal feasible solution 30

6 Computational results **36**
6.1 SAA algorithm 37
6.2 SAA algorithm with dual heuristic and its improvement 38

7 Conclusion **42**

A Condensed dual problem **43**

List of Tables

1.1 Lead times and costs of different sources 4
3.1 Notations 17
6.1 Parameter settings 37
6.2 Variance of SAA statistical optimal solutions with varying M and N 38
6.3 SAA and SAA with dual heuristic results 39
6.4 SAA and SAA with improved dual heuristic results 39
6.5 SAA with improved dual heuristic results 40
6.6 SAA with improved dual heuristic results (2) 40
6.7 SAA with varying N' 41

List of Figures

1.1 Distribution System Structure 6

Chapter 1

Introduction

Facility location problems (FLP), which have been extensively researched since the first formulation proposed by Weber *et al.* (1929), have a wide application in supply chain network design (SCND). Melo *et al.* (2009) provide a detailed review and discusses the compatibility of discrete facility location problems and SCND problems. The uncapacitated facility location problem (UFLP) is one of the most basic and fundamental discrete FLP models, which acts an important role in SCND models (Revelle *et al.* (2008)). In UFLP model, there is a set of potential facility locations and a set of demand locations. Facilities are chosen to open to minimize overall cost while satisfying all known demands. Transportations are then made to the demand location with demand from its closest open facility. It is assumed that each facility has unlimited capacity, and thus the size of an open facility is automatically the summation of all demands served.

Stochastic facility location problem (SFLP) is an important extension of UFLP for several reasons. Location decisions are usually involved with allocations of large amount of investment, and as such they are difficult to revert and often remain effective for a long period of time. This makes it more difficult to estimate model parameters, like the amount of demand, market price, or transportation cost. It also increases uncertainty of environment, for example demand pattern might fluctuate during a long period of time. All of these make deterministic models less reliable for decision makers upon which to rely. Snyder (2006) provides a thorough review about recent developments and applications of SFLP.

Louveaux and Peeters (1992) introduce a very interesting extension of UFLP, which takes uncertainty into account, and models capacity decisions as first stage decisions similar to location decisions. They use a scenario approach to represent uncertain demand and cost parameters, and formulate a two-stage mixed-integer stochastic program with recourse. Location and capacity decisions are established at the first stage considering all possible scenarios, and then the most profitable distribution decisions are made. The dual ascent heuristic algorithm, which is developed by Erlenkotter (1978) for a deterministic UFLP model, is extended for their stochastic UFLP problem. Their contributions are not only the stochastic extension of UFLP model and its solution procedure, but also their unique way of regarding capacity decisions strategic as similar to location decisions.

More and more literature on incorporating inventory decisions in location decisions, known as location-inventory problems, are emerging recently. In typical location problems, tactical and operational decisions, like inventory and transportation decisions, are often neglected while making location decisions. This assumption is made for the benefit of modelling simplification, and often leads to sub-optimal solutions of realistic problems. On the other hand, most research on inventory management are built on predetermined facility locations. They focus on deciding the optimal replenishment policy, like re-order-point and order quantity, and safety stock level under uncertainty without modifying the existing facility location network. Integrated location-inventory models may lead to better location and inventory decisions, especially when holding cost and demand variability are high. Farahani *et al.* (2015) describe a general location-inventory model and summarize recent developments in this area.

Inspired by an industrial case from a large retail chain with slow-moving products, we propose a stochastic UFLP model integrated with inventory and recourse decisions. A heuristic procedure based on Louveaux and Peeters (1992) is implemented, adjusted and tested. Based on its experimental results, we propose to embed the heuristic procedure into the sample average approximation (SAA) method. First, it is not practical to enumerate all possible scenarios in our two-stage stochastic problem; Second, computational results show that the heuristic procedure provides a tight lower bound and other features suitable for SAA method. Experiment results of both the standard SAA method and the SAA method with heuristic procedure are reported.

We consider the integrated location-inventory problem of a large retail chain for spare parts, which have a low turnover rate. In the supply chain, there are a few national Distribution Centers (DCs) and hundreds of stores. Each DC keeps all the SKUs, which are stock keeping units. The stores can be divided into two types: the Super Store (SS) is larger compared with the Regular Store (RS) and keeps more SKUs. We focus on slow moving SKU. We assume that the demand arrives according to Poisson process at each store. Usually, an RS does not keep the SKU. Once a demand happens in an RS, three scenarios could happen. The RS could get one unit

SKU from a nearby SS, or from a national DC, or from a third-party logistics (3PL). The distribution system structure is illustrated in Figure 1.1.

Note that the lead times and costs of the different sources (SS, DC, or 3PL) are quite different. The approximate leading times and costs are listed in Table 1.1.

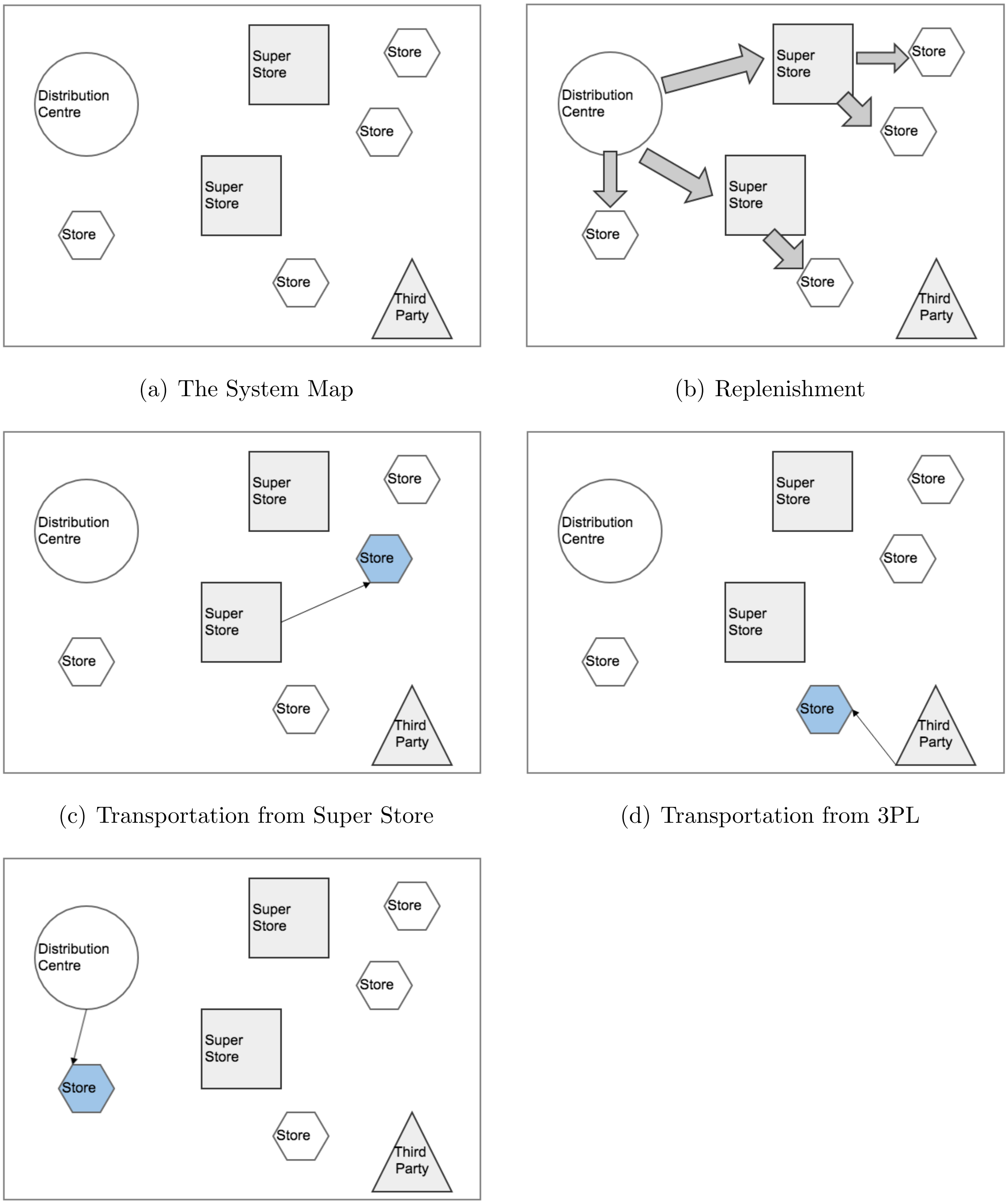
Table 1.1: Lead times and costs of different sources

Source	**Lead Time**	**Procurement Cost**
DC	Around one day	Cheapest
SS	Hours, less than a day	Medium
3PL	Hours, less than a day	Most expensive

Our objective is to minimize the expected total costs under demand uncertainty. The decisions can be classified into two categories. Before the random demand is realized, we select SSs to store the SKU and the inventory levels of the SKU at the selected SSs. After the random demand is realized, we decide the supplier, which is among the selected SSs, to satisfy the demand. This setting leads to a two-stage stochastic facility location problem with inventories and penalties (SFLPILP).

Under the assumption of slow-moving demands, our model features two aspects: first, it models uncertainty of demands using a two-stage stochastic facility location model with recourse; second, it integrates inventory decisions into the classical uncapacity facility location model. Although there exists splendid literatures on each seperated stream of research area, there is a few literature combining inventory decisions with location decisions, especially viewing inventory decisions as strategic decisions. Another contribution is that we successfully embed the dual heuristic procedure into the sample average approximation method and demonstrate that the SAA with heuristic procedure method is more efficient with no significant negative impact on solution quality.

In the following chapters of this paper, chapter 2 gives a brief literature review on related research; chapter 3 presents our extended mixed-integer stochastic facility location model; chapter 4 illustrates sample average approximation method; chapter 5 describes details of a dual heuristic procedure and its adjustment; chapter 6 reports computational results on the standard sample average approximation procedure, and the SAA with heuristic procedure.



(a) The System Map

(b) Replenishment

(c) Transportation from Super Store

(d) Transportation from 3PL

(e) Transportation from DC

Figure 1.1: Distribution System Structure

Chapter 2

Literature review

Supply chain (SC) design involves three levels of decisions: strategic, tactical, and operational decisions. Strategic decisions, usually refering to location decisions, are long-term decisions and are made independently from tactical and operational decisions, which usually refer to inventory and transportation decisions. Discrete facility location models are naturally applicable for strategic supply chain design; Melo *et al.* (2009) provides a thorough analysis on literatures using facility location models to tackle strategic supply chain design problems. However, Farahani *et al.* (2015) pointed out that this traditional way of modeling strategic, tactical, and operational decisions might cause sub-optimal solutions; and, integrating location and inventory decisions is a new trending research direction.

2.1 Risk-pooling effects

It is not an innovatory idea to configure an SS as a retailer-based distribution centre (DC) in a retail chain supply chain network. A retailer-based DC is a retailer who acts as a DC, receiving replenishment directly from warehouses or higher level of DCs, and distributing products to retailers assigned to it. It does not lose its function as a retailer; it can serve customers by itself.

Eppen (1979) first introduce the term 'Risk-pooling effect' where centralization can decrease inventory cost significantly in a decentralized system. Based on the Newsvendor model, they simulate inventory cost required for a centralized system and a decentralized system. In the decentralized system, all DCs hold inventory to satisfy their own demand, while in the centralization system only one DC hold inventory for all demand in the system.

Barahona and Jensen (1998) approach the inventory decision problem using the facility location model. Their research is motivated by a logistic system design problem for computer spare parts. They have a similar assumption as us, which is that distribution decisions are binary. This assumption is very common in the design of spare parts logistic system models. They consider multiple kinds of products. As well, inventory decisions are also assumed strictly binary. The binary programming problem is solved using the Dantzig-Wolfe decomposition method. Their main contributions are integrating inventory decisions into facility location problems, and using a subgradient optimization method to improve the convergence rate of the Dantzig-Wolfe decomposition method. However, their model is under strict assumptions, which is deterministic and has binary inventory decisions.

A retailer-based DC not only serve its own demands but also distribute inventories to other retailers which have demand request, which is exactly our configuration of Super Stores' functions. There are several benefits of this setting. Also motivated by an industrial case, Shu *et al.* (2005) explain this idea that instead of maintaining inventories at every single retailer, pooling inventory at several selected retailers, which are referred to as retailer-based DCs, can decrease inventory costs, improve service levels, and shorten service waiting times. They study a two-stage scenario-based stochastic model to minimize the expected total cost including converting retailers to retailer-based DCs, holding inventory, and transporting products. The fixed conversion cost includes training the personnel to handle this new line of business, upgrading facilities to fit this new logistic tasks and others related. At the first stage, several retailer-based DCs are chosen from all retailers upon all possible scenarios under uncertain demands. Then, transportation decisions are made after the realisation of random parameters. The inventory holding cost is integrated into the formulation of their model as a cost component in the objective function, which is expressed as a concave function of transportation decision variables. In this way, the model takes the impact of inventory holding cost into consideration while selecting locations. Snyder *et al.* (2007) describe a similar and more generic model.

2.2 Location-inventory problem

Recently, there is a growing amount of research focusing on integrating inventory decisions into facility location models.

Daskin *et al.* (2002) introduce an integrated model for generic high-volume demand in which the replenishment cost from suppliers, working inventory holding cost, and safety stock holding cost at DCs are calculated using Economic Order Quantity (EOQ) model. The total amount of inventory requests faced by each DC is expressed by the summation of distribution decisions. Therefore all the inventory related cost components are non-linear functions of distribution decisions. These cost components are then incorporated into the object function of the traditional uncapacitated facility location model, which leads to a non-linear FLP problem. The problem is solved using Lagrangian solution algorithm. Jayaraman (1998), who proposes a similar model, uses a linear function to represent inventory related cost components. For more literature about high-volume demand integrated location-inventory problems, refer to Farahani *et al.* (2015).

Nozick and Turnquist (2001) point out that for lower-demand products it is more beneficial to centralize inventories compared to higher-demand products. The paper uses two separate models for inventory problems and location problems respectively. The inventory model provides an estimation value, which serves as cost coefficients in the location model. In this way, they integrate the effect of inventory policy into location decision process. Gzara *et al.* (2014) study an integrated location-inventory problem for a service-part logistics system, which is closely related to our research. The assumption of only slow-moving and high-value items are considered distinguishes it from most of the other literature on this topic. They also state that for low-volume and random demands, which is the same as our assumption, it is sufficient to use base-stock $(s-1, s)$ policies to approximate inventory cost instead of using the economic-order-quantity reorder-point policy used for high demand items (see Graves (1985), Sherbrooke (2006)). Differing from their approach, where they use probability constraints to control service levels under uncertain demands, we use the scenario approach combined with recourse actions to evaluate lost sale costs.

Only slow moving, intermittent and unitary demands are considered in our paper, that is to say that demands are unpredictable, scarce, and only one unit request per demand occurrence. This assumption applies to many situations, like vehicle maintenance, repair service, spare parts, and service parts. For example, when a vehicle or a machine breaks down it will require only the one specific spare part for that specific model of the vehicle or machine. We use base stock replenishment policy as our inventory management policy. We assume that the supply chain system has low-volume demands, that the centre warehouses are always well stocked, and that replenishments are frequent and periodic. Recourse actions are modeled to control the system-wise service quantity by setting penalties for unmet demands. One of the difficulties solving integrated location-inventory models is that inventory considerations bring in non-linearities (Erlebacher and Meller (2000), Miranda and Garrido (2004), Shu *et al.* (2005)), which is avoided for slow-moving products.

2.3 Stochastic discrete facility location problem

Our model is based on uncapacitated facility location problem (UFLP), see equation (2.1).

$$\begin{array}{lll}
\min & \sum_i f_i y_i + \sum_i \sum_j c_{ij} x_{ij} & \\
\text{s.t.} & \sum_i x_{ij} = 1 & \forall j \\
 & x_{ij} \le y_i & \forall i,j \\
 & y_i \in \{0,1\} & \forall i \\
 & x_{ij} \in \{0,1\} & \forall i,j
\end{array} \tag{2.1}$$

In the equation, index $i \in I$ is the index for potential facilities, and set I is the set of all potential facilities; index $j \in J$ is the index for demand locations, and set J is the set of all demand locations. All parameters are deterministic, where f_i represents fixed open cost for facility i, and c_{ij} represents transportation cost from facility i to demand location j. For decision variables,

$$y_i = \begin{cases} 1, & \text{if facility } i \text{ is open} \\ 0, otherwise \end{cases}$$

$$x_{ij} = \begin{cases} 1, & \text{if demand } j \text{ is served by facility } i \\ 0, otherwise. \end{cases}$$

UFLPs chose the optimal facility locations and distribution plans to minimize location and allocation cost under the assumption that each facility has unlimited capacity. Revelle *et al.* (2008) summarize some basic facility location models and the recent development of their applications in their paper.

Two-stage stochastic programming is naturally suitable for facility location problems under unceratinty. Location decisions, which involve deploying large amount of resources and incur a relatively high cost, usually remain effective for a long period of time. During this period, other decisions may need to be made according to the current situations, like demand or cost parameters at that time. As such, at the first stage location decisions are made for all possible scenarios; at the second stage the other decisions are made for each scenario, like inventory, allocation, transportation, or penalty decisions.

For two-stage stochastic facility location problems, one aspect distinguishes them from each other, which is that decisions belong to the first stage. Stochastic transportation-location problems are the ones in which both transportation and location decisions are the first stage decisions, that is transportation plans are determined for all possible situations. Albareda-Sambola *et al.* (2011) also consider allocation decisions as first-stage decisions, which are decided before the observation of random Bernoulli distribution demands. Unsatisfied demands are then outsourced when the total requirements exceed the capacity of a facility. Under the assumption that random demand variable distributions for each demand location are homogeneous, their optimization formulation can be re-formed as a mixed-integer programming problem. Bieniek (2015) considers a similar problem with a more general random demand distribution assumption. Louveaux (1986), Snyder (2006), Melo *et al.* (2009), and Laporte *et al.* (2015) describe both recent developments and applications of stochastic FLP models.

Louveaux (1986) proposes a scenario-based two-stage stochastic FLP to represent uncertainties of demands, profits, and transporation costs with an objective to maximize the system's profit. Two distinguished features of their model are that both locations and capacities are decided before the realisation of uncertain events, and the most profit actions are taken after random events are observed.

Their model is known as the Stochastic Simple Plant Location Problem (SSPLP) as follows.

$$
\begin{array}{lll}
\max & -\sum_j g_j z_j - \sum_j f_j x_j + \mathbf{E}_\xi \max \sum_i \sum_j d_i(p_i - c_{ij}) y_{ij} & \\
\text{s.t.} & \sum_j y_{ij} \leq 1 & \forall i \\
& \sum_i d_i y_{ij} - z_j \leq 0 & \forall j \\
& y_{ij} - x_j \leq 0 & \forall i, j \\
& x_j \in \{0, 1\} & \forall i \\
& y_{ij} \geq 0 & \forall i, j \\
& z_j \geq 0 & \forall j
\end{array}
$$

x and z represent location and capacity decisions, respectively, which are made before random demand d, price p, and c are realised. After random variables are observed, distribution decisions y are made to maximize the objective profit. The unique feature about this model is that inventory decisions are first-stage decisions, while most location-inventory problems assume that inventory decisions are operational, that is to say, second-stage decisions. Based on this formulation, we extended it by allowing recourse actions and restricted our distribution decision variables as binary variables and inventory decision variables as integral variables. Since we are considering slow-moving Bernoulli distribution demand, it does not make sense to use fractional distribution decisions.

Louveaux and Peeters (1992) later extend an heuristic algorithm based on the famous dual ascent procedure developed for deterministic facility location problem (Erlenkotter (1978)) to solve the stochastic FLP model. Recent improvements on dual heuristic procedure can be found at Cánovas *et al.* (2007), Janáček and Buzna (2008), Letchford and Miller (2012), Marques and Dias (2013).

One difficulty in discrete stochastic problems is that the number of scenarios can be too large to be solvable. Kleywegt *et al.* (2002) first introduce the sample average approximation (SAA) method using sampling approach to conquer this difficulty. Ahmed *et al.* (2002) study the application of SAA specifically for two-stage stochastic problems with integer recourse. Both of their studies show that the SAA algorithm converges under certain parameter settings. Based on their studies, we implemented the SAA algorithm for our model, and creatively integrated the dual heuristic into SAA algorithm. We showed that the integrated SAA algorithm has a better performance.

Chapter 3

The model

Among all the stochastic facility location models, Louveaux and Peeters (1992) attracts our attention, because capacity decisions are made at the first stage in their model, which is consist with our base stock inventory policy. In our model, inventory at each opened facility is assumed to be replenished up to a determined level as an initial status of the system.

3.1 Notations

We use i to denote one SS, where $i \in I$ and $|I| = N$; j denotes a RS, where $j \in J$ and $|J| = M$. $s \in S$ is the index of scenarios in the second stage of the two-stage stochastic program; p_s is the probability of scenario s. Sets I, J, and S are such the set of SSs, RSs, and scenarios. We use d_{js} to represent the demand at RS j, where $d_{js} = 0$ means that RS j has zero demand under scenario s, and $d_{js} = 1$ means that RS j has demand request under scenario s. Note that the demand arrives according to a Poisson process, so that either $d_{js} = 0$ or $d_{js} = 1$. Once the demand arrives, we can satisfy the demand from SS, DC or 3PL. If the demand is satisfied by DC or 3PL, we can tell which one to choose by comparing their costs and the customer waiting time requirements. See table (3.1) for a complete list of notations.

Table 3.1: Notations

Sets(Index)	
i	index of SS, $i \in I$
j	index of RS, $j \in J$
s	index of Scenario, $s \in S$
Decision variables	
y_i	binary variables
	$y_i = 1$ if and only is SS i has positive stock.
V_i	nonnegative integer variables
	that indicates inventory level of SS i.
x_{ijs}	binary variables
	$x_{ij} = 1$ if and only if the demand at RS j is satisfied by SS i.
z_{js}	binary variables
	$z_j = 1$ if and only if the demand at RS j is satisfied by DC or 3PL.
Parameters	
f_i	fixed cost of SS i
h_i	Holding cost of one unit in SS i
t_{ij}	Transportation cost from SS i to RS j
c_{js}	The cost of satisfying the demand at RS j by DC or 3PL.

3.2 Formulation

The optimization problem (P) is represented as follows:

$$
\begin{aligned}
\min \quad & \sum_i f_i y_i + \sum_i h_i V_i + \sum_s p_s \sum_j (\sum_i t_{ij} x_{ijs} + c_{js} z_{js}) && && (3.1)\\
\text{s.t.} \quad & x_{ijs} \leq y_i && \forall i, j, s && (3.2)\\
& \sum_j x_{ijs} \leq V_i && \forall i, s && (3.3)\\
& \sum_i x_{ijs} + z_{js} \geq d_{js} && \forall j, s && (3.4)\\
& y_i \in \{0, 1\} && \forall i && (3.5)\\
& z_{js} \in \{0, 1\} && \forall j, s && (3.6)\\
& x_{ijs} \in \{0, 1\} && \forall i, j, s && (3.7)\\
& V_i \in \mathbb{Z}_+ && \forall i. && (3.8)
\end{aligned}
$$

The objective funciton (3.1) consists of fixed costs for configuring an RS as a retailer-based DC, inventory holding costs, transportation costs, and either DC or 3PL cost. Constraint (3.2) indicates that only when an SS is configured as retailer-based DC, its inventory can be transported to satisfy a demand of RS. Constraint (3.3) requires that the total amount of SKUs transported from an SS can not exceed its inventory level. Constraint (3.4) requires that a demand will always be satisfied either from an SS within its feasible range, or a DC, or a 3PL. Constraints (3.5), (3.6), (3.7) and (3.8) define the decision variables respectively. Note that we use $\mathbb{Z}_+$ to represent nonnegative integer variables.

Chapter 4

Sample Average Approximation algorithm

Solving our discrete stochastic problem exactly is not practically possible, as the scale of the number of scenarios makes it computationally prohibitive. If we only consider two possibilities for each demand location, with demand or no demand, there would be 2^M different scenarios, where M is the number of demand locations. That is without considering varieties brought by varying waiting time. Ahmed *et al.* (2002) studies the Sample Average Approximation (SAA) Method for stochastic problems with integer recourse, which provides us theoretical foundations for applying SAA method to our problem. For stochastic problems with discrete and finite first stage decision variables and recourse decision variables, it is shown that the probability of an SAA problem solution being a true optimal solution approaches one exponentially fast as the size of SAA problem increases. They also point out that solutions of SAA problems are all candidate optimal solutions of the true problem, where the true problem refer to the original problem. The SAA problem with sample size N for our problem is shown in equation (4.1).

$$
\begin{aligned}
\min \quad & \textstyle\sum_i f_i y_i + \sum_i h_i V_i + \frac{1}{N}\sum_{s=1}^{N}\sum_j(\sum_i t_{ij}x_{ijs} + c_{js}z_{js}) & \\
\text{s.t.} \quad & x_{ijs} \le y_i & \forall i,j,s \\
& \textstyle\sum_j x_{ijs} \le V_i & \forall i,s \\
& \textstyle\sum_i x_{ijs} + z_{js} \ge d_{js} & \forall j,s \\
& y_i \in \{0,1\} & \forall i \\
& z_{js} \in \{0,1\} & \forall j,s \\
& x_{ijs} \in \{0,1\} & \forall i,j,s \\
& V_i \in \mathbb{Z}_+ & \forall i.
\end{aligned}
\tag{4.1}
$$

The main idea of SAA method is quite simple. Since we know the solution of our SAA problem is potentially optimal with positive probability, we can replicate and solve SAA problems until it reaches a stopping criteria. With M replications and p probability that an SAA problem solution is optimal, the probability that at least one of the solutoins is optimal is $1-(1-p)^M$ (Kleywegt *et al.* (2002). Random sample variables are generated independently, the corresponding SAA problems are solved and a lower bound is estimated based on their solutoins.

The estimated lower bound for our original problem (P) is given as follow, using the notation convention used for SAA method in (Kleywegt *et al.* (2002)). Using $\hat{f}_N$ to represent the SAA problem based on a sample with size N, the expected value of the optimal objective value of SAA problem for M iterations provides us the estimated lower bound, $\bar{f}_N^M$ is:

$$\bar{f}_N^M = \frac{1}{M}\sum_{m=1}^{M}\hat{f}_N^m. \tag{4.2}$$

For every iteration, paired with one SAA problem, another random sample is generated independently with the sample size N', and the corresponding problem, called reference problem, is solved. The reference problem, given in equation (4.3), provides an estimate upper bound for the SAA problem solution. The upper bound $\hat{f}_{N'}$ for the first stage decisions $(\hat{y}, \hat{V})$ is given as:

$$\hat{f}_{N'}(\hat{y}, \hat{V}) = \sum_i f_i\hat{y}_i + \sum_i h_i\hat{V}_i + \frac{1}{N'}\sum_{n=1}^{N'} Q(\hat{y}, \hat{V}, \xi^n), \tag{4.3}$$

where $Q(\hat{y}, \hat{V}, \xi^n)$ is the second-stage recourse function

$$\begin{aligned}
Q(\hat{y}, \hat{V}, \xi^n) \quad = \min \quad & \textstyle\sum_j(\sum_i t_{ij}x_{ij} + c_j(\xi)z_j) & \\
\text{s.t.} \quad & \hat{y}_i - x_{ij} \geq 0 & \forall i, j \\
& \hat{V}_i - \textstyle\sum_j x_{ij} \geq 0 & \forall i \\
& z_j + \textstyle\sum_i x_{ij} \geq d_j(\xi) & \forall j, \\
& z_j \in \{0, 1\} & \forall j \\
& x_{ij} \in \{0, 1\} & \forall i, j.
\end{aligned}$$

With the lower and upper bound, the optimality gap is estimated by

$$\hat{f}_{N'}(\hat{y}, \hat{V}) - \bar{f}_N^M \tag{4.4}$$

Details of the SAA algorithm are given in algorithm (1).

An important measurement parameter on the solution quality is the variance of

Algorithm 1 SAA Algorithm

1: Initialize:
2: sample size N and N';
3: iteration number M.
4: **for** $m = 1, ..., M$ **do**
5: Generate random sample $\xi^1, ..., \xi^N$;
6: Solve the SAA problem (4.1) to optimal, and record its solution as $(\hat{y}, \hat{V})_N^m$ and objective value as $\hat{f}_N^m$;
7: Generate random sample $\xi^1, ..., \xi^{N'}$;
8: Evaluate $\hat{f}_{N'}^m(\hat{y}, \hat{V})$ using (4.3);
9: **end for**
10: Evaluate $\bar{f}$ using (4.2);
11: **for** $m = 1, ..., M$ **do**
12: Calculate the estimated optimality gap for solution $(\hat{y}, \hat{V})_N^m$ using (4.4);
13: **end for**
14: Return the candidate solution with the smallest estimated optimality gap

SAA problem solutions and reference problem solutions. Based on Ahmed *et al.* (2002), the solutoin variance can be estimated by equation (4.5):

$$\frac{1}{M(M-1)}\sum_{m=1}^{M}(\hat{f}_N^m - \bar{f}_N^M)^2 + \frac{1}{N'(N'-1)}\sum_{n=1}^{N'}\left[Q(\hat{y}, \hat{V}, \xi^n) - \frac{1}{N'}\sum_{n=1}^{N'}Q(\hat{y}, \hat{V}, \xi^n)\right]^2. \tag{4.5}$$

Solving SAA problems to optimal at every iteration can be extremely time consuming. We propose an SAA method with dual heuristic algorithm, the dual embedded SAA algorithm, in which SAA problems are solved by a heuristic algorithm to being near-optimal. This heuristic procedure provides a lower bound for the objective and a feasible primal solution of the SAA problem, which is suitable for replacing step (6) in algorithm (1). We use $LB(\hat{f}_N)$ to represent the lower bound given by the heuristic procedure for an SAA problem with sample size N, and $(\hat{y^\epsilon}, \hat{V^\epsilon})$ to represent its corresponding near-optimal primal solution within ϵ optimality gap. The upper bound is thusly estimated as $\hat{f}_{N'}(\hat{y^\epsilon}, \hat{V^\epsilon})$, similar to equation (4.3), and the lower bound is estimated as $\frac{1}{M}\sum_{m=1}^{M} LB(\hat{f}_N^m)$ similar to equation (4.2). The rest of algorithm (1) remains the same.

Near-optimal solutions from the heuristic procedure for the SAA problems are potential optimal solutions for the real problem. We know that the feasibility of the near-optimal solutions for SAA problems are guaranteed by the heuristic procedure. And, for stochastic programs with integer recourse, it is proved by Ahmed *et al.* (2002) that feasible solutions for SAA problems are also feasible solutions for the real problem. Kleywegt *et al.* (2002) states that as $N \to \infty$, near-optimal solutions $(\hat{y_N^\epsilon}, \hat{V_N^\epsilon})$ belong to the set of all feasible solutions for the original problem within ϵ optimality gap. As such, we can conclude that near-optimal solutions for SAA problems are potential optimal solutions for the real problem. Although we can not be certain of the convergent rate of heuristic solutions, experimental results show that it has similar performance as the standard SAA algorithm.

Similar to the reasoning for the selection process from optimal solutions of SAA problems, with a positive probability p^ϵ that the near-optimal solution for SAA problem is the optimal solution for the real problem, the probability that at least one of the solutions is indeed optimal for the real problem is $1-(1-p^\epsilon)^M$. In the following section, we discuss the heuristic procedure.

Chapter 5

A primal-dual algorithm

In this section, a primal-dual heuristic algorithm is proposed to solve the two-stage stochastic facility location problem with penalties and inventories. The procedure is inspired by Erlenkotter (1978) for deterministic uncapacitated problems and Louveaux and Peeters (1992) for extended stochastic uncapacitated problems. The algorithm was original designed for stochastic problems with linear recourse functions, and our problem features integer recourse function. However, as shown in Louveaux and Peeters (1992), the primal-dual heuristic algorithm naturally returns integral solutions, which is suitable for our problem with some minor adjustments.

5.1 Dual procedures

The condensed dual problem formulation is essential for dual ascent heuristic procedure, which is shown as below.

$$
\begin{aligned}
Z_D \quad = \quad \max \quad & \sum_j \sum_s d_{js} v_{js} && && (5.1)\\
\text{s.t.} \quad & \sum_j \sum_s \max\{0, v_{js} - u_{is} - t'_{ijs}\} \leq f_i && \forall i && (5.2)\\
& \sum_s u_{is} \leq h_i && \forall i && (5.3)\\
& v_{js} \leq c'_{js} && \forall j, s && (5.4)\\
& u_{is} \geq 0 && \forall i, s && (5.5)\\
& v_{js} \geq 0 && \forall j, s. && (5.6)
\end{aligned}
$$

Comparing our condensed dual problem with the one in Louveaux and Peeters (1992), there are some differences deriving from their original problems. Our model requests that all demand are satisfied either by allocaitons from SS or by penalty actions from 3PL, whereas their model only require an allocation resulting in a maximum profit. Therefore, there are three obvious differences: first, demand random variables are integrated in the objective function of our condensed dual problem; second, there is an upper boundary on the value of variable v. The dual variable v represents the marginal cost of shipping one unit of product from a SS and it should not exceed 3PL cost. Third, the coefficients of dual variable u are uniform in the first constraints. Tcha and Yoon (1985) extend the dual heuristic procedure on a more complicated situation, where coefficients of both the objective and constraint of the condensed dual problem are nonuniform. By associating each dual variable with a measure based on the value of their coefficients, they distinguish their priorities in the dual ascent procedure. Our case is quite straightforward, for the dual variables with zero coefficients, $d_{js} = 0$, do not contribute any improvement on the objective value, so we can simply omit them, that is, set them to be zero.

Similar to the approach by Louveaux and Peeters (1992), we have two separate procedures corresponding to v and u variables. In the v-ascent procedure, the objective is increased by increasing the value of v until it is blocked by constraints; in the u-ascent procedure, the slackness constraints are loosened as much as is allowed by increasing the value of u. In the following section, we present both dual heuristic procedures. The combination of both procedures return the final result for the condensed dual problem, based on which we can construct solutions for the primal problem.

5.1.1 The v-ascent procedure

Based on Louveaux and Peeters (1992), we fix the value of u and try to increase the value of v. For ease of discussion, we introduce the slack variables s_i of (A.10) written as

$$s_i = f_i - \sum_j \sum_s \max\{0, v_{js} - (u_{is} + t'_{ijs})\}, \quad \forall i \in I.$$

The value of slack variables decreases while we increase the value of variable v till it reaches zero. At the beginning, each v_{js} is set to $v_{js} = \min_{i \in I}\{u_{is} + t'_{ijs}\}$ for all $j \in J$ and $s \in S$ if $d_{js} = 1$; otherwise, v_{js} is set to be zero. When this procedure is repeated in a subsequent iteration, the initial v_{js} values are set to the last computed values.

Starting from an initial and feasible solution, the v-ascent procedure repeatedly increases the value of $\{v_{js} | d_{js} \neq 0\}$ to the next higher value of $u_{is} + t'_{ijs}$. The procedure terminates when an increase in the value of v leads to a violation of constraints (A.10) or (A.12). The procedure improves the dual objective value monotonically since we only increase the v varaibles with positive coefficient in the objective function. Values of $t'_{ijs} + u_{is}$ are computed and ordered nondecreasingly for each pair of index j and s similar to Louveaux and Peeters (1992).

A detailed description of the algorithm is provided in algorithm (2) and (3).

Algorithm 2 Initialization

1: Calculate and order sequence $q_{js}(*)$ for each pair of index (j, s).
2: Initialize $v_{js} \leftarrow q_{js}(1)$ for each $j \in J, s \in S$, or the nearest value in sequence q_{js} to its last computed result.
3: Initialize $s_i \leftarrow f_i - \sum_j \sum_s max\{0, v_{js} - (u_{is} + t'_{ijs})\}$ for each $i \in I$
4: Initialize index $k(j, s) \leftarrow \min\{k : v_{js} \leq q_{js}(k)\}$ for all $j \in J, s \in S$

5.1.2 The u-ascent procedure

In u-ascent procedure, the value of the v variables are fixed and u variables are updated to create more slack values of the first constraints.

The slack variables s_i of (A.10) can be written as

$$s_i = f_i - \sum_j \sum_s \max\{0, (v_{js} - t'_{ijs}) - u_{is}\} \quad \forall\ i.$$

For a given i^*, we would like to minimize $\sum_j \sum_s \max\{0, (v_{js} - t'_{i^*js}) - u_{i^*s}\}$ to increase the value of s_{i^*}. The optimization problem can be written as:

$$\begin{array}{ll} \min & \sum_j \sum_s \max\{0, (v_{js} - t'_{i^*js}) - u_{i^*s}\} \\ \text{s.t.} & \sum_s u_{i^*s} \leq h_{i^*} \\ & u_{i^*s} \geq 0 \qquad\qquad\qquad\qquad \forall s. \end{array}$$

The main step of u-ascent procedure is decreasing the value of u_{i^*s} where $(v_{js} - t'_{i^*js}) -$

Algorithm 3 The v-ascent procedure

repeat
        $\delta \leftarrow 0$
    for all j such that $j \in J^+$ **do**
        for all s such that $s \in S^+$ **do**
            if $d_{js} > 0$ **and** $v_{js} < c'_{js}$ **then**
                $\Delta \leftarrow \min_{i \in I}\{s_i : v_{js} - (u_{is} + t'_{ijs}) \geq 0\}$
                if $\Delta > q_{js}(k(j,s)+1) - v_{js}$ **or** $\Delta > c'_{js} - v_{js}$ **then**
                    if $q_{js}(k(j,s)+1) > c'_{js}$ **then**
                        $\Delta \leftarrow c'_{js} - v_{js}$
                    else
                        $\Delta \leftarrow q_{js}(k(j,s)+1) - v_{js}$
                        $k(j,s) \leftarrow k(j,s)+1$
                    end if
                    $\delta \leftarrow 1$
                    for $i \in \{i : v_{js} - (u_{is} + t'_{ijs}) \geq 0\}$ **do**
                        $s_i \leftarrow s_i - \Delta$
                    end for
                    $v_{js} \leftarrow v_{js} + \Delta$
                end if
            end if
        end for
    end for
until $\delta = 0$
return $Z_D^{new} = \sum_j \sum_s d_{js} v_{js}$

$u_{i^*s} > 0$ if it is allowed by constraints $\sum_s u_{i^*s} \leq h_{i^*}$. A detailed description is given in algorithm (4). The initial idea of the algorithm is to increase the value of u_{i^*s} where $(v_{js} - t'_{i^*js}) - u_{i^*s}$ is the smallest positive value. However, finding the minimum value of $(v_{js} - t'_{i^*js}) - u_{i^*s} > 0$ itself requires $O(|J| * |S|)$ complexity. Instead, we increase the first u_{i^*s} where $(v_{js} - t'_{i^*js}) - u_{i^*s} > 0$, which shortens the computational time significantly.

Algorithm 4 The u-ascent procedure

initialization
    $g_{i^*} \leftarrow h_{i^*} - \sum_s u_{i^*s}$
    $q_{i^*js} = t'_{i^*js} + u_{i^*s}$ for all (j, s)
repeat
  $\delta \leftarrow 0$
  $(\Delta, s') \leftarrow \{(v_{js} - q_{i^*js}, s) | v_{js} - q_{i^*js} > 0\}$
  $\Delta \leftarrow \min\{\Delta, g_i^*\}$
  $u_{i^*s'} \leftarrow u_{i^*s'} + \Delta$
  $g_i^* \leftarrow g_i^* - \Delta$
  if $g_i^* \geq 0$ **then**
    $\delta \leftarrow 1$
  end if
  for all j **do**
    if $v_{js'} - q_{i^*js'} > 0$ **then**
      Slack variables from v-ascent procedure are updated by $s_{i^*} = s_{i^*} + \Delta$
    end if
    $q_{i^*js'} \leftarrow q_{i^*js'} + \Delta$
  end for
until $\delta = 0$

5.1.3 The u-v ascent procedure

By applying v-ascent and u-ascent procedure sequentially, we have v-u ascent procedure, which increases the objective value of the condensed dual problem nondecreasingly and returns a feasible solution.

5.2 Construction of primal feasible solution

By the complementary slackness theorem from linear programming theory (Luenberger and Ye (1984)), the primal optimal solution of the linear relaxation problem and the dual optimal solution of the condensed dual problem must satisfy the following conditions:

$$y_i \times s_i = 0 \qquad \forall i \tag{5.7}$$

$$V_i \times (\sum_s u_{is} - h_i) = 0 \qquad \forall i \tag{5.8}$$

$$z_{js} \times (v_{js} - c'_{js}) = 0 \qquad \forall j, s \tag{5.9}$$

$$\max\{0, v_{js} - u_{is} - t'_{ijs}\} \times (y_i - x_{ijs}) = 0, \qquad \forall i, j, s \tag{5.10}$$

$$u_{is} \times (V_i - \sum_j x_{ijs}) = 0, \qquad \forall i, s \tag{5.11}$$

$$v_{js} \times (\sum_i x_{ijs} + z_{js} - d_{js}) = 0 \qquad \forall j, s \tag{5.12}$$

We define eligible facilities as $I^* = \{i \in I | s_i = 0\}$ and the subset $I^+ \in I^*$ where for each $\{(j, s)|d_{js} \neq 0\}$, there exists some $i \in I^+$ such that $v_{js} > t'_{ijs} + u_{is}$. The construction of I^+ follows a revised procedure of Erlenkotter (1978) and is described in algorithm (5).

For each demand location j and scenario s, the minimum cost facility location $i^+(j, s) \in I^+$ is defined by :

$$i^+(j, s) = \underset{i \in I^+}{\operatorname{argmin}}\{t'_{ijs} + u_{is}\}. \tag{5.13}$$

When we try to decide which facility to serve a demand for a location and scenario

Algorithm 5 I^+ construction procedure

Define $B_{js} \leftarrow \{i \in I^* \mid t'_{ijs} + u_{is} \leq v_{js}\}$ for all $\{(j,s) \mid d_{js} \neq 0\}$
$I^+ \leftarrow \emptyset$
for all $j \in J$ **do**
  for all $s \in S$ **do**
    if $B_{js} = \{i\}$ **then**
      $I^+ = I^+ + \{i\}$
    end if
    if $|B_{js}| > 1$ **and** $B_{js} \bigcap I^+ = \emptyset$ **then**
      $i^+ = \text{argmin}_i\{t'_{ijs} + u_{is} | i \in I^*\}$
      $I^+ = I^+ + \{i^+\}$
    end if
  end for
end for

(j,s), the natural thought is to pick the minimum cost facility location, suggested by Louveaux (1986). However, in our discrete stochastic problem, this approach might not be optimal or near optimal. Under different scenarios, inventories at each facility have a pooling effect. That is, the inventory decision made at one scenario prevade all the other scenarios. Under a single scenario, the scenario-based optimal solution might be adding an extra unit of inventory to the closet facility to the new added demand. However, if there already exists extra inventories in a sub-optimal facility, it would be more economic to use the existing inventory instead of increasing any inventory levels. We propose algorithm (6) to construct transportation decisions and inventory decisions, where extra inventories are used prior to the increasement of any inventory level. We use V_{is} to represent the inventory used at facility i under scenario s.

Since we assume that inventory holding costs are relatively smaller than facility opening costs, which is a practical assumption, u-ascent procedure guarantees that $\sum_s u_{is} - h_i = 0$ when $s_i = 0$. By applying complementary slackness conditions

Algorithm 6 x_{ijs} and V_i construction algorithm

```
Initialize x_{ijs} = 0, V_i = 0, V_{is} = 0 for all i, j, s.
for all {j, s | d_{js} = 1 and v_{js} > 0} do
    if There is {i | V_i > V_{is}} then
        i^+ = argmin_{i∈I^+} {t'_{ijs} | V_i > V_{is}}
        x_{i^+js} = 1
        V_{i^+s} = V_{i^+s} + 1
    else
        i^+ = argmin_{i∈I^+} {t'_{ijs} + u_{is}}
        x_{i^+js} = 1
        V_{i^+s} = V_{i^+s} + 1
        if V_{i^+s} > V_{i^+} then
            V_{i^+} = V_{i^+s}
        end if
    end if
end for
```

(5.7)-(5.9) and (5.12), the primal solution is constructed as

$$
\begin{aligned}
y_i &= \begin{cases} 1 & \text{if } i \in I^+ \\ 0 & \text{otherwise} \end{cases} \\
z_{js} &= \begin{cases} 1 & \text{if } v_{js} = c'_{js} \\ 0 & \text{otherwise} \end{cases} \\
x_{ijs} &= \begin{cases} 1 & \text{if } c'_{js} > v_{js} > 0 \text{ and } i = i^+(j,s) \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
\tag{5.14}
$$

From the above, complementary slackness conditions (5.7)-(5.9) and (5.12) are satisfied. However, conditions (5.10) and (5.11) might be violated. Conditions (5.10) are violated when there are more than one $i \in I^+$ satisfying $t'_{ijs} + u_{is} \leq v_{js}$, and conditions (5.11) are violated when there are different units of transportation request for a facility under different scenarios, that is $V_{is} < V_i$ for some $s \in S$.

Lemma 1. *Let (u, w, v) be the dual feasible solution resulting from our dual-ascent procedures, and (y, V, x, z) be the primal feasible solution we construct from the above. The gap $Z_P - Z_D$ between the primal objective value Z_P and the dual objective value Z_D is bounded by*

$$\sum_i \sum_s (V_i - \sum_j x_{ijs}) u_{is} + \sum_i \sum_j \sum_s \max\{0, v_{js} - u_{is} - t'_{ijs}\}(y_i - x_{ijs}) \tag{5.15}$$

Proof. As we know (y, V, x, z) is feasible, $Z_P = \sum_i f_i y_i + \sum_i h_i V_i + \sum_s \sum_j \sum_i t'_{ijs} x_{ijs}$. And both the dual solution and primal soluton satisfy complementary slackness conditions (5.7)-(5.9) and (5.12), we have

$$\begin{aligned} & y_i * s_i = 0 && \forall i \\ & V_i * (\sum_s u_{is} - h_i) = 0 && \forall i \\ & z_{js} * (v_{js} - c'_{js}) = 0 && \forall j, s \\ & v_{js} * (\sum_i x_{ijs} + z_{js} - d_{js}) = 0 && \forall j, s \end{aligned}$$

which equal to

$$\sum_i y_i * (f_i - \sum_j \sum_s w_{ijs}) = 0 \tag{5.17a}$$

$$\sum_i V_i * (\sum_s u_{is} - h_i) = 0 \tag{5.17b}$$

$$\sum_j \sum_s z_{js} * (v_{js} - c'_{js}) = 0 \tag{5.17c}$$

$$\sum_j \sum_s v_{js} * (\sum_i x_{ijs} + z_{js} - d_{js}) = 0. \tag{5.17d}$$

Such that,

$$\begin{aligned}
& Z_P - Z_D \\
=& Z_P - (5.17a) + (5.17b) + (5.17c) - Z_D - (5.17d) \\
=& \sum_i \sum_s V_i u_{is} + \sum_i \sum_j \sum_s w_{ijs} y_i \\
& + \sum_i \sum_j \sum_s t_{ijs} x_{ijs} - \sum_i \sum_j \sum_s v_{js} x_{ijs} \\
=& \sum_i \sum_s V_i u_{is} - \sum_i \sum_j \sum_s u_{is} x_{ijs} \\
& + \sum_i \sum_j \sum_s w_{ijs} y_i \\
& - \sum_i \sum_j \sum_s (v_{js} - u_{is} - t_{ijs}) x_{ijs} \\
& \text{where } i \in I^+, j \in J, s \in S
\end{aligned} \tag{5.18}$$

With $w_{ijs} = max\{0, v_{js} - u_{is} - t'_{ijs}\}$ and $v_{js} - u_{is} - t'_{ijs} \geq 0,$ where $i \in I^+$, we have

$$\begin{aligned} & Z_P - Z_D \\ = & \sum_i \sum_s (V_i - \sum_j x_{ijs}) u_{is} + \sum_i \sum_j \sum_s max\{0, v_{js} - u_{is} - t'_{ijs}\}(y_i - x_{ijs}) \end{aligned} \tag{5.19}$$

□

Chapter 6

Computational results

All computations are conducted on a PC with 2.6 GHz Intel Core i5 processor and 8GB 1600 MHz memory. Both the SAA algorithm and the dual heuristic algorithm are programmed using $C++$ programming language. We also compared our algorithms with a general solver, IBM ILOG CPLEX Optimization Studio 12.6.2, which is also implemented in C++ programming language using Concert technology to call CPLEX solver.

Since there are no existing problem instances for our model, instances are generated randomly within a given range according to table (6.1). The total number of demand requests under one scenario are generated according to Poisson Distribution with a mean value λ. Then, demand locations with demand request are chosen randomly without exceeding the total number of demand request at each scenario. Cost parameters are generated according to uniform distribution within the determined range. Unless explicitly indicated, all instances are generated according to this table.

Table 6.1: Parameter settings

Number of potential facility locations	30
Number of demand locations	100
Mean demand occurance at one scenario	40
Open cost	Uniform[200,300]
Hold cost	Uniform[50,10]
Transportation	Uniform[10, 100]
3PL or DC cost	Uniform[10, 100]

6.1 SAA algorithm

We first examine the performance of the standard SAA algorithm briefly, where both the SAA problem and the reference problem at each iteration are solved to optimal using the general sovler. We use $N' = 1000$ as the sample size for reference problems for all instances. Experiment instances with different parameters, N for the sample size of SAA problems and M for the number of replications, are repeated four times. We study cases where $N = 20, 40$, and $M = 20, 40$. Because all instances find their estimated optimal solution, that is the optimality gap is zero, we therefore focus on the statistical quality of the results, which is measured by variance. The results are reported in table (6.2). Column M is the number of replication; Column N is the sample size for SAA problem; Column Avg Var is the average variance of four instances under the same parameter M and N. The smaller the variances, the more reliable the optimal solution. From the table, both the increase of M and N improves the quality of results; and, the change of N has a better impact than M. Especially when $N = 40$, the variance improves a little from 109.35 to 93.78 where M doubles from 20 to 40.

Table 6.2: Variance of SAA statistical optimal solutions with varying M and N

M	N	Avg Var
20	20	191.19
20	40	109.35
40	20	130.15
40	40	93.78

6.2 SAA algorithm with dual heuristic and its improvement

In this section, the standard SAA algorithm and SAA algorithm with dual heuristic are compared in table (6.3). All instances are with $M = 20$, $N = 40$, and $N' = 1000$ for the ease of comparison. Column λ is the mean demand request for each scenario; Column Opt. is the estimated optimal objective for the real problem; Column Var. is the variance of SAA problems; Column Time is the computational time. It shows that SAA with dual heuristic is almost one hundred times faster than the standard SAA algorithm where every SAA problem is solved to optimal. However, the estimated optimal solution selected by SAA with dual heuristic is not as good as standard SAA, especially when λ increases. This consists with our observation of the dual heuristic algorithm. When demand request increases, the original dual heuristic algorithm perform poorly on allocating inventories.

As such, we improved the inventory decision logic introduced in section (5.2), and similar comparison is given in table (6.4) for the SAA algorithm with improved dual heuristic. It is surprising that with improved heuristic the estimated solutions are sometimes even better than the standard SAA algorithm. We then focused on experimental results for SAA with improved heuristic algorithm.

Table 6.3: SAA and SAA with dual heuristic results

	SAA			SAA with dual heuristic		
λ	Opt.	Var.	Time(sec)	Opt.	Var.	Time(sec)
20	1006.23	186.03	86.89	1006.23	187.03	1.53
40	1778.53	187.87	134.43	1784.41	206.17	1.98
60	2292.26	170.14	297.97	2372.90	181.35	2.50

Table 6.4: SAA and SAA with improved dual heuristic results

	SAA			SAA with improved heuristic		
λ	Opt.	Var.	Time(sec)	Opt.	Var.	Time(sec)
20	994.38	135.85	87.74	1020.34	134.32	1.51
40	1683.49	198.09	214.62	1671.74	177.61	2.05
60	2238.59	187.65	222.97	2171.36	169.40	2.57

Table (6.5) shows results emphasizing estimated optimal objective gaps between the standard SAA algorithm and SAA with improved dual heuristic. All experiments have parameter $M = 20$, $N' = 1000$, and $\lambda = 40$. Column N is the value of N; and, Column Gap is the relative gap of the estimated optimal objective between SAA with improved heuristic and the standard SAA algorithm. Negative values of Gap mean that the estiamted optimal objective of SAA with improved dual heuristic are better than the standard SAA algorithm. From the results, it can be concluded that SAA with improved heuristic is not worse than the standard SAA algorithm based on the measure of optimality gap, solution variance and computational time.

Table (6.6) shows results similar to the previous table (6.5). It provides extra computational results for different problem instances. Column Fac. represents the number of potential facility location; Column Loc. represent the number of demand location. The meaning of the rest columns are the same as table (6.5). The unspecified parameters are as default, where $M = 20$, $N = 60$, $N' = 1000$ and $\lambda = 40$.

Table 6.5: SAA with improved dual heuristic results

	SAA			SAA with improved heuristic			
N	Opt.	Var.	Time(sec)	Opt.	Var.	Time(sec)	Gap
60	1749.28	111.49	286.46	1747.38	96.22	3.48	-0.1%
60	1684.61	179.90	299.37	1677.95	168.08	3.22	-0.4%
60	1725.65	192.94	282.71	1703.36	173.72	3.26	-1.3%
70	1774.10	159.40	407.62	1785.46	134.21	4.45	0.6%
70	1715.79	106.33	532.39	1704.56	84.22	4.29	-0.7%
70	1703.84	100.25	529.14	1703.30	97.48	4.34	0.0%
80	1699.26	170.96	230.36	1696.16	145.10	4.92	-0.2%
80	1685.18	151.19	258.45	1693.51	138.73	4.54	0.5%
80	1724.58	128.36	256.65	1725.69	117.69	4.63	0.1%

Table 6.6: SAA with improved dual heuristic results (2)

		SAA			SAA with improved heuristic			
Fac.	Loc.	Opt.	Var.	Time(sec)	Opt.	Var.	Time(sec)	Gap
30	50	1445.42	144.60	254.99	1435.01	120.44	2.84	-0.7%
30	50	1471.55	106.06	267.04	1482.52	91.12	3.18	0.7%
30	50	1479.48	87.60	268.02	1473.32	77.99	3.18	-0.4%
40	50	1442.03	121.65	613.38	1455.81	106.71	4.82	1.0%
40	50	1437.83	103.06	597.90	1437.83	106.73	4.70	0.0%
40	50	1466.94	131.20	584.86	1466.90	127.47	4.65	0.0%
40	100	1697.20	139.47	954.69	1665.63	106.64	5.09	-1.9%
40	100	1726.12	144.25	892.90	1727.99	130.99	6.41	0.1%
40	100	1675.76	116.85	918.39	1711.58	105.38	6.20	2.1%

At last, we examined how the values of N' affect the solution quality on both SAA and SAA with improved heuristic algorithm. Table (6.7) shows that smaller N' lead to larger solution variance, which means that the estimated optimal solutions are less reliable.

Table 6.7: SAA with varying N'

	SAA			SAA with improved heuristic			
N'	Opt.	Var.	Time(sec)	Opt.	Var.	Time(sec)	Gap
500	1722.73	163.00	237.82	1726.44	163.53	5.10	0.2%
500	1723.06	165.87	236.85	1735.64	165.95	4.75	0.7%
500	1704.87	170.27	245.59	1694.07	128.80	5.00	-0.6%
1000	1699.26	170.96	230.36	1696.16	145.10	4.92	-0.2%
1000	1685.18	151.19	258.45	1693.51	138.73	4.54	0.5%
1000	1724.58	128.36	256.65	1725.69	117.69	4.63	0.1%

Chapter 7

Conclusion

We proposed a stochastic facility location problem with inventory and recourse decisions for a multiple-location supply chain problem. To solve the large discrete stochastic problem, we studied Sample Average Approximation algorithm. Taken into the feature of a heuristic procedure, we combined these two algorithms. Furthermore, we improved the original heuristic procedure. Our experiments show that the combined algorithm with our adjustment provides solutions which have similar quality as the standard SAA algorithm and much less computational time. We assumed that demand requests were unitary; in the future, we would like to expand the assumption to small integral number of requests. Another direction for future research is expanding the assumption of one product to multiple products.

Appendix A

Condensed dual problem

Follow Louveaux and Peeters (1992) and Erlenkotter (1978), to obtain a natural integral solution, we will apply the dual-primal heuristic procedure on a condensed dual problem from the relaxation problem of the original problem (P). The relaxation problem (RP) is obtained by relaxing the integral constraints on variables y, V, x, z to nonnegative constraints. The formulation of problem (RP) is given as below.

$$
\begin{array}{lll}
\min & \sum_i f_i y_i + \sum_i h_i V_i + \sum_s \sum_j \sum_i t'_{ijs} x_{ijs} + \sum_s \sum_j c'_{js} z_{js} & \\
\text{s.t.} & y_i - x_{ijs} \geq 0 & \forall i, j, s \\
& V_i - \sum_j x_{ijs} \geq 0 & \forall i, s \\
& z_{js} + \sum_i x_{ijs} \geq d_{js} & \forall j, s \\
& V_i \geq 0 & \forall i \\
& y_i \geq 0 & \forall i \\
& z_{js} \geq 0 & \forall j, s \\
& x_{ijs} \geq 0 & \forall i, j, s,
\end{array}
$$

where we substitute $t'_{ijs} = p_s t_{ij}$ and $c'_{js} = p_s c_{js}$ to simplify the formulation. The dual problem of (RP) is the following:

$$Z_D = \max \sum_{j \in J} \sum_{s \in S} d_{js} v_{js} \tag{A.1}$$

$$\text{s.t.} \quad v_{js} - u_{is} - w_{ijs} \leq t'_{ijs} \qquad \forall i, j, s \tag{A.2}$$

$$\sum_j \sum_s w_{ijs} \leq f_i \qquad \forall i \tag{A.3}$$

$$\sum_s u_{is} \leq h_i \qquad \forall i \tag{A.4}$$

$$v_{js} \leq c'_{js} \qquad \forall j, s \tag{A.5}$$

$$w_{ijs} \geq 0 \qquad \forall i, j, s \tag{A.6}$$

$$u_{is} \geq 0 \qquad \forall i, s \tag{A.7}$$

$$v_{js} \geq 0 \qquad \forall j, s, \tag{A.8}$$

where the w, u, and v variables are dual variables corresponding to constraints (3.2), (3.3) and (3.4), respectively. As showed in Spielberg (1969),we can replace w_{ijs} by $\max\{0, v_{js} - u_{is} - t'_{ijs}\}$ without affecting the dual problem's feasibility and objective value to have the condensed form of the dual problem. The condensed dual problem formulation is essential for dual ascent heuristic procedure, which is shown as below.

$$
\begin{aligned}
Z_D \quad = \quad \max \quad & \sum_j \sum_s d_{js} v_{js} && && \text{(A.9)} \\
\text{s.t.} \quad & \sum_j \sum_s \max\{0, v_{js} - u_{is} - t'_{ijs}\} \leq f_i && \forall i && \text{(A.10)} \\
& \sum_s u_{is} \leq h_i && \forall i && \text{(A.11)} \\
& v_{js} \leq c'_{js} && \forall j, s && \text{(A.12)} \\
& u_{is} \geq 0 && \forall i, s && \text{(A.13)} \\
& v_{js} \geq 0 && \forall j, s. && \text{(A.14)}
\end{aligned}
$$

Bibliography

Ahmed, S., Shapiro, A., and Shapiro, E. (2002). The sample average approximation method for stochastic programs with integer recourse. *Submitted for publication*, pages 1–24.

Albareda-Sambola, M., Fernández, E., and Saldanha-da Gama, F. (2011). The facility location problem with bernoulli demands. *Omega*, **39**(3), 335–345.

Barahona, F. and Jensen, D. (1998). Plant location with minimum inventory. *Mathematical Programming*, **83**(1-3), 101–111.

Bieniek, M. (2015). A note on the facility location problem with stochastic demands. *Omega*, **55**, 53–60.

Cánovas, L., García, S., and Marín, A. (2007). Solving the uncapacitated multiple allocation hub location problem by means of a dual-ascent technique. *European Journal of Operational Research*, **179**(3), 990–1007.

Daskin, M. S., Coullard, C. R., and Shen, Z.-J. M. (2002). An inventory-location model: Formulation, solution algorithm and computational results. *Annals of operations research*, **110**(1-4), 83–106.

Eppen, G. D. (1979). Note-effects of centralization on expected costs in a multi-location newsboy problem. *Management Science*, **25**(5), 498–501.

Erlebacher, S. J. and Meller, R. D. (2000). The interaction of location and inventory in designing distribution systems. *Iie Transactions*, **32**(2), 155–166.

Erlenkotter, D. (1978). A dual-based procedure for uncapacitated facility location. *Operations Research*, **26**(6), 992–1009.

Farahani, R. Z., Rashidi Bajgan, H., Fahimnia, B., and Kaviani, M. (2015). Location-inventory problem in supply chains: a modelling review. *International Journal of Production Research*, **53**(12), 3769–3788.

Graves, S. C. (1985). A multi-echelon inventory model for a repairable item with one-for-one replenishment. *Management science*, **31**(10), 1247–1256.

Gzara, F., Nematollahi, E., and Dasci, A. (2014). Linear location-inventory models for service parts logistics network design. *Computers & Industrial Engineering*, **69**, 53–63.

Janáček, J. and Buzna, L. (2008). An acceleration of erlenkotter-körkel's algorithms for the uncapacitated facility location problem. *Annals of Operations Research*, **164**(1), 97–109.

Jayaraman, V. (1998). Transportation, facility location and inventory issues in distribution network design: An investigation. *International Journal of Operations & Production Management*, **18**(5), 471–494.

Kleywegt, A. J., Shapiro, A., and Homem-de Mello, T. (2002). The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, **12**(2), 479–502.

Laporte, G., Nickel, S., and da Gama, F. S. (2015). *Location science*, volume 145. Springer.

Letchford, A. N. and Miller, S. J. (2012). Fast bounding procedures for large instances of the simple plant location problem. *Computers & Operations Research*, **39**(5), 985–990.

Louveaux, F. (1986). Discrete stochastic location models. *Annals of Operations research*, **6**(2), 21–34.

Louveaux, F. V. and Peeters, D. (1992). A dual-based procedure for stochastic facility location. *Operations research*, **40**(3), 564–573.

Luenberger, D. G. and Ye, Y. (1984). *Linear and nonlinear programming*, volume 2. Springer.

Marques, M. d. C. and Dias, J. M. (2013). Simple dynamic location problem with uncertainty: a primal-dual heuristic approach. *Optimization*, **62**(10), 1379–1397.

Melo, M., Nickel, S., and Saldanha-da-Gama, F. (2009). Facility location and supply chain management – a review. *European Journal of Operational Research*, **196**(2), 401–412.

Miranda, P. A. and Garrido, R. A. (2004). Incorporating inventory control decisions into a strategic distribution network design model with stochastic demand. *Transportation Research Part E: Logistics and Transportation Review*, **40**(3), 183–207.

Nozick, L. K. and Turnquist, M. A. (2001). A two-echelon inventory allocation and distribution center location analysis. *Transportation Research Part E: Logistics and Transportation Review*, **37**(6), 425–441.

Revelle, C. S., Eiselt, H. A., and Daskin, M. S. (2008). A bibliography for some fundamental problem categories in discrete location science. *European Journal of Operational Research*, **184**(3), 817–848.

Sherbrooke, C. C. (2006). *Optimal inventory modeling of systems: multi-echelon techniques*, volume 72. Springer Science & Business Media.

Shu, J., Teo, C.-P., and Shen, Z.-J. M. (2005). Stochastic transportation-inventory network design problem. *Operations Research*, **53**(1), 48–60.

Snyder, L. V. (2006). Facility location under uncertainty: a review. *IIE Transactions*, **38**(7), 547–564.

Snyder, L. V., Daskin, M. S., and Teo, C.-P. (2007). The stochastic location model with risk pooling. *European Journal of Operational Research*, **179**(3), 1221–1238.

Spielberg, K. (1969). Algorithms for the simple plant-location problem with some side conditions. *Operations Research*, **17**(1), 85–111.

Tcha, D.-W. and Yoon, M.-G. (1985). A dual-based heuristic for the simple facility location problem with stochastic demand. *IIE transactions*, **17**(4), 364–369.

Weber, A., Friedrich, C. J., *et al.* (1929). Alfred weber's theory of the location of industries.