# INTEGRATED SUPPLY CHAIN MODELS AND ANALYSIS

by

JIANING ZHI

BURCU B. KESKIN, COMMITTEE CHAIR
CHARLES R. SOX
JOHN MITTENTHAL
VOLODYMYR MELNYKOV
GLENN RICHEY

A DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Information Systems,
Statistics, and Management Science
in the Graduate School of
The University of Alabama

TUSCALOOSA, ALABAMA

2016

© Copyright by Jianing Zhi, 2016.
ALL RIGHTS RESERVED.

## ABSTRACT

This dissertation reports on three integrated supply chain problems that incorporate several key components of modern supply chains, including location, transportation, inventory, and customer selection. The first part of the dissertation investigates a multi-product, three-stage distribution system with transshipment and direct shipping from supply centers to customers. The objective is to determine the locations of a fixed number of capacitated warehouses to minimize the total transportation and fixed facility costs in the supply chain network. The second part of the dissertation focuses on the integrated location-inventory problem in a multi-retailer newsvendor setting with both decentralized and centralized decision making. The third part of the dissertation explores the coordination between operations management and marketing through an integration of marketing strategies and the inventory decisions to maximize the total expected profit of the company.

The contribution of this dissertation is four-fold:

- First, we define two new problems with respect to integrated decision making in supply chain management: one combines inventory and location decisions based on two designs of supply chain network, and the other one studies the interface of operations management and marketing on top of a selective newsvendor problem with quantity dependent lead time. For both problems, we offer mathematical models and effective solution approaches.

- Second, we develop two meta-heuristic solution approaches for a multi-product production/distribution system design (PDSD) problem, which has been studied in literature and solved with Scatter search and Tabu search. We propose to solve the problem

with two meta-heuristic procedures, simulated annealing (SA) and greedy randomized adaptive search procedure (GRASP), which demonstrate better solution quality and time performance compared to Scatter Search.

- Third, we establish a practical connection between operations and marketing in the selective newsvendor problem. This effort demonstrates that a joint decision-making process is more profitable, and opens up more cooperation opportunities between the two disciplines.

- Lastly, the proposed mathematical models, solution approaches, as well as managerial insights for either new problems or existing problems will potentially shed light on the research of problem variants and the development of new techniques beyond those considered in this dissertation.

## ACKNOWLEDGMENTS

I am pleased to have this opportunity to express my appreciation to the faculty members, family, colleagues, and friends who have been offering their support and help throughout the five-year course of my graduate studies.

First of all, I would like to thank my advisor, Dr. Burcu B. Keskin, for her professional training and continuous encouragement in the past five years. There is no way I can sharpen my research skills and widen my scientific vision without numerous research meetings and technical discussions with Dr. Keskin. Apart from that, Dr. Keskin has been influencing me with her enthusiasm on research and teaching, her rigorous attitude toward details, and her persistence and diligence on work.

Besides my advisor, I would also like to thank the rest of the committee members, Dr. Charles R. Sox, Dr. John Mittenthal, Dr. Volodymyr Melnykov, and Dr. Glenn Richey, for their invaluable suggestion, inspiring questions, and constructive feedback of both the dissertation and my academic progress. In addition, I am extremely grateful to the faculty and staff in the Department of Information Systems, Statistics, and Management Science for all of their help and support. In particular, I am grateful to Dr. Sharif Melouk, whose guidance and encouragement have been important to my success.

This research would not have been possible without the support and encouragement of my family. My deepest appreciation goes to my parents for their love and support throughout my life. Last but not the least, I am particularly grateful to my husband, for being my strongest back.

## CONTENTS

**ABSTRACT** . . . . . ii

**ACKNOWLEDGMENTS** . . . . . iv

**LIST OF TABLES** . . . . . viii

**LIST OF FIGURES** . . . . . x

**1 INTRODUCTION** **1**

**CHAPTER 1 REFERENCES** . . . . . 5

**2 SIMULATED ANNEALING AND GRASP APPROACHES FOR A MULTI-PRODUCT PRODUCTION/DISTRIBUTION SYSTEM DESIGN PROBLEM** **6**

2.1 **INTRODUCTION** . . . . . 6

2.2 **LITERATURE REVIEW** . . . . . 8

2.3 **THE MODEL** . . . . . 10

2.3.1 General Model . . . . . 10

2.3.2 A Special Case . . . . . 14

2.4 **SIMULATED ANNEALING HEURISTIC** . . . . . 15

2.4.1 Step 0: Initial Parameters . . . . . 15

2.4.2 Step 1: Initial Seed Solution Construction . . . . . 15

2.4.3 Step 2: Local Search . . . . . 16

2.4.4 Step 3: Pre-checks and Neighbor Selection Criteria . . . . . 16

2.4.5 Step 4: Candidate Evaluation and Update . . . . . 17

2.5 **GRASP HEURISTICS** . . . . . 21

2.5.1 Step 0: Randomized Greedy Selection . . . . . 21

2.5.2 Step 1: Local Search . . . . . 22
2.5.3 Step 2: Solution Evaluation and Update . . . . . 22
2.6 **COMPUTATIONAL RESULTS** . . . . . 22
2.6.1 Solving the PDSD Problem with SA and GRASP . . . . . 24
2.6.2 Overall Performance of SA and GRASP . . . . . 24
2.6.3 Parameter Analysis . . . . . 26
2.6.4 Numerical Study of The Special Case . . . . . 36
2.7 **CONCLUSIONS** . . . . . 41
**CHAPTER 2 REFERENCES** . . . . . 42

**3 INTEGRATED WAREHOUSE LOCATION AND INVENTORY DECISIONS IN A MULTI-LOCATION NEWSVENDOR PROBLEM 45**
3.1 **INTRODUCTION** . . . . . 45
3.2 **LITERATURE REVIEW** . . . . . 49
3.2.1 Multi-Location Newsvendor Problem . . . . . 49
3.2.2 Location-Inventory Problem . . . . . 51
3.3 **GENERAL ASSUMPTIONS AND NOTATION** . . . . . 52
3.4 **MODELS AND SOLUTIONS** . . . . . 54
3.4.1 Model 1: Decentralized System Model (DSM) . . . . . 54
3.4.2 Solution of DSM . . . . . 56
3.4.3 Model 2: Centralized System Model (CSM) . . . . . 58
3.4.4 Solutions of CSM . . . . . 59
3.4.5 Choosing a Retailer as DC . . . . . 64
3.5 **A COMPARISON OF DSM AND CSM** . . . . . 65
3.5.1 Case 1: quantity-dependent transportation cost . . . . . 65
3.5.2 Case 2: distance-dependent transportation cost . . . . . 69
3.5.3 Case 3: quantity-distance-dependent transportation cost . . . . . 70
3.6 **COMPUTATIONAL EXPERIMENTS** . . . . . 70

3.6.1 Data Setting . . . . . 71
3.6.2 DSM and CSM: a Comparison for Case 3 . . . . . 71
3.6.3 Retailer-as-DC: a Potentially Better Option . . . . . 73
3.6.4 Parameter Sensitivity Analysis . . . . . 75
3.7 **CONCLUSION** . . . . . 78
**CHAPTER 3 REFERENCES** . . . . . 79

**4 SELECTIVE NEWSVENDOR PROBLEM WITH DEPENDENT LEADTIME AND JOINT MARKETING DECISIONS 81**
4.1 **INTRODUCTION** . . . . . 81
4.2 **LITERATURE REVIEW** . . . . . 84
4.2.1 Selective Newsvendor Models . . . . . 84
4.2.2 Marketing and Inventory Joint Decision Models . . . . . 85
4.2.3 Service Level Constraint . . . . . 86
4.3 **PROBLEM DESCRIPTION AND MODEL FORMULATION** . . . 87
4.3.1 All-or-Nothing Selective Newsvendor Problem (AON-SNP) . . . . . 88
4.3.2 Selective Newsvendor Problem with Dependent Lead Time and Price Related Demands(DL-SNP) . . . . . 90
4.4 **EVALUATION** . . . . . 99
4.4.1 Data Setting . . . . . 100
4.4.2 Results . . . . . 100
4.4.3 Testing the R-search Algorithm . . . . . 103
4.4.4 Parameter Sensitivity Analysis . . . . . 104
4.5 **CONCLUSION** . . . . . 109
**CHAPTER 4 REFERENCES** . . . . . 110

**5 CONCLUSION 112**

## LIST OF TABLES


1.1 Research Topic . . . . . 2

2.1 Related Work . . . . . 8

2.2 Data Setting . . . . . 23

2.3 Summary of Solution Quality . . . . . 25

2.4 Summary of Running Time Improvement . . . . . 26

2.5 Shipping Assignment for DS 1 - DS6 . . . . . 30

2.6 Shipping Assignment for DS 7 - DS 11 . . . . . 31

2.7 A t-test on $x_{jil}$ with Two Levels of $p$ . . . . . 32

2.8 A t-test on $x_{jil}$ with Two Levels of $K$ . . . . . 34

2.9 A t-test on $x_{jil}$ with Two Levels of $N$ . . . . . 35

2.10 Data Setting of Special Case . . . . . 36

2.11 Overall Comparisons between SA and SS . . . . . 37

2.12 SA with Two-neighbor Search . . . . . 38

2.13 SA with Three-neighbor Search . . . . . 38

2.14 Overall Comparisons between GRASP and SS . . . . . 39

2.15 GRASP with One Swap Local Search . . . . . 40

2.16 GRASP with One and Two Swap Local Search . . . . . 40

3.1 Data Specification . . . . . 71

3.2 DSM Vs. CSM: Optimal Expected Profits . . . . . 72

3.3 DSM Vs. CSM: Profits with Random Demand Samples . . . . . 73

3.4 CSM Case 3: Optimal DC Vs. Retailer-as-DC . . . . . 74

3.5 CSM Profit Comparison: Optimal DC Vs. Retailer-as-DC . . . . . 75
3.6 Sensitivity Analysis . . . . . 75
3.7 Difference of CSM and DSM . . . . . 77

4.1 Data Specification . . . . . 100
4.2 Results for the Small Data Set . . . . . 101
4.3 Results for the Large Date Set . . . . . 102
4.4 Parameter Sensitivity Analysis . . . . . 105
4.5 Impact of Market Size on Instance ($I = 4$, $J = 50$) . . . . . 105
4.6 Impact of Agent Capacity on Instance ($I = 4$, $J = 100$) . . . . . 106
4.7 Impact of Price Scale $\lambda$ on Instance ($I = 4$, $J = 50$) . . . . . 108
4.8 Impact of Waiting Time on Instance ($I = 4$, $J = 50$) . . . . . 108

## LIST OF FIGURES


2.1 A Two-stage Supply Chain Network . . . . . 11

2.2 Flow Chart of the Greedy Solver . . . . . 18

2.3 A Performance Comparison of SA, GRASP, and CPLEX on a Particular Instance 25

2.4 Box and Whisker Charts of SA and GRASP in Comparison to CPLEX . . . 27

2.5 Effect of Parameters Settings on Total Cost . . . . . 29

2.6 Effect of the Number of Supply Centers on Shipping Options . . . . . 34

2.7 Optimality Gap and Running Time of All Algorithms . . . . . 41

3.1 Decentralized and centralized systems . . . . . 46

3.2 Optimal DC Vs. Retailer-as-DC on Instance n=10 . . . . . 74

3.3 Impact of $\gamma$ . . . . . 76

3.4 Impact of Map Size . . . . . 78

4.1 Big Picture of the Selective Newsvendor Network . . . . . 88

4.2 R-search for Instance (I=4, J=100) with a running time of 1.2 second . . . . 104

4.3 Naive Search for Instance (I=4, J=100) with a running time of 13.9 seconds 104

4.4 Impact of Base Price on Instance ($I = 4$, $J = 50$) . . . . . 107

4.5 Impact of Unit Production Time on Instance ($I = 4$, $J = 50$) . . . . . 107

# CHAPTER 1

# INTRODUCTION

Competition in global markets, the introduction of products with shorter life cycles, and heightened expectations of customers has forced modern enterprises to focus on supply chain operations. Enterprises start to combine and coordinate different departments such as inventory, marketing, and transportation, to be more responsive and cost-efficient. The integration of supply chain not only means the coordination with partners from upper and lower tiers, but it also means making joint decisions strategically with other functional departments of the company. Therefore, the topic of coordination between inventory and other departments has received interest from both researchers and practitioners, since the early 2000s. Issues like capacity decisions and demand management are not just marketing or operations issues, but rather lie on the interface between the two. Explanatory, empirical, and analytical research between the two functions is required to provide a necessary foundation on which new frameworks and theory can be established. This coordination is expected to benefit the company as a whole since it helps maintain and sustain a healthy, stable customer relationship.

Currently, many companies adopt new expansion trends to serve the needs of their customers in a cost-effective way and to increase market share. In this dissertation, we study cross-functional interfaces and develop integrated decision models because such research is useful from both tactical and strategic perspectives. A primary example can be seen from capital allocation and capacity expansion problems. We can divide the supply chain coordination problem into two categories: (i) the vertical coordination between partners/firms

Table 1.1: Research Topic

| Horizontal Coordination | Vertical Coordination | |
|---|---|---|
| | Two-Stage | Three-Stage |
| OM Decisions | Chapter 3: DSM | Chapter 2; PDSD, Chapter 3: CSM |
| OM & Marketing | Chapter 4: SNP(AON) | Chapter 4: S-DLSNP |

from different tiers in a supply chain; and (ii) the horizontal coordination between functional departments inside a company. This dissertation follows a three article format and presents new problems and solution approaches incorporating the coordination issues within a supply chain. Table 1.1 shows the dissertation research topics within different SC categories. We consider (i) integration within operations decisions (i.e., transportation, inventory, and location) in a multi-stage supply chain network in Chapters 2 and 3, and (ii) coordination across OM and marketing decisions in both two-stage and three-stage supply chains in Chapter 4. General descriptions of the three articles follow.

I. Simulated Annealing and GRASP Approaches for A Multi-product Production/ Distribution System Design Problem (PDSD): In Chapter 2 of the dissertation, we study a multi-product, two-stage production/distribution system design problem. We determine the locations of a fixed number of capacitated local distribution centers to minimize the total transportation and fixed facility costs. Production distribution networks provide an effective tool for modeling the manufacturing and logistics activities of a firm. The network not only includes the supply centers and central DCs, but also includes local warehouses and retailers (customers) themselves. The multi-stage PDSD is widely applied in the assembly industry and supply chain management. In this problem, we consider four types of transportation flows: sending multiple products from supply centers to local DCs or directly to retailers, from local DCs to retailers, and transshipments between two local DCs. We assume that the locations of supply centers and retailers are known, and capacities of each supply center and demand of retailers are also given. There exists a set of potential locations for DCs with capacity

limitations. We determine the combination of opened DCs to minimize the total cost of the network, including location costs of DCs and transportation costs occurred in both stages. We formulate the problem as a mixed integer linear programming model. We use simulated annealing (SA) and greedy randomized adaptive search (GRASP) procedures to solve large problem instances. Although there have been a number of prior studies on PDSD problems, there have not been adequate studies involving solutions using meta-heuristic approaches. Inspired by Tabu Search, we create long and short memory lists to improve the searching procedure in both SA and GRASP algorithms. One long-term memory list is used to avoid repetitive searching, and two short-term memory lists are refreshed at each iteration to keep the qualified candidates. Both SA and GRASP outperform our benchmarking algorithm, and these hybrid algorithms can be applied to other p-median problems besides PDSD.

II. Integrated Warehouse Location and Inventory Decisions in a Multi-location Newsvendor Problem: In Chapter 3 of the dissertation, we consider an integrated location-inventory problem in a multi-retailer newsvendor setting with both decentralized and centralized decision making. Only in the last decade have the three logistical decisions of facility location, inventory, and transportation decisions been integrated into a single system view. However there are still some open research problems in this area. Note that the facility location decisions are strategic decisions, whereas inventory and transportation decisions are tactical. Addressing the coordination issues between the two types of decisions will help us to understand the supply chain better. In this chapter, we consider a multi-location newsvendor problem where the demand for each location is satisfied by the same supplier. More specifically, the model we consider consists of a single supplier and multiple newsvendors who observe a random demand for a seasonal (short-life time) product. In the original (decentralized) two-stage system (DSM), before the sales period, each retailer (newsvendor) places an order to meet the random demand independent of other retailers. The retailer determines the order

quantity in a way to maximize its profit. On the other hand, after receiving the order, the supplier ships the order to the retailer in a timely manner. This process repeats for each retailer. Therefore, all parties (the supplier and the retailers) in the supply chain maximize their gains. Many researchers note that when the demands from several retailers are consolidated, the expected holding and shortage costs of the consolidated system are lower than that of a decentralized system (Chang and Lin, 1991; Cherikh, 2000; Eppen, 1979; Lin et al., 2001). To take advantage of the centralized ordering and the economies of scale in transportation, the supplier expects each retailer to order through a distribution center (DC) whose location needs to be determined. Hence, the second problem that we consider in this chapter is a three-stage system where a DC is located to coordinate and consolidate the orders from the retailers. In the three-stage centralized system (CSM), the supplier oversees the whole supply chain and fulfills demands of retailers from the DC. The DC receives a single consolidated shipment and distributes the orders to the retailers using that shipment quantity when demand occurs. Many large retail industries including car manufacturers opt for large warehouses to distribute the ordered products to their stores and dealerships throughout the region, rather than carrying separate inventory at different locations. We develop a non-linear programming model and maximize the total profit for the whole, integrated supply chain. Through an analysis and experiment of both centralized and decentralized models under different transportation cost structures, we can question and identify the impact of the distribution center's location on the expected payoffs obtained from both centralized and decentralized systems.

III. Selective Newsvendor Problem with Dependent Lead Time and Joint Marketing Decisions: In Chapter 4 of the dissertation, we focus on the coordination between operations management and marketing. We consider a company that procures a product from a distant supplier to serve geographically distributed customers in a single selling season. The company experiences quantity-dependent lead times from the supplier, i.e., large

orders take longer than smaller ones. Due to having a limited sales force and lead time issues, the company may not be able to meet all of the customer demands at all times. Furthermore, if the waiting time grows excessively, the company may pay penalties or even, lose the business of customers in future. For this problem, we integrate marketing strategies of determining the selling price and assigning customers to sales agents with the operational inventory decisions to maximize the total expected profit of the company. We formulate a mixed integer non-linear programming model, and develop a custom solution algorithm that can efficiently identify the optimal solution. We experiment with different factors such as demands, abilities of sales agents, waiting time of customers , and lead time of production to estimate their impacts on the total expected profit, the ordering policies, and marketing strategies.

We conclude this dissertation in Chapter 5 by providing a summary of each work along with managerial insights, extensions, and future research.

## CHAPTER 1 REFERENCES


P. Chang and C. Lin. On the effect of centralization on expected costs in a multi-location newsboy problem. Journal of Operational Research Society, 42:1025–1030, 1991.

M. Cherikh. On the effect of centralization on expected profits in a multi-location newsboy problem. Journal of Operational Research Society, 51:755–761, 2000.

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

C. Lin, C. Chen, and H. Hsieh. Effects of centralization on expected profits in a multi-location newsboy problem. Journal of Operational Research Society, 52:839–841, 2001.

# CHAPTER 2

# SIMULATED ANNEALING AND GRASP APPROACHES FOR A MULTI-PRODUCT PRODUCTION/DISTRIBUTION SYSTEM DESIGN PROBLEM

## 2.1. INTRODUCTION

A production distribution network serves as an effective tool for modeling manufacturing and logistics activities of a firm. In general, a production distribution system design (PDSD) problem involves the determination of the best configuration regarding location and size of the suppliers and distribution centers (DCs), their technology, and decisions of product offerings and transportation, for achieving a firm's long-term goals (Keskin and Üster, 2007a). The involved parties not only include manufacturers and suppliers, but also includes warehouses and retailers (customers) themselves. This kind of multi-stage PDSD is widely applied in the assembly industry and supply chain management. In this paper, we consider a two-stage network including suppliers, capacitated DCs, and retailers that can order multiple types of products. In the first stage, the products are shipped from capacitated suppliers to capacitated DCs; in the second stage, they are transported from DCs to retailers to satisfy the demands of various products. We assume that the locations of suppliers and retailers are known, and that the capacities of suppliers and demands of retailers are also given. There exists a set of potential locations of capacitated DCs. We need to decide a group of opened DCs to minimize the total cost of the network, including the location costs of DCs, and transportation costs occurred in both stages. Figure 2.1 shows the basic structure of the two-stage PDSD problem.

The problem belongs to a capacitated facility location problem and can be reduced to the p-median problem which is known as NP-hard (Kariv and Hakimi, 1979). The known

complexity of the problem drives us to solve the problem via heuristic approaches to find high-quality solutions to large sized problem instances. Existing optimization solvers such as CPLEX are efficient to solve small problem instances but time-consuming for large ones. The proposed heuristic methods in this chapter can facilitate the decision-making process while still maintaining the solution quality.

In this chapter, we propose two customized meta-heuristic approaches, which are built on top of a Simulated Annealing (SA) algorithm and a Greedy Randomized Adaptive Search Procedure (GRASP) to solve the two-stage PDSD problem. Prior studies (Keskin and Üster, 2007a) and (Keskin and Üster, 2007b) have employed Scatter Search (SS) and Tabu Search algorithms to solve the PDSD problem without transhippments among DCs. However, to the best of our knowledge, no solution approaches based on SA and GRASP have been explored in the PDSD context. Moreover, we employ strategies of performance improvement for both SA and GRASP. In particular, we design a construction procedure for GRASP to generate a good initial solution, which can speed up the searching process for locating a high-quality solution. Also, we devise a greedy selection heuristic that accelerates the local search without sacrificing quality. The experimentation results show that both SA and GRASP are competitive with SS regarding both solution quality and algorithm efficiency.

The remainder of this chapter is organized as follows. In Section 2.2, we provide a review of related literature. In Section 2.3, we formulate a mixed integer linear programming (MILP) model for PDSD. In Section 2.4 and 2.5, we present the design of the SA-based and GRASP-based solution algorithms, as well as the adjustments and improvements that are integrated into the algorithms. Section 2.6 delivers the results of the computational experiments regarding the performance of the two proposed heuristics and a comparison with the SS which serves as a benchmark. In Section 2.7, we conclude the chapter and discuss future research directions.

Table 2.1: Related Work

| Literature | Source | Capacitied | Trans. | D.S. | Methods |
|---|---|---|---|---|---|
| Kariv and Hakimi (1979) | multi | | √ | | Lagr. B and B |
| Tragantalerngsak et al. (2000) | single | √ | | | Lagr. B and B |
| Diaz and Fernandez (2006) | single | √ | | | SS, path-relinking |
| Contreras and Diaz (2008) | single | √ | | | SS, Tabu, GRASP |
| Jawahar and Balaji (2009) | multi | √ | | | B.D. |
| Keskin and Üster (2007a) | multi | √ | | | Tabu |
| Keskin and Üster (2007b) | multi | √ | | | SS |
| our work | multi | √ | √ | √ | SA and GRASP |

## 2.2. LITERATURE REVIEW

There are noteworthy studies on the subject of integrated supply chain network. In this section, we review related studies of PDSD problems as well as meta-heuristic based solution approaches. Table 2.1 displays a comparison of our work and other prior studies.

For comprehensive surveys on the PDSD problems, we recommend (Goetschalckx et al., 2002), (Arostegui et al., 2006), and (Olhager et al., 2015). Kariv and Hakimi (1979) consider the location of p-transshipment points using 2-index transportation variables. They develop a solution algorithm using Lagrangian relaxation and branch-and-bound. As opposed to the ample suppliers and DCs considered in their study, our model assumes capacitated suppliers and DCs. Tragantalerngsak et al. (1997) discuss a single-product, single-source, and two-stages production/distribution problem and solve it with Lagrangian heuristics; in another paper (Tragantalerngsak et al., 2000), they solve the problem with a Lagrangian relaxation-based branch and bound algorithm. In (Diaz and Fernandez, 2006), the authors combine path relinking with SS for the capacitated p-median problem. A GRASP-based method is used to generate an initial solution. Later, Contreras and Diaz (2008) propose a SS-based heuristic which provides upper bounds for the optimal solution of the single source capacitated facility location problem. Brimberg and Drezner (2013) develop a new local search procedure that avoids using random starting points for solving a continuous p-median problem. Chiyoshi and Galvao (2000) present a statistical analysis of SA applied to the p-median problem.

They combine the vertex substitution method with the general SA procedure. Fahimnia et al. (2012) model a multi-period two-echelon supply chain problem with a mixed integer non-linear programming problem. They consider direct shipment from suppliers to retailers, and also discuss the inventory management at the suppliers and backlogs at end users. In their model, decisions to be made include the production shipment quantities, which does not concern the location strategy. Jawahar and Balaji (2009) use Genetic Algorithm to solve a single-product two-stage distribution problem with linearly increasing transportation cost and fixed cost. For a general review of heuristic methods on the p-median problem we recommend (Mladenovic et al., 2007).

Feo and Resende (1989) firstly introduce GRASP for solving a set covering problem with unit costs. The authors consider a multi-start meta-heuristic that applies local search to find an initial solution. GRASP is also used to solve NP-hard problems, such as p-median and k-covering problems (Feo and Resende, 1995). Festa and Resende (2009) provide a summary of applications of GRASP in different areas of operations research. Ahmadi and Osman (2005) combine GRASP with Adaptive Memory Programming for a capacitated clustering problem. The Adaptive Memory Programming method is used to store information for an individual element found in a pool of previously generated solutions. The stored information captures features of high-quality solutions, which are added to the initial data for the problem to enable the construction of different initial solutions. Pedrola et al. (2013) develop a GRASP algorithm with path-relinking to solve a wavelength switched optical network problem. Pessoa et al. (2013) consider a k-covering problem and solve it with the hybrid GRASP Lagrangian heuristic, which employs the GRASP with path-relinking heuristic using modified costs to obtain approximate solutions for the original problem. Saez-Aguado and Trandafir (2012) propose a hybrid heuristic that integrates local search, GRASP, and Lagrangean heuristic to solve the p-median problem with an additional coverage constraint.

Based on our survey, prior studies have explored various meta-heuristics, including Lagrangian relaxation, Scatter Search, Tabu Search, and Genetic Algorithm to develop solution approaches for different types of production/distribution problems. SA and GRASP have been good options for generating high-quality initial solutions, while no prior research designs solution approaches entirely based SA and GRASP. This chapter aims to bridge this gap. Moreover, the two proposed heuristics not only apply to the particular PDSD model considered in this chapter, they can also be easily adopted for generic production/distribution problems.

## 2.3. THE MODEL

### 2.3.1. General Model

We consider a three-level supply chain network. The first level comprises suppliers that provide multiple products to local DCs/ wholesale stores or directly to supply retailers. The second level comprises local DCs, and we take transshipments between DCs into account in our model. The third level consists of retailers with known demands of products. We need to determine the locations of $p$ selected DCs, as well as the quantity of each product. Our model considers four possible shipping flows: (1) from suppliers to local DCs, (2) from suppliers to retailers, (3) from local DCs to retailers, and (4) between two local DCs. Our objective is to minimize the total fixed location costs of DCs and transportation costs in the system. For each product, we assume a known demand that is specific at each retailer. In addition, both suppliers and local DCs have capacity limitations. We define the following notations for our model.

**Sets and Indices:**

$\mathcal{I}$: Set of retailers/customers, $i = \{1, ..., I\}$

$\mathcal{J}$: Set of potential DCs/wholesale stores, $j = \{1, ..., J\}$

$\mathcal{K}$: Set of suppliers, $k = \{1, ..., K\}$

$\mathcal{L}$: Set of products, $l = \{1, ..., L\}$

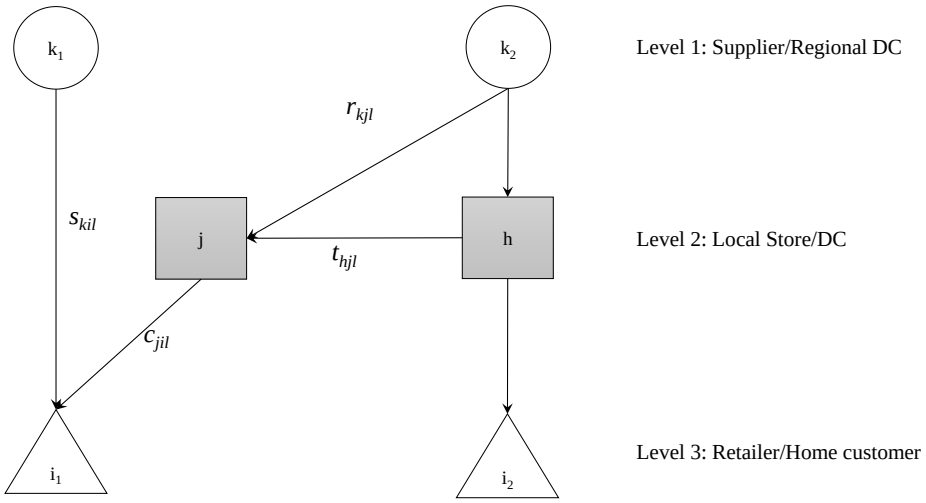



Figure 2.1: A Two-stage Supply Chain Network

**Parameters:**

$p$: number of local DCs/wholesale stores to be located

$D_{il}$: demand at retailers $i \in \mathcal{I}$ for product $l \in \mathcal{L}$

$W_j$: capacity limit at local DC $j \in \mathcal{J}$

$P_k$: capacity limit at plant $k \in \mathcal{K}$

$f_j$: fixed cost of locating a local DC at $j \in \mathcal{J}$

$c_{jil}$: unit transportation cost of product $l \in \mathcal{L}$ from DC $j \in \mathcal{J}$ to retailer $i \in \mathcal{I}$

$r_{kjl}$: unit transportation cost of product $l \in \mathcal{L}$ from supplier $k \in \mathcal{K}$ to DC $j \in \mathcal{J}$

$s_{kil}$: unit transportation cost of product $l \in \mathcal{L}$ from supplier $k \in \mathcal{K}$ to retailer $i \in \mathcal{I}$

$t_{hjl}$: unit transshipment cost of product $l \in \mathcal{L}$ from DC $h \in \mathcal{J}$ to another DC $j \in \mathcal{J}$

The model consists of the following decision variables:

**Decision Variables:**

$z_j$: 1 if DC at location j is used, 0 o.w., $j \in \mathcal{J}$

$x_{jil}$: amount of product $l \in \mathcal{L}$ transported from DC $j \in \mathcal{J}$ to retailer $i \in \mathcal{I}$

$y_{kjl}$: amount of products $l \in \mathcal{L}$ transported from supplier $k \in \mathcal{K}$ to DC $j \in \mathcal{J}$

$u_{kil}$: amount of products $l \in \mathcal{L}$ transported from supplier $k \in \mathcal{J}$ to retailer $i \in \mathcal{I}$

$v_{hjl}$: amount of products $l \in \mathcal{L}$ transported from DC $h \in \mathcal{J}$ to another DC $j \in \mathcal{J}$

Then, we can formulate the PDSD problem:

**Objective (PDSD):**

$$\begin{aligned}\min \quad F(x,y,u,v,z) &= \sum_{j\in\mathcal{J}} f_j z_j + \sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} c_{jil}x_{jil} \\ &+ \sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} r_{kjl}y_{kjl} + \sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} s_{kil}u_{kil} + \sum_{h\in\mathcal{J}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} t_{hjl}v_{hjl}\end{aligned}$$

**Subject to:**

**Retailer Demand**
$$\sum_{j\in\mathcal{J}} x_{ijl} + \sum_{k\in\mathcal{K}} u_{kil} = D_{il}, \qquad \forall i \in \mathcal{I}, \forall l \in \mathcal{L}. \tag{2.1}$$

**Flow of DC**
$$\sum_{h\in\mathcal{J}} v_{jhl} + \sum_{k\in\mathcal{K}} x_{jil} = \sum_{k\in\mathcal{K}} y_{kjl} + \sum_{h\in\mathcal{J}} v_{hjl}, \qquad \forall j \in \mathcal{J}, \forall l \in \mathcal{L}, \forall h \in \mathcal{L}. \tag{2.2}$$

**DC Capacity**
$$\sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}} y_{kjl} + \sum_{h\in\mathcal{J}}\sum_{l\in\mathcal{L}} v_{hjl} \le W_j z_j, \qquad \forall j \in \mathcal{J}. \tag{2.3}$$

**Supplier Capacity**
$$\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} y_{kjl} + \sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} u_{kil} \le P_k, \qquad \forall k \in \mathcal{K}. \tag{2.4}$$

**Open DCs**
$$\sum_{j\in\mathcal{J}} z_j = p, \tag{2.5}$$

**Stronger C1**
$$x_{ijl} \le min(D_{il}, W_j z_j), \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \forall l \in \mathcal{L}. \tag{2.6}$$

**Stronger C 2**
$$\sum_{l\in\mathcal{L}} y_{kjl} \le min(W_j z_j, P_k), \qquad \forall j \in \mathcal{J}, \forall k \in \mathcal{K}. \tag{2.7}$$

**Stronger C 3**
$$u_{kil} \le min(D_{il}, P_k), \qquad \forall i \in \mathcal{I}, \forall k \in \mathcal{K}. \tag{2.8}$$

**Stronger C 4**
$$\sum_{l\in\mathcal{L}} v_{hjl} \le min(W_h z_h, W_j z_j), \qquad \forall j \in \mathcal{J}, \forall h \in \mathcal{J}. \tag{2.9}$$

**Non-negativity**
$$x_{ijl}, y_{jkl}, u_{kil}, v_{hjl} \ge 0, \qquad \forall i \in \mathcal{I}, \forall j \in \hat{\mathcal{J}}, \forall l \in \mathcal{L}, \forall k \in \hat{\mathcal{K}}. \tag{2.10}$$

**Binary Variable**
$$z_j \in \{0,1\}, \qquad \forall j \in \mathcal{J}. \tag{2.11}$$

In the objective function, the first term represents the fixed cost of locating $p$ DCs, the second term to the fifth term represent the transportation costs from local DCs to retailers, from suppliers to local DCs, from suppliers directly to retailers, and transshipment costs among local DCs, respectively. Constraint 2.1 ensures that the demand for each product $l$ at each retailer $i$ is satisfied. Constraint 2.2 represents the flow conservation at each local DCs. Constraints 2.3 and 2.4 ensure the capacity restrictions at local DCs and suppliers, respectively. There is no specific capacity requirement for a specific product at the suppliers and DCs. Constraint 2.5 specifies the number of open DCs to be located. In constraint set 2.6, $x_{ijl}$ is given by $\min(D_{il}z_j, W_j z_j)$ for $i \in \mathcal{I}$, $j \in \mathcal{J}$ and $l \in \mathcal{L}$. These constraints state that a retailer can only be assigned to an open DC, and the size of shipment is restricted by the demand and capacity of an opened DC. In constraint set 2.7, $y_{kjl}$ is given by $\min(W_j z_j, P_k)$ for $j \in \mathcal{J}$ and $k \in \mathcal{K}$. Constraint set 2.8 is a stronger formulation for shipments between suppliers and retailers. Constraint set 2.9 is a stronger constraint for the transshipment quantities between any two local DCs. Constraint sets 2.6 to 2.9 are added to obtain a stronger formulation, and are implied by constraints 2.3, 2.4 and 2.5. Finally, constraints 2.10 and 2.11 are non-negativity and integrality constraints, respectively.

To solve this p-median multi-flow transportation problem, we provide a sub-problem that eliminates the binary variable. Let $\hat{J} \subset \mathcal{J}$ be the set of open DCs such that $|\hat{J}| = p$ and $\sum_{j\in\hat{\mathcal{J}}} W_j \geq \sum_{i\in\mathcal{I}} D_i$. Given the open DCs, the problem becomes a multi-product capacitated transhipment problem, which is relatively easier to solve since it is a linear programming formulation, which has all binary variable eliminated. We call this model Tr-PDSD:

**Objective (Tr-PDSD):**

$$\begin{aligned}\min \quad F(x,y,u,v) = &\sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} c_{jil}x_{jil} + \sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} r_{kjl}y_{kjl} \\ &+ \sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} s_{kil}u_{kil} + \sum_{h\in\mathcal{J}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} t_{hjl}v_{hjl}\end{aligned}$$

In our solution procedures, we determine the set of open DCs using SA and GRASP. To solve this capacitated transshipment problem (Tr-PDSD) for a given set of open DCs by $\hat{J}$, we obtain an exact and optimal solution via IBM CPLEX 12.6. Note that this general solution approach to our problem can be easily adopted to handle product specific capacities at suppliers and DCs since the only additional difficulty is in the sub-problem. As explained in detail in the following sections, the additional constraint will not affect the heuristic algorithms.

### 2.3.2. A Special Case

Our new model is an extension and generalized version of the model discussed in (Keskin and Üster, 2007a), which can be regarded as a special case of our work, and is used as a benchmark for comparison. This section describes the model of the special case, as well as the related solution algorithms. Compared with the generalized model, the special case does not consider direct shipment from suppliers to retailers and the transshipment among DCs. The model determines the locations of p DCs and the quantity of each product to transport from suppliers to DCs and from DCs to retailers such that the total fixed location costs and transportation costs in the system are minimized.

**Objective (Special Case):**

$$\begin{aligned} \min \quad F(x,y,u,v,z) &= \sum_{j\in\mathcal{J}} f_j z_j + \sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} c_{jil}x_{jil} \\ &+ \sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} r_{kjl}y_{kjl} \end{aligned}$$

**Subject to:**

$$\textbf{Demand Requirement} \qquad \sum_{j\in\mathcal{J}} x_{jil} = D_{il}, \qquad \forall i \in \mathcal{I}, \forall l \in \mathcal{L}, \tag{2.12}$$

$$\textbf{DC Capacity} \qquad \sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} x_{jil} \leq W_j z_j, \qquad \forall j \in \mathcal{J} \tag{2.13}$$

$$\textbf{Product Balance} \qquad \sum_{i\in\mathcal{I}}\sum_{l\in\mathcal{L}} x_{jil} = \sum_{k\in\mathcal{K}}\sum_{l\in\mathcal{L}} y_{ijl}, \qquad \forall j \in \mathcal{J} \tag{2.14}$$

$$\textbf{Plant Capacity} \qquad \sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} y_{kjl} \leq P_k, \qquad \forall k \in \mathcal{K} \tag{2.15}$$

In the following sections, we present the solution algorithms based on SA and GRASP to solve the special case, and compare the solution qualities with the benchmark algorithm in (Keskin and Üster, 2007a).

## 2.4. SIMULATED ANNEALING HEURISTIC

In this section, we present a SA-based heuristic approach to solve the PDSD problem and the special case.

### 2.4.1. Step 0: Initial Parameters

In Step 0, we initialize a list of the parameters of SA, including the initial temperature $T$, the cooling rate $r$, the final temperature $T'$, and the seed solution $\hat{J}$. The parameter setting is critical to algorithm performance. For the best effect, we test $T = [5000, 25000]$ with a step size of 500, $r \in \{0.9, 0.95\}$, and $T' \in \{2000, 5000, 7000\}$, and find the best parameter set of $T = 20000$, $r = 0.9$, and $T' = 7000$, which delivers shorter running time and better solutions.

### 2.4.2. Step 1: Initial Seed Solution Construction

The algorithm needs a seed solution in each iteration for local search. We generate the initial seed via the Random + Greedy Heuristic (RG), which is suggested by (Resende and Werneck, 2004) for p-median problems. In this heuristic, we introduce a parameter $\alpha \in [0, 1]$

that specifies the fraction of $p$ DCs to be opened in a greedy fashion, while the rest of open DCs are randomly chosen. In particular, $\alpha = 1$ means that all of the $p$ DCs are greedily chosen, and $\alpha = 0$ implies that the heuristic becomes random selection. We use the fixed cost of opening a DC (i.e., $f_j$) as the criteria for greedy ordering. In other words, we sort list $\mathcal{J}$ based on the fixed cost in increasing order. The greedy heuristic selects the first $\lceil p \cdot \alpha \rceil$ DCs in the sorted list to obtain an initial seed solution.

### 2.4.3. Step 2: Local Search

After generating an initial solution, we apply one-swap and two-swap strategies to search in the neighborhood of the initial solution. For the one-swap operation, we randomly select a pair of DCs, one from the open DCs and one from the closed DCs. Then, by closing the open DC and opening the closed one, we obtain a new solution. Similarly, for the two-swap, we exchange two open DCs with two closed ones, all chosen randomly.

To improve the efficiency of local search, we employ both long-term memory and short-term memory lists. The long-term memory list *checked-list* contains all neighbors we have visited. For each iteration, we create a short memory list: *neighbor-list*, which contains all unvisited neighbors generated by one-swap and/or two-swap on the current seed solution. As soon as we add an instance into *neighbor-list*, the same instance is also added into *checked-list*, so that when the algorithm happens to generate the same instance in another iteration, we can find it in *checked-list*, and simply skip it. Also, since we only evaluate a certain number of neighbors (denoted by $m$) in *neighbor-list*, we create the second short memory list *selected-list* to record the candidates that we select to solve in the Tr-PDSD model. For each iteration, we store $m$ candidates in the *selected-list*. Both *neighbor-list* and *selected-list* are cleared at the beginning of each iteration, while *checked-list* is never cleared for the entire run of the algorithm.

### 2.4.4. Step 3: Pre-checks and Neighbor Selection Criteria

In addition to the swap operations, we develop neighbor selection criteria. Let $\hat{J}$ be a set of selected $p$ DCs, then for an initial solution, the number of its one-swap neighbors is

$p(|J| - |\hat{J}|)$, and the number of two-swap neighbors is $\binom{p}{2}\binom{|J|-|\hat{J}|}{2}$. Searching and evaluating all of these neighbors are quite time-consuming. To improve the algorithm performance, we devise a greedy heuristic that can select a subset of "good" neighbors, which have a high potential to lead to lower total costs. After an analysis of the data setting in (Keskin and Üster, 2007a), we find out that the fixed location cost of DCs takes a large portion of the total cost, with an average of 70%. Therefore, it is reasonable to choose the total fixed location cost as criteria to design the heuristic: after both one-swap and two-swap searches, we sort all candidates in *neighbor-list* in terms of the total fixed cost in increasing order, and then pick the first $m$ neighbors. The selected $m$ neighbors are added into *selected-list*.

To ensure the initial constructions are distinct, for both SA and GRASP, we conduct a pre-check for duplication inspection, which examines whether a generated instance has already been visited previously, i.e., it is found in *checked-list*. If duplication occurs, we simply move to the next instance until a fresh instance is found.

Another pre-check is to verify the capacity constraints of suppliers and DCs. To ensure the selected $p$ DCs have sufficient capacities to satisfy all retailer demands, we compare the total capacity of selected DCs with the total retailer demands, and eliminate the candidates without enough capacities.

#### 2.4.5. Step 4: Candidate Evaluation and Update

For each iteration when the temperature does not reduce, the algorithm identifies a current seed solution $\hat{J}$, searching in its neighbors using one-swap and/or two-swap, and returns *selected-list*, which is a set of "good" solution candidates. Then, for each candidate in *selected-list*, we develop a greedy heuristic (referred to as greedy solver) to quickly evaluate the Tr-PDSD problem, and save the result in *rst-list*, which is another long-term list created to store solution results.

The idea of the greedy solver is based on a greedy strategy on shipping assignments: with a given candidate, the solver looks for the cheapest (Origin, Destination) pair ((O, D) pair); in other words, a route with lowest shipping cost per unit is identified between the

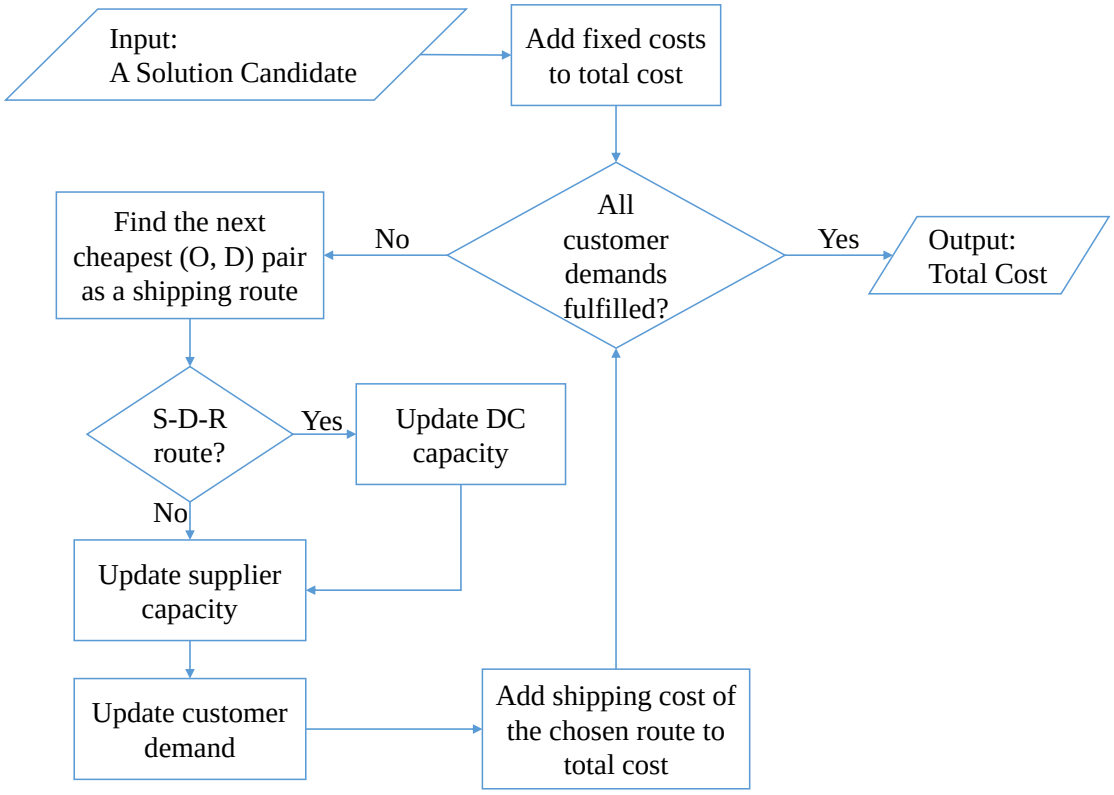



Figure 2.2: Flow Chart of the Greedy Solver

origin (i.e., supplier) and the destination (i.e., retailer). The greedy solver assigns as much retailer demand on the selected route as capacity allows. There are two route types: the Supplier-DC-Retailer (S-D-R) route and the Supplier-Retailer (S-R) route. Once a shipping task is assigned, the solver updates the capacity of the supplier on the route. If it is a S-D-R route, the solver also adjusts the DC capacity. Meanwhile, the demand of the served retailer needs an update. Now, we add the cost of this shipping assignment to the total cost. The solver repeats this process until all retailer demands are fulfilled. Figure 2.2 shows a flow chart of the greedy solver.

We tend to keep the solution with the lowest cost seen so far and use it as the seed solution for the next iteration. However, as featured in SA, we also have the option of keeping a less adequate solution as a seed solution in case of a special event: if a generated random value is less than $e^{\frac{\delta}{T}}$, in which $\delta$ is the difference between the objective value of the current solution

and that of the best solution in the current iteration. Keeping a less adequate solution helps the algorithm jump out of the local optima and search other areas of the solution space. At the end of one iteration, the temperature $T$ decreases by a rate of $r$. The entire algorithm terminates when the temperature hits a pre-defined minimum temperature $T'$. We then pick the top candidate in *rst-list*, and solve it via CPLEX to obtain the optimal solution as the outcome of the SA procedure.

We describe the pseudo code of the SA algorithm in Algorithm 1 and 2. Note that Algorithm 1 is the local search, which is a subroutine of both SA and GRASP.

**Algorithm 2** Pseudo-code of the Simulated Annealing Heuristic

**Input:** $T$, $T'$, $r$, $\alpha$, $p$, $m$, $\{f_j \mid j \in \mathcal{J}\}$, one-swap, two-swap

$\hat{J} \leftarrow \emptyset$, *checked-list* $\leftarrow \emptyset$, *rst-list* $\leftarrow \emptyset$
Sort $J$ in terms of $f_j$ in increasing order.
Pick the first $\lceil \alpha p \rceil$ DCs from $J$, add them to $\hat{J}$.
Pick the rest DCs randomly, and add them to $\hat{J}$.
$obj_c \leftarrow$ GreedySolver($\hat{J}$).
**while** $T > T'$ **do**
  *neighbor-list* $\leftarrow \emptyset$, *selected-list*$\leftarrow \emptyset$
  *selected-list* $\leftarrow$ NeighborSearch(one-swap, two-swap, *checked-list*, *neighbor-list*, *selected-list*, $m$, $J$, $\hat{J}$)
  **for** each instance $\hat{J}'$ in *selected-list* **do**
    $obj_{temp} \leftarrow$ GreedySolver($\hat{J}'$)
    add $obj_{temp}$ to *rst-list*
    **if** $obj_c > obj_{temp}$ **then**
      $obj_c \leftarrow obj_{temp}$
    **else**
      $\delta \leftarrow obj_{temp} - obj_c$
      **if** unifRand()$< e^{\delta/T}$ **then**
        $obj_c \leftarrow obj_{temp}$
        $\hat{J} \leftarrow \hat{J}'$
      **end if**
    **end if**
  **end for**
  $T \leftarrow T * r$
**end while**
Output CPLEX.solve(top candidate in rst-list)

**Algorithm 1** Pseudo-code of NeighborSearch

**Input:** one-swap, two-swap, *checked-list*, *neighbor-list*, *selected-list*, $m$, $J$, $\hat{J}$

**if** one-swap **then**
  **for** each DC $k \in \hat{J}$ where $z_k = 1$ **do**
    **for** each DC $h \in J - \hat{J}$ where $z_h = 0$ **do**
      $z_k \leftarrow 0$, $z^h \leftarrow 1$
      $\hat{J}' \leftarrow oneswap(k, h, \hat{J}, J - \hat{J})$
      **if** $\hat{J}' \in$ *checked-list* or $\hat{J}'$ fails the capacity checking **then**
        Continue
      **else**
        Add $\hat{J}'$ to *checked-list*
      **end if**
      Calculate the total fixed cost for $\hat{J}'$, denoted by $F(\hat{J}') = \sum_{j \in \hat{J}'} f_j$
      Add $\hat{J}'$ to *neighbor-list*
    **end for**
  **end for**
  Sort *neighbor-list* in terms of $F(\hat{J}')$ in increasing order
  Pick the first $m$ neighbors, and add them in *selected-list*
**end if**
**if** two-swap **then**
  **for** each DC pair $k_1, k_2 \in \hat{J}$ where $z_{k1} = z_{k2} = 1$ **do**
    **for** each DC pair $h_1, h_2 \in J - \hat{J}$ where $z_{h1} = z_{h2} = 0$ **do**
      $z_{k1} \leftarrow 0$, $z_{k2} \leftarrow 0$
      $z_{h1} \leftarrow 1$, $z_{h2} \leftarrow 1$
      $\hat{J}' \leftarrow twoswap(k_1, k_2, h_1, h_2, \hat{J}, J - \hat{J})$
      **if** $\hat{J}' \in$ *checked-list* or $\hat{J}'$ fails the capacity checking **then**
        Continue.
      **else**
        add $\hat{J}$' to *checked-list*
      **end if**
      Calculate the total fixed cost for $\hat{J}'$, denoted by $F(\hat{J}') = \sum_{j \in \hat{J}'} f_j$
      Add $\hat{J}'$ to *neighbor-list*
    **end for**
  **end for**
  Sort *neighbor-list* in terms of $F(\hat{J}')$ in increasing order
  Pick the first $m$ neighbors, and add them into *selected-list*
**end if**
Output *selected-list*

## 2.5. GRASP HEURISTICS

GRASP is an iterative process. Each iteration starts with a random seed solution for local search, which explores the neighborhood of the seed solution and only keeps the best candidate for each iteration. Standard greedy algorithms for the p-media problem (Cornuiaejols, Fisher, and Nemhauser, 1977; Whitaker, 1983) start with a clear solution and adds one DC at a time. The chosen DC is the most profitable one among the remaining DCs, i.e., the one causing the greatest drop in cost. A seed solution is ready when $p$ DCs are chosen. This greedy strategy applies to uncapacitated p-median problems quite well. However, it is not suitable for capacitated facility location problems like the PDSD problem, because opening one DC at a time may lead to an infeasible solution in the first few iterations due to insufficient capacities. To overcome this issue, we could adopt the inverse greedy strategy, which opens all DCs in the beginning, and then closes one at a time based on the amount of cost reduction, until only $p$ DCs remain open. However, both the greedy and inverse greedy algorithms are time-consuming due to the cost calculation of closing (or opening) one DC, which requires solving a MILP problem numerous times.

We design an efficient GRASP algorithm for the PDSD problem. In particular, we create a seed solution by randomly choosing a subset of DCs, sorting them in terms of the fixed cost, and only keeping the first $p$ DCs. Moreover, our GRASP algorithm employs the greedy solver described in Figure 2.2 to avoid the computation of solving the MILP problems. The details of this procedure is described as follows.

### 2.5.1. Step 0: Randomized Greedy Selection

Our GRASP procedure runs for a pre-set number of iterations, denoted by *max_iter*. Each iteration kicks off by randomly choosing a pool of DCs denoted by $DC_{init}$ with a size of $A$. To ensure that there are enough DCs to form a seed solution, we let $A = (|J| + p)/2 > p$. We then sort $DC_{init}$ in increasing order of the fixed cost and choose the first $p$ DCs as the initial seed solution. This randomized greedy procedure not only generates a good starting point, but also enables parallel computation on a multi-processor system, because the algorithm can

produce multiple initial solutions that can be handled by a multiple CPU cores in parallel. We solve the initial instance with the greedy solver as shown in Figure 2.2 and store the objective value.

### 2.5.2. Step 1: Local Search

We apply both one-swap and two-swap strategies to search the neighborhood of the seed solution and obtain a list of "good" solution candidates, i.e., *selected-list*. Overall, both GRASP and SA employ the same local search procedure.

### 2.5.3. Step 2: Solution Evaluation and Update

For each instance in *selected-list*, we solve it via the greedy solver, and store the current objective value as $obj_c$. If $obj_c$ is less than the best solution $obj_{min}$ seen so far, we overwrite $obj_{min}$ with $obj_c$.

At the end of each iteration, we clear the two short memory lists *neighbor-list* and *selected-list* for the next iteration. When all *max_iter* iterations complete, we simply pass the best candidate, i.e., the one corresponding to $obj_{min}$, to CPLEX to obtain the final solution, which is the output of GRASP.

We provide the Pseudo-code of GRASP in Figure 3.

## 2.6. COMPUTATIONAL RESULTS

To test the performance of the proposed heuristics, we conduct a series of experiments on randomly generated problems with various problem scales. We construct 13 data sets with parameter settings shown in Table 2.2. Furthermore, we randomly generate the demand for each retailer ($D_{ij}$) with uniform distribution $U[90, 100]$, and the transportation cost between suppliers to DCs ($t_{kjl}$) and DCs to retailers ($c_{jil}$) with uniform distribution $U[\$1, \$10]$. Also, we assume that the fixed cost of DC ($f_j$) is a constant multiple of the capacity of DC $j$, i.e, $f_j = M * W_j$, where M is a constant. Lastly, we generate the capacities of suppliers from a uniform distribution $U[0.8\bar{P}, 1.2\bar{P}]$ where $\bar{P}$ is set to $\sum_i D_i/K$; also, the capacity of DCs is generated from a uniform distribution $U[0.8\bar{W}, 1.2\bar{W}]$ where $\bar{W}$ is equal to $\sum_i D_i/p$.

**Algorithm 3** Pseudo-code of GRASP

**Input:** *max_iter*, $p$, $J$, $A$, $m$, $\{f_j \mid j \in \mathcal{J}\}$, one-swap, two-swap

$\hat{J} \leftarrow \emptyset$
*checked-list* $\leftarrow \emptyset$
$A \leftarrow (|J| + p)/2$
**for** n $\leftarrow$ 1 to *max_iter* **do**
  randomly select $A$ DCs from $J$, denoted by set $DC_{init}$
  sort $DC_{init}$ in terms of $f_j$ in increasing order
  choose the first p DCs from $DC_{init}$, denoted by set $\hat{J}$
  $obj_c \leftarrow$ GreedySolver $(\hat{J})$, $obj_{min} \leftarrow obj_c$
  *neighbor-list*$\leftarrow \emptyset$, *selected-list*$\leftarrow \emptyset$.
  *selected-list* $\leftarrow$ NeighborSearch(one-swap, two-swap, *checked-list*, *neighbor-list*, *selected-list*, $m$, $J$, $\hat{J}$)
  **for** each solution $\hat{J}'$ in *selected-list* **do**
    $obj_c \leftarrow$ GreedySolver($\hat{J}'$)
    **if** $obj_c < obj_{min}$ **then**
      $obj_{min} \leftarrow obj_c$
      $\hat{J}'_{min} \leftarrow \hat{J}'$
    **end if**
  **end for**
**end for**
$obj_{min} \leftarrow$ CPLEX( $\hat{J}'_{min}$)
Output ($\hat{J}'_{min}$, $obj_{min}$)

Table 2.2: Data Setting

| Data Set | # Suppliers $K$ | # DCs $N$ | # Retailers $M$ | # Products $L$ | # Opened DCs $p$ |
|---|---|---|---|---|---|
| 1 | 10 | 50 | 50 | 5 | 10 |
| 2 | 10 | 150 | 50 | 5 | 10 |
| 3 | 10 | 150 | 50 | 5 | 50 |
| 4 | 10 | 50 | 100 | 5 | 10 |
| 5 | 10 | 150 | 100 | 5 | 10 |
| 6 | 10 | 150 | 100 | 5 | 50 |
| 7 | 40 | 50 | 50 | 5 | 10 |
| 8 | 40 | 150 | 50 | 5 | 10 |
| 9 | 40 | 150 | 50 | 5 | 50 |
| 10 | 40 | 50 | 100 | 5 | 10 |
| 11 | 40 | 150 | 100 | 5 | 10 |
| 12 | 40 | 150 | 100 | 5 | 50 |
| 13 | 10 | 50 | 50 | 50 | 10 |

### 2.6.1. Solving the PDSD Problem with SA and GRASP

We run CPLEX, SA, and GRASP to solve a particular instance chosen from data set 1, and plot the results in Figure 2.3. For SA and GRASP, we highlight three points, including the initial solution, the best solution obtained by the greedy solver when all iterations are finished, and the final solution obtained by CPLEX at the end. We also plot the optimal objective value obtained by CPLEX. For this particular instance, CPLEX takes about 18 seconds to find the optimal solution, while the running times for SA and GRASP are 15.2 seconds and 8.37 seconds, respectively. As for solution quality, both SA and GRASP start with an inadequate solution, which is greatly improved after the solver applies the heuristics. In particular, SA takes about 12 seconds to hit the pre-set temperature limit with the best objective value of \$71098, while GRASP finishes the iterations with only 4.3 seconds and finds the best objective value of \$70421. Note that so far both SA and GRASP use the greedy solver to evaluate the problem. Clearly, SA takes a longer time to complete the iterations. Now, they pass the best candidate seen so far to CPLEX to obtain the final solution, and this step takes about 4 seconds. Interestingly, although SA finds a solution slightly worse than GRASP (\$71098 Vs. \$70421, evaluated by the greedy solver), the solution ends up being better (\$66418 Vs. \$67142), after it is evaluated by CPLEX. This means that greedy solver is not as optimal as the CPLEX solver, but it still offers a clear direction during the iteration and saves huge running time.

### 2.6.2. Overall Performance of SA and GRASP

In Tables 2.3 and 2.4, we present an overall comparison among CPLEX, SA, and GRASP in terms of solution quality and running time. The percentage of optimality gaps is defined as [100∗(Heuristic Obj. Value − Optimal Obj. Value)/Optimal Obj. Value], and the running time is reported in seconds. We run all experiments in Microsoft Windows 2003 Enterprise x64 Edition with a CPU of Dual-Core AMD Opteron 3.00 GHz, and a RAM of 64 GBs.

For small size instances, e.g., set 1, 4, and 7, CPLEX is quite fast; for large problems (set 5, 6, 11-13), however, CPLEX is far from satisfying. Compared with GRASP, SA presents a

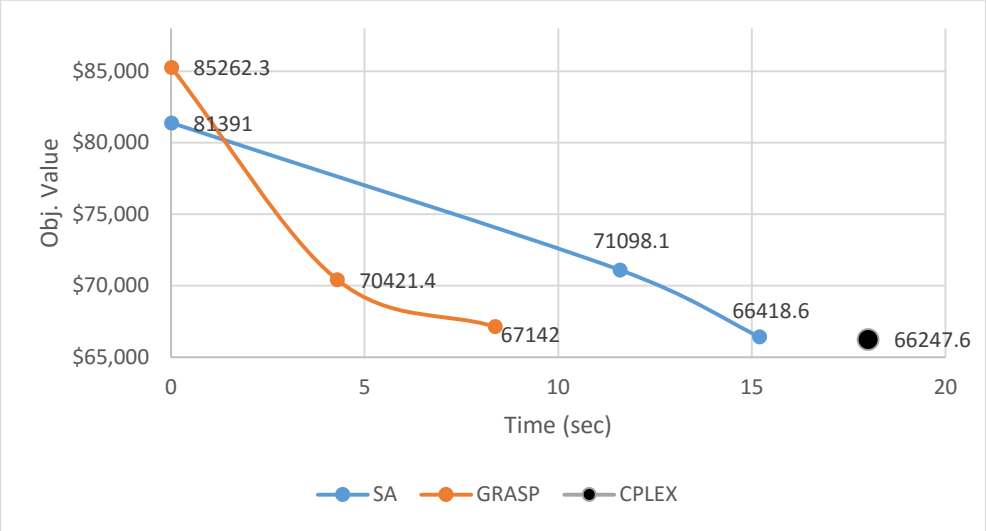



Figure 2.3: A Performance Comparison of SA, GRASP, and CPLEX on a Particular Instance

Table 2.3: Summary of Solution Quality

| Data Set | Optimal | SA | Gap | GRASP | Gap |
|---|---|---|---|---|---|
| 1 | $64,746 | $64,833 | 0.13% | $65,015 | 0.41% |
| 2 | $59,583 | $59,671 | 0.15% | $59,869 | 0.48% |
| 3 | $68,576 | $68,664 | 0.13% | $68,930 | 0.52% |
| 4 | $130,978 | $131,400 | 0.32% | $131,430 | 0.34% |
| 5 | $118,966 | $119,209 | 0.20% | $119,802 | 0.70% |
| 6 | $137,880 | $137,989 | 0.08% | $138,262 | 0.27% |
| 7 | $63,474 | $63,528 | 0.08% | $63,548 | 0.11% |
| 8 | $59,065 | $59,121 | 0.10% | $59,202 | 0.23% |
| 9 | $66,937 | $66,965 | 0.04% | $67,191 | 0.37% |
| 10 | $127,334 | $127,382 | 0.04% | $127,382 | 0.04% |
| 11 | $115,797 | $115,851 | 0.05% | $116,069 | 0.23% |
| 12 | $134,095 | $134,123 | 0.02% | $134,924 | 0.62% |
| 13 | $655,057 | $655,881 | 0.12% | $656,403 | 0.20% |
| Average | $138,653 | $138,817 | 0.11% | $139,079 | 0.35% |

Table 2.4: Summary of Running Time Improvement

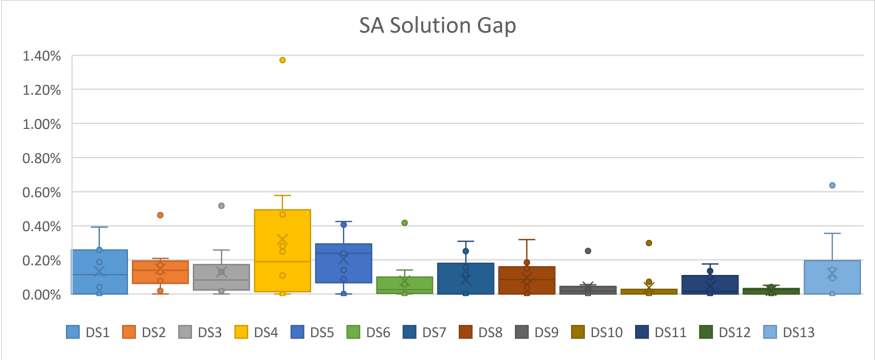
| Data Set | Optimal | SA | Improvement | GRASP | Improvement |
|---|---|---|---|---|---|
| 1 | 18.1 | 15.7 | 13.11% | 8.2 | 54.47% |
| 2 | 382.3 | 317.7 | 16.72% | 136.6 | 64.18% |
| 3 | 373.9 | 414.5 | -10.86% | 410.8 | -9.86% |
| 4 | 46.1 | 39.5 | 14.33% | 23.2 | 49.60% |
| 5 | 734.8 | 600.1 | 18.28% | 279.1 | 62.00% |
| 6 | 725.8 | 706.8 | 2.61% | 574.7 | 20.81% |
| 7 | 33.5 | 33.5 | -0.05% | 20.3 | 39.40% |
| 8 | 511.9 | 444.0 | 13.27% | 212.2 | 58.55% |
| 9 | 517.5 | 546.1 | -5.59% | 506.6 | 2.04% |
| 10 | 86.9 | 89.7 | -3.32% | 62.9 | 27.54% |
| 11 | 984.2 | 857.0 | 12.93% | 425.5 | 56.76% |
| 12 | 991.2 | 999.5 | -0.90% | 856.3 | 13.57% |
| 13 | 1624.4 | 1417.8 | 12.63% | 940.7 | 42.03% |
| Average | 540.8 | 498.6 | 6.40% | 342.9 | 37.01% |

smaller optimality gap, with an average gap of 0.11%, and an average time improvement of 6.4%. On the other hand, GRASP is 37.01% faster than CPLEX with an average optimality gap of 0.35%.

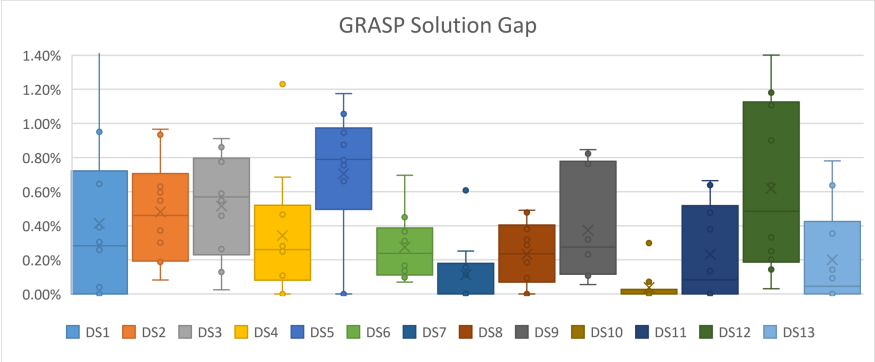
We also provide the box and whisker charts of SA and GRASP in Figure 2.4 to demonstrate the variation of solution and time gaps for each tested data set. We observe that GRASP presents a higher variation of solution gap than SA. Also, both SA and GRASP present a low variation in time gap, but GRASP outperforms SA to a great extent.

### 2.6.3. Parameter Analysis

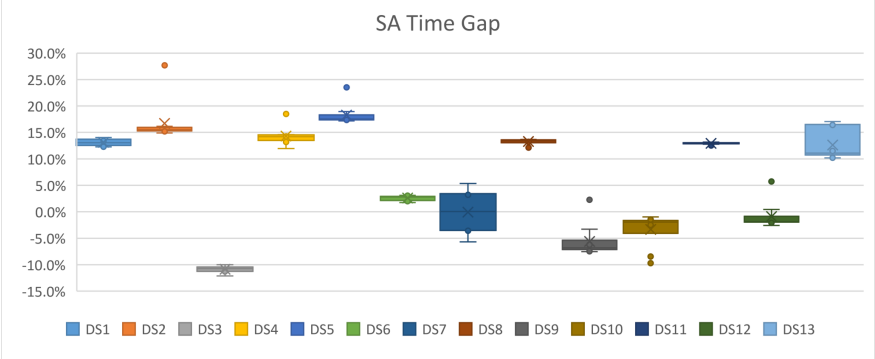
In this section, we discuss the impact of parameters on the total cost and shipping options. In the general model, we consider two shipping options to deliver products to retailers: (i) regular shipping, i.e., shipments from a DC to retailers ($x_{jil}$) and (ii) direct shipping, i.e., shipments from a supplier to retailers ($u_{kil}$). In general, given a certain demand, the number of regular shipping is mainly decided by shipping costs and distances. In the experiments, we assume that the unit mileage shipping cost of direct shipping is higher than the cost of regular shipping, i.e., for any retailer $i$ and its product $l$, $s_{kil} > r_{kjl} + c_{jil}$. Under this



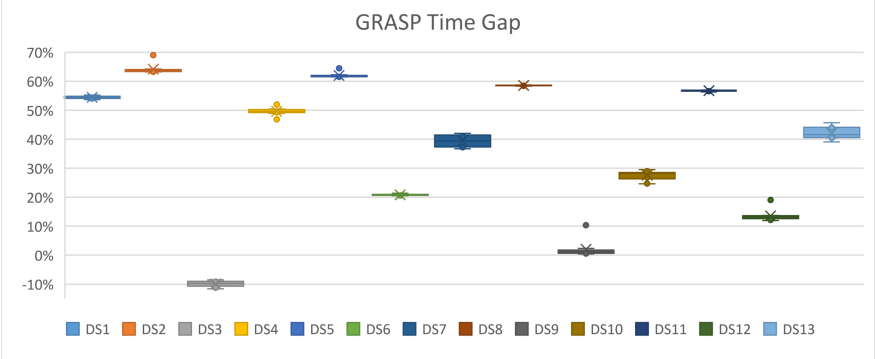

(a)




(b)




(c)




(d)

Figure 2.4: Box and Whisker Charts of SA and GRASP in Comparison to CPLEX

assumption, for one replication/instance, the total number of $x_{jil}$ is greater than $u_{kil}$. In addition to mileage cost, the number of DCs also has an impact on shipping options. Figure 2.5 displays the average total cost for the 13 scenarios with the following factors set at low and high levels: number of supply centers K (10 vs. 40), number of retailers $I$ (50 vs. 100), number of opened DCs $p$ (10 vs. 50), and number of DCs (50 vs. 100). Using the data setting in Table 2.2 and the average total cost of each scenario, we ran a regression model with the five parameters $(K, N, M, L, p)$ to estimate the total cost, and obtained a regression model that achieved an adjusted R-square of 0.998. Although the coefficients of regression can vary with different cost settings, it provides a convenient approach to estimate the total cost with any changes of the numbers of products and DCs.

$$\text{Total Cost} = -77.9K - 175.2N + 1275.6M + 12579.1L + 453.4p - 51140.7$$

Examining the results in Figure 2.5, we observe that the average total cost decreases as the numbers of suppliers and DCs increase. This outcome results from a decision maker being able to better utilize the increased resource capacity by choosing a closer facility to finish shipping assignments. Thus, as a company has more resources, its operational "efficiency" improves. Another observation is about the impact of retailers on the total cost. Intuitively, as the number of retailers increases, the total cost also increases. When the number of open DCs increases to 50, the total cost increases as well. Having more open DCs, the model has more options to assign the shipment, leading to a reduced transportation cost. However, the fixed cost increases in a faster way, therefore, the total cost rises when more DCs are opened.

The number of DCs not only affects the total cost, but also relates to different types of shipments. In the PDSD model, we consider four types of shipping flow: shipments from a supply center to a DC, shipments from a DC to a retailer, shipments from a supply center to a retailer, and transshipments between two DCs. In the following subsections, we will investigate how the parameters $p$, $K$, and $N$ impact shipping decisions. In Table sets 2.5 and 2.6, we display the shipping assignments for DS 1 - DS 11. For each data set, we list

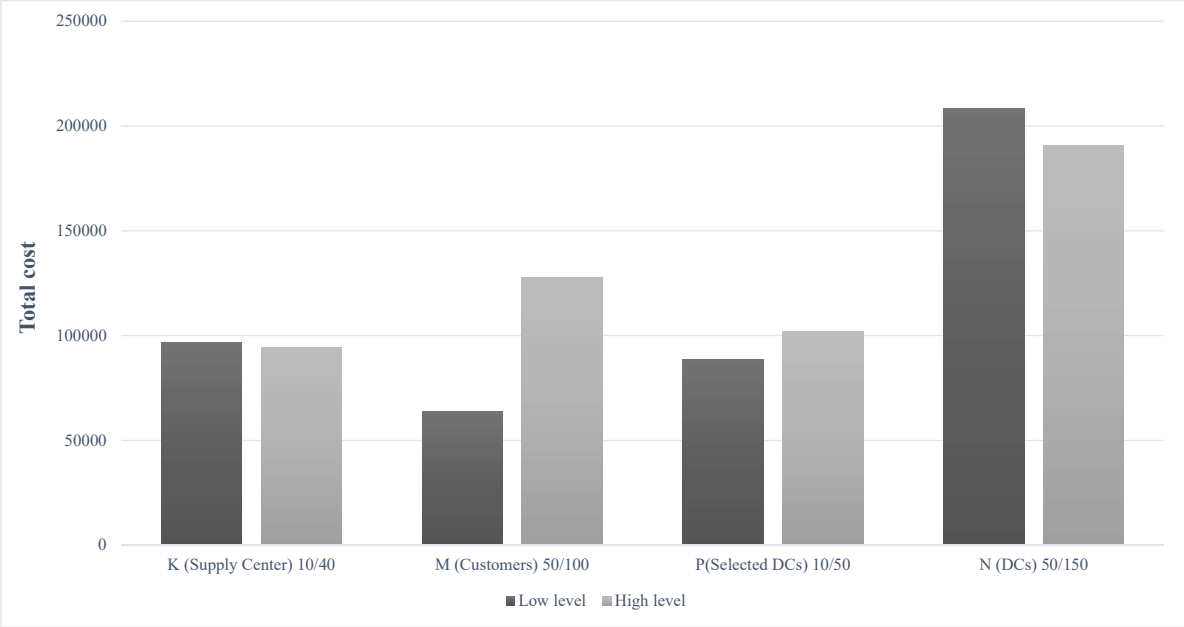



Figure 2.5: Effect of Parameters Settings on Total Cost

the quantity for each shipping type in first four columns, and the ratio of direct shipping quantity to regular shipping quantity in the last column.

**Number of Open DCs:** $p$

We perform a t-test regarding the quantity of regular shipping $x_{jil}$ with two levels of $p$: $p = 10$ and $p = 50$, as given in the data setting. We pick three pairs of data sets including (DS 2, DS 3), (DS 5, DS 6), and (DS 8, DS 9), and ensures that the two data sets in each pair only differ in $p$, and that the first data set takes the low level, i.e., $p = 10$. Since each data set contains 10 instances, we collect a total of 30 pairs of $x_{jil}$ values for a t-test. Let $\mu(x_{jil} \mid p = 10)$ denote the mean of $x_{jil}$ given $p = 10$. When choosing a significant level $\alpha = 0.05$, we can make a null hypothesis $H_0$ and an alternative hypothesis $H_1$:

$$H_0 : \quad \mu(x_{jil} \mid p = 10) \geq \mu(x_{jil} \mid p = 50)$$

$$H_1 : \quad \mu(x_{jil} \mid p = 10) < \mu(x_{jil} \mid p = 50)$$

Table 2.5: Shipping Assignment for DS 1 - DS6

| | DS 1 | | | | | DS 2 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| ID | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x |
| 1 | 3189 | 3189 | 328 | 9 | 0.10 | 2928 | 2928 | 501 | 0 | 0.17 |
| 2 | 3118 | 3118 | 343 | 0 | 0.11 | 2914 | 2914 | 560 | 0 | 0.19 |
| 3 | 3123 | 3123 | 329 | 0 | 0.11 | 3024 | 3024 | 517 | 0 | 0.17 |
| 4 | 3059 | 3059 | 423 | 0 | 0.14 | 3010 | 3010 | 524 | 0 | 0.17 |
| 5 | 3239 | 3239 | 251 | 0 | 0.08 | 2966 | 2966 | 428 | 0 | 0.14 |
| 6 | 3212 | 3212 | 343 | 18 | 0.11 | 3010 | 3010 | 493 | 0 | 0.16 |
| 7 | 3139 | 3139 | 423 | 30 | 0.13 | 3075 | 3075 | 475 | 0 | 0.15 |
| 8 | 3261 | 3261 | 231 | 36 | 0.07 | 3072 | 3072 | 414 | 28 | 0.13 |
| 9 | 3011 | 3011 | 457 | 0 | 0.15 | 2953 | 2953 | 566 | 0 | 0.19 |
| 10 | 3272 | 3272 | 213 | 0 | 0.07 | 3085 | 3085 | 395 | 0 | 0.13 |
| Avg. | 3162 | 3162 | 334 | 9 | 0.11 | 3004 | 3004 | 487 | 3 | 0.16 |
| | DS 3 | | | | | DS 4 | | | | |
| ID | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x |
| 1 | 3218 | 3218 | 239 | 0 | 0.07 | 6632 | 6632 | 408 | 92 | 0.06 |
| 2 | 3224 | 3224 | 291 | 0 | 0.09 | 6320 | 6320 | 634 | 87 | 0.10 |
| 3 | 3321 | 3321 | 185 | 0 | 0.06 | 6063 | 6063 | 884 | 0 | 0.15 |
| 4 | 3284 | 3284 | 221 | 0 | 0.07 | 6350 | 6350 | 703 | 44 | 0.11 |
| 5 | 3256 | 3256 | 218 | 0 | 0.07 | 6114 | 6114 | 965 | 0 | 0.16 |
| 6 | 3317 | 3317 | 242 | 0 | 0.07 | 6625 | 6625 | 338 | 0 | 0.05 |
| 7 | 3176 | 3176 | 337 | 0 | 0.11 | 6167 | 6167 | 839 | 0 | 0.14 |
| 8 | 3253 | 3253 | 331 | 0 | 0.10 | 6285 | 6285 | 660 | 18 | 0.11 |
| 9 | 3220 | 3220 | 264 | 0 | 0.08 | 6405 | 6405 | 629 | 0 | 0.10 |
| 10 | 3248 | 3248 | 171 | 0 | 0.05 | 6490 | 6490 | 524 | 0 | 0.08 |
| Avg. | 3252 | 3252 | 250 | 0 | 0.08 | 6345 | 6345 | 658 | 24 | 0.10 |
| | DS 5 | | | | | DS 6 | | | | |
| ID | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x |
| 1 | 6002 | 6002 | 997 | 0 | 0.17 | 6787 | 6787 | 231 | 0 | 0.03 |
| 2 | 5941 | 5941 | 1053 | 0 | 0.18 | 6560 | 6560 | 431 | 0 | 0.07 |
| 3 | 6074 | 6074 | 970 | 14 | 0.16 | 5700 | 5700 | 248 | 0 | 0.04 |
| 4 | 5914 | 5914 | 1020 | 0 | 0.17 | 5611 | 5611 | 398 | 0 | 0.07 |
| 5 | 5872 | 5872 | 1119 | 0 | 0.19 | 6480 | 6480 | 500 | 0 | 0.08 |
| 6 | 6224 | 6224 | 799 | 0 | 0.13 | 6318 | 6318 | 630 | 0 | 0.10 |
| 7 | 6047 | 6047 | 916 | 0 | 0.15 | 6634 | 6634 | 407 | 0 | 0.06 |
| 8 | 6059 | 6059 | 890 | 0 | 0.15 | 6377 | 6377 | 635 | 0 | 0.10 |
| 9 | 5987 | 5987 | 1080 | 0 | 0.18 | 6403 | 6403 | 593 | 0 | 0.09 |
| 10 | 5909 | 5909 | 1071 | 15 | 0.18 | 6580 | 6580 | 396 | 0 | 0.06 |
| Avg. | 6003 | 6003 | 992 | 3 | 0.17 | 6345 | 6345 | 447 | 0 | 0.07 |

Table 2.6: Shipping Assignment for DS 7 - DS 11

| | DS 7 | | | | | DS 8 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| ID | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x |
| 1 | 3064 | 3064 | 408 | 0 | 0.12 | 3051 | 3051 | 519 | 0 | 0.17 |
| 2 | 3189 | 3189 | 290 | 0 | 0.08 | 3001 | 3001 | 554 | 0 | 0.18 |
| 3 | 2994 | 2994 | 506 | 0 | 0.14 | 2933 | 2933 | 576 | 0 | 0.20 |
| 4 | 3265 | 3265 | 223 | 0 | 0.06 | 3052 | 3052 | 459 | 0 | 0.15 |
| 5 | 3087 | 3087 | 366 | 0 | 0.11 | 3016 | 3016 | 522 | 0 | 0.17 |
| 6 | 3329 | 3329 | 201 | 0 | 0.06 | 3267 | 3267 | 214 | 0 | 0.07 |
| 7 | 3241 | 3241 | 239 | 0 | 0.07 | 3002 | 3002 | 500 | 0 | 0.17 |
| 8 | 3171 | 3171 | 289 | 0 | 0.08 | 3142 | 3142 | 435 | 0 | 0.14 |
| 9 | 3126 | 3126 | 331 | 0 | 0.10 | 3040 | 3040 | 520 | 0 | 0.17 |
| 10 | 3359 | 3359 | 144 | 0 | 0.04 | 2944 | 2944 | 503 | 0 | 0.17 |
| Avg. | 3183 | 3183 | 300 | 0 | 0.09 | 3045 | 3045 | 480 | 0 | 0.16 |
| | DS 9 | | | | | DS 10 | | | | |
| ID | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x |
| 1 | 3296 | 3296 | 248 | 0 | 0.08 | 6171 | 6171 | 649 | 0 | 0.11 |
| 2 | 3272 | 3272 | 224 | 0 | 0.07 | 6876 | 6876 | 196 | 0 | 0.03 |
| 3 | 3156 | 3156 | 386 | 0 | 0.12 | 6283 | 6283 | 685 | 0 | 0.11 |
| 4 | 3212 | 3212 | 298 | 0 | 0.09 | 6395 | 6395 | 546 | 0 | 0.09 |
| 5 | 3240 | 3240 | 221 | 0 | 0.07 | 6487 | 6487 | 560 | 0 | 0.09 |
| 6 | 3215 | 3215 | 238 | 0 | 0.07 | 6117 | 6117 | 871 | 0 | 0.14 |
| 7 | 3345 | 3345 | 155 | 0 | 0.05 | 6338 | 6338 | 636 | 0 | 0.10 |
| 8 | 3390 | 3390 | 186 | 0 | 0.05 | 6330 | 6330 | 674 | 0 | 0.11 |
| 9 | 3232 | 3232 | 259 | 0 | 0.08 | 6298 | 6298 | 624 | 0 | 0.10 |
| 10 | 3299 | 3299 | 202 | 0 | 0.06 | 6514 | 6514 | 440 | 0 | 0.07 |
| Avg. | 3266 | 3266 | 242 | 0 | 0.07 | 6381 | 6381 | 588 | 0 | 0.09 |
| | DS 11 | | | | | | | | | |
| ID | $x_{jil}$ | $y_{kjl}$ | $u_{kil}$ | $v_{hjl}$ | u/x | | | | | |
| 1 | 6171 | 6046 | 953 | 0 | 0.16 | | | | | |
| 2 | 6876 | 5936 | 1015 | 0 | 0.17 | | | | | |
| 3 | 6283 | 5979 | 1088 | 0 | 0.18 | | | | | |
| 4 | 6395 | 6149 | 888 | 0 | 0.14 | | | | | |
| 5 | 6487 | 6319 | 597 | 0 | 0.09 | | | | | |
| 6 | 6117 | 6010 | 953 | 0 | 0.16 | | | | | |
| 7 | 6338 | 5980 | 1029 | 0 | 0.17 | | | | | |
| 8 | 6330 | 6081 | 915 | 0 | 0.15 | | | | | |
| 9 | 6298 | 5946 | 1004 | 0 | 0.17 | | | | | |
| 10 | 6514 | 5812 | 1144 | 0 | 0.20 | | | | | |
| Avg. | 6381 | 6026 | 959 | 0 | 0.16 | | | | | |

Table 2.7 shows the result, in which p-value of the one-tail test is far less than $\alpha$, indicating an invalid hypothesis and a valid alternative. In other words, we reject the null hypothesis, and conclude that $p$ does positively impact the number of regular shipping. Also, since the sum of the quantities of regular and direct shipping is fixed, an increased amount of regular shipping indicates a reduced amount of direct shipping. Thus, $p$ is negatively related to the quantity of direct shipping. To explain this phenomenon, we let route A and route B represents direct shipping and regular shipping, respectively. When routes A and B have the same distance, route B is preferred because of the assumption $s_{kil} > r_{kjl} + c_{jil}$. Increasing the number of opened DCs offers more opportunity to choose a shorter route.

Table 2.7: A t-test on $x_{jil}$ with Two Levels of $p$

| Regular Shipping $x_{jil}$ | $p = 10$ | $p = 50$ |
|---|---|---|
| Mean | 4017.13 | 4287.47 |
| Variance | 2047422.40 | 2238722.81 |
| Observations | 30 | 30 |
| Pearson Correlation | 0.97 | |
| Hypothesized Mean Difference | 0 | |
| df | 29 | |
| $t$ Stat. | -6.02 | |
| $P(T \leq t)$ one-tail | 7.47317E-07 | |
| $t$ Critical one-tail | 1.699127027 | |
| $P(T \leq t)$ two-tail | 1.49463E-06 | |
| $t$ Critical two-tail | 2.045229642 | |

**Number of Supply Centers:** $K$

For this experiment, we pick five pairs of data sets: (DS 1, DS 7), (DS 2, DS 8), (DS 3, DS 9), (DS 4, DS 10), and (DS 5, DS 11), and collect a total of 50 pairs of $x_{jil}$ values for the test. Again, the first data set takes the low level, compared to the second data set in the pair, i.e., $K = 10$ Vs. $K = 40$.

The first observation is that transshipments appear in DS 1, DS 2, DS 4, and DS 5, i.e., $v_{hjl} > 0$, while there is no transshipment in any of the data set with a high level of $K$. This is rational, because increased suppliers might bring more cost-efficient routes (including A and B) into the scenario, making transshipment less attractive.

Similar to the previous subsection, we perform a t-test on $x_{jil}$ with two levels of $K$. With the same significant level $\alpha = 0.05$, we make a null hypothesis $H_0$ and an alternative hypothesis $H_1$ as follows:

$$H_0 : \quad \mu(x_{jil} \mid K = 10) \geq \mu(x_{jil} \mid K = 40)$$
$$H_1 : \quad \mu(x_{jil} \mid K = 10) < \mu(x_{jil} \mid K = 40)$$

Table 2.8 shows the result, in which the p-value of the one-tail test is far less than $\alpha$, indicating an invalid null hypothesis. Therefore, the alternative hypothesis holds. In other words, we reject the null hypothesis, and conclude that $K$ does positively impact the number of regular shipping, and negatively affect the number of direct shipping. We can also explain this: consider a given retailer and its products, when route A (i.e., route S-R) and route B (i.e., route S-D-R) have the same travel distance, transportation cost of route B is lower than that of route A. When there are 10 supply centers, finding a route having low mileage cost and short distance is quite a challenge, so the solver may choose direct shipping. However, when the number of supply centers increases to 40, the solver has more options of route B. We provide a minimal example in Figure 2.6, which describes a network of two supply centers, three DCs, and one retailer. When the network only has one supply center and two DCs, we have one option of Route A ($A_1$) and two options of Route B ($B_1$ and $B_2$) to serve retailer $i_1$. As we add one more supply center, the number of potential options of Route A increases by one ($A_2$), while the number of Route B increases by two (($B_3$ and $B_4$). Following this logic, with 30 more supply centers, a retailer has 30 more direct shipping choices (i.e., Route A), and 300 (30*10) more choices for Route B. Therefore, there is a higher chance that the solver finds a Route B better than a Route A.

**Number of Potential DCs:** $N$

For this experiment, we pick four pairs of data sets: (DS 1, DS 2), (DS 4, DS 5), (DS 7, DS 8), and (DS 10, DS 11), and gather a total of 40 pairs of $x_{jil}$ values. Again, the first

Table 2.8: A t-test on $x_{jil}$ with Two Levels of $K$

| Regular Shipping $x_{jil}$ | $K = 10$ | $K = 40$ |
|---|---|---|
| Mean | 4353.14 | 4450.96 |
| Variance | 2285546.60 | 2560744.61 |
| Observations | 50 | 50 |
| Pearson Correlation | 0.99 | |
| Hypothesized Mean Difference | 0 | |
| df | 49 | |
| $t$ Stat | -2.74 | |
| $P(T \leq t)$ one-tail | 0.004255535 | |
| $t$ Critical one-tail | 1.676550893 | |
| $P(T \leq t)$ two-tail | 0.008511071 | |
| $t$ Critical two-tail | 2.009575237 | |

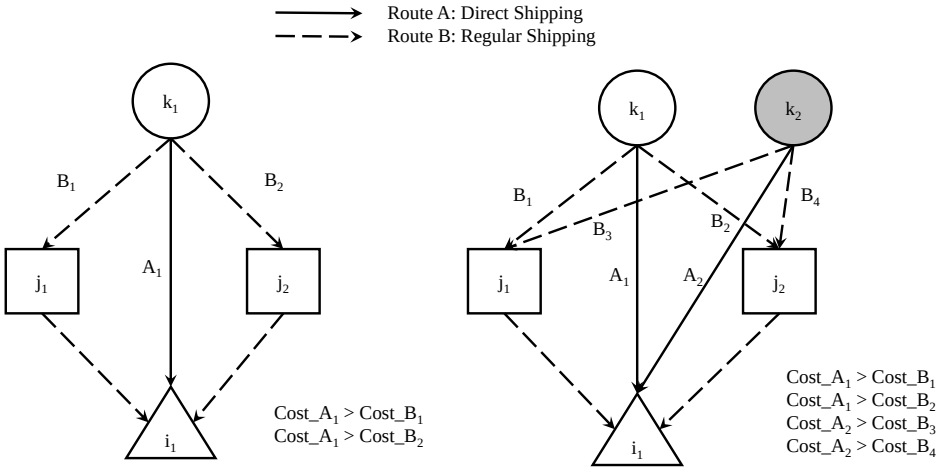



Figure 2.6: Effect of the Number of Supply Centers on Shipping Options

data set takes the low level, compared to the second data set in the pair, i.e., $N = 50$ Vs. $N = 150$. We show the t-test result in 2.9. Using the same significant level $\alpha = 0.05$, we can invalidate the hypothesis: $\mu(x_{jil} \mid N = 50) \leq \mu(x_{jil} \mid N = 150)$, and conclude that $N$ poses a negative impact on regular shipping, and thus makes a positive impact on direct shipping. In other words, the quantity of direct shipping rises when there are more potential DCs. This result contradicts our intuition but can be explained. The key is the fixed cost of opening a DC. Let us assume that the cost of increasing one unit of capacity in a DC is $\$\alpha$, and that retailer $i$ needs one unit of a product. Consider the case of opening 10 DCs, when there are 50 DCs available to choose. The optimal solution for retailer $i$ could be receiving the product from DC $j$ with a capacity of $W_j$. The cost for this optimal solution is $(r_{kjl} + c_{jil})$. When there are 150 DCs available, assume that we find a new DC located at the same place as DC $j$ but with a capacity lower than $W_j - 1$. The new DC cannot fulfill retailer $i$'s order due to insufficient capacity; we then have to either use direct shipping from supply center $k$ or choose another DC:

Table 2.9: A t-test on $x_{jil}$ with Two Levels of $N$

| Regular Shipping $x_{jil}$ | $N = 50$ | $N = 150$ |
|---|---|---|
| Mean | 4767.7 | 4608.08 |
| Variance | 2634980.04 | 2607425.15 |
| Observations | 40 | 40 |
| Pearson Correlation | 0.99 | |
| Hypothesized Mean Difference | 0 | |
| df | 39 | |
| $t$ Stat | 5.87 | |
| $P(T \leq t)$ one-tail | 3.97979E-07 | |
| $t$ Critical one-tail | 1.684875122 | |
| $P(T \leq t)$ two-tail | 7.95959E-07 | |
| $t$ Critical two-tail | 2.02269092 | |

**Option 1.** Direct shipping: The new transportation cost will be $s_{kil}$, but we also save $\$\alpha$ by reducing the size of DC by one unit. Therefore, the difference of total cost is $(r_{kjl} + c_{jil}) - (-\alpha + s_{kil}) = \alpha - s_{kil} + r_{kjl} + c_{jil}$. Based on the assumption that $s_{kil} > r_{kjl} + c_{jil}$ (i.e., direct shipping costs more). Let us define the difference of direct

shipping and regular shipping $\beta = s_{kil} - r_{kjl} - c_{jil}$. If $\alpha \geq \beta$, we will choose direct shipping and open the smaller DC; if $\alpha < \beta$, we will open the large DC and use regular shipping. Based on our data setting, $\alpha > \beta$, which explains why the algorithm prefers direct shipping when there are more candidate DCs.

**Option 2.** Transshipment from another DC $j$': The logic is similar. The transportation cost for one additional product is $(r_{khl} + t_{hjl})$ instead of $s_{kil}$.

### 2.6.4. Numerical Study of The Special Case

In this section, we examine SA and GRASP algorithms on the special case discussed in section 2.3.2. In addition, we test the SS heuristic (Keskin and Üster, 2007a) as a benchmark algorithm for performance comparison. Table 2.10 presents the data setting for this experiment. For each data set, we generate 10 instances with the parameters $K$, $N$, $M$, $L$, $p$ randomly chosen from the given ranges in the table. We first solve the 8 data sets (a total of 80 instances) using CPLEX to obtain the optimal solutions, and list the average and maximum running times in the last two columns. We then run SA, GRASP, and SS on the same instances for performance comparison.

Table 2.10: Data Setting of Special Case

| Data Set | Parameters | | | | | Running Time (s) | |
|---|---|---|---|---|---|---|---|
| | $K$ | $N$ | $M$ | $L$ | $p$ | average | max |
| 1 | [2,5] | [10, 20] | [30, 40] | [2, 5] | [1, 5] | 2.64 | 10.22 |
| 2 | [2,5] | [20, 30] | [30, 40] | [2, 5] | [1, 5] | 11.67 | 48.36 |
| 3 | [2,5] | [20, 30] | [30, 40] | [2, 5] | [6, 10] | 9.35 | 35.64 |
| 4 | [2,5] | [10, 20] | [30, 40] | [6, 10] | [1, 5] | 31.70 | 158.35 |
| 5 | [2,5] | [20, 30] | [30, 40] | [6, 10] | [1, 5] | 93.90 | 382.08 |
| 6 | [2,5] | [30, 50] | [40, 100] | [2, 5] | [6, 10] | 35.69 | 168.22 |
| 7 | [2,5] | [20, 30] | [30, 40] | [6, 10] | [6, 10] | 75.07 | 318.91 |
| 8 | [2,5] | [30, 50] | [40, 100] | [6, 10] | [6, 10] | 1289.77 | 4427.36 |

**Simulated Annealing**

In Table 2.11 we summarize the performance of SA and SS regarding the average optimality gap and the average running time. Overall, SA outperforms SS in both solution quality and algorithm efficiency. We run the SA algorithm with a selection of two or three neighbors ($m = 2$ or 3) in local search. The average optimality gap of SA with two-neighbor selection (SA(2)) is 0.86%, and the gap becomes 0.64% for three-neighbor selection (SA(3)). For SA(2), seven out of eight sets have a gap less than 1%, compared to the CPLEX optimal solution. For SA(3), six out of eight reach this standard. Such a small gap in SA is a noticeable improvement from SS. Regarding running time, SA(3) is not as good as SA(2), because the algorithm selects three instances (instead of two) from *selected-list* for evaluation. Although SA(3) is slower than SA(2), it is still three times faster than SS.

Table 2.11: Overall Comparisons between SA and SS

| | Gap | | | Running Time (s) | | |
|---|---|---|---|---|---|---|
| Data Set | SA (2) | SA(3) | SS | SA (2) | SA(3) | SS |
| 1 | 0.77% | 0.20% | 3.25% | 2.05 | 3.25 | 7.25 |
| 2 | 0.98% | 0.61% | 3.06% | 6.29 | 10.65 | 29.68 |
| 3 | 0.96% | 1.07% | 3.55% | 7.13 | 13.50 | 41.20 |
| 4 | 0.26% | 0.24% | 1.52% | 11.31 | 19.40 | 50.49 |
| 5 | 0.80% | 0.35% | 2.52% | 29.54 | 59.51 | 123.72 |
| 6 | 1.48% | 1.01% | 4.56% | 29.27 | 47.67 | 141.40 |
| 7 | 0.81% | 0.82% | 2.86% | 38.57 | 82.70 | 190.34 |
| 8 | 0.79% | 0.77% | 2.03% | 551.88 | 975.96 | 3727.26 |
| Average | 0.86% | 0.64% | 2.92% | 84.63 | 151.38 | 538.92 |

Tables 2.12 and 2.13 summarize the statistics of solutions for SA(2) and SA(3) algorithms. In both tables, we list the average, median, and maximum value of optimality gap and running time. The last column shows the number of optimal solutions we obtain in each set. For SA(2), the algorithm has a stable performance on all sets. For example, the difference of average optimality gap between the smallest set (set 1) and the largest set (set 8) is 0.02% in SA(2), while in SA(3), this difference is 0.57%. We can find a similar trend on running

time. The variation of running time in SA(3) is higher than that in SA(2), especially for the large sets (sets 5 to 8).

Table 2.12: SA with Two-neighbor Search

| | Gap | | | Running Time (s) | | | # Optimal |
|---|---|---|---|---|---|---|---|
| Data Set | average | median | max | average | median | max | Solutions |
| 1 | 0.77% | 0.11% | 2.68% | 2.05 | 1.07 | 7.45 | 5 |
| 2 | 0.98% | 0.63% | 2.15% | 6.29 | 3.44 | 17.20 | 2 |
| 3 | 0.96% | 0.94% | 1.95% | 7.13 | 3.56 | 21.38 | 2 |
| 4 | 0.26% | 0.00% | 1.14% | 11.31 | 7.13 | 34.94 | 7 |
| 5 | 0.80% | 0.81% | 1.99% | 29.54 | 21.6 | 94.64 | 3 |
| 6 | 1.48% | 1.39% | 3.68% | 29.27 | 19.31 | 114.67 | 1 |
| 7 | 0.81% | 1.08% | 1.43% | 39.57 | 20.17 | 131.38 | 3 |
| 8 | 0.79% | 0.89% | 1.42% | 551.88 | 237.86 | 1944.24 | 1 |
| Average | 0.86% | 0.73% | 2.06% | 84.63 | 39.27 | 295.74 | 3 |

Table 2.13: SA with Three-neighbor Search

| | Gap | | | Running Time (s) | | | # Optimal |
|---|---|---|---|---|---|---|---|
| Data Set | average | median | max | average | median | max | Solutions |
| 1 | 0.20% | 0.00% | 1.99% | 3.25 | 1.50 | 12.13 | 9 |
| 2 | 0.61% | 0.23% | 2.89% | 10.65 | 5.95 | 35.94 | 5 |
| 3 | 1.07% | 1.09% | 2.41% | 13.50 | 6.34 | 38.24 | 1 |
| 4 | 0.24% | 0.00% | 1.14% | 19.40 | 11.49 | 71.31 | 8 |
| 5 | 0.35% | 0.08% | 1.19% | 59.51 | 29.72 | 290.41 | 5 |
| 6 | 1.01% | 1.13% | 2.28% | 47.67 | 36.55 | 162.49 | 1 |
| 7 | 0.82% | 0.91% | 1.82% | 82.70 | 39.85 | 359.54 | 2 |
| 8 | 0.77% | 0.49% | 1.81% | 975.96 | 434.77 | 3033.92 | 0 |
| Average | 0.64% | 0.50% | 1.94% | 151.58 | 70.77 | 500.50 | 4 |

**GRASP**

We examine two GRASP algorithms: in the first one (GRASP(1)), we utilize one-swap in local search, and the second one (GRASP(2)) uses both one-swap and two-swap procedures, leading to a larger neighborhood space. For both implementations we set $max_iter = 10$, $m = 3$; in other words, the algorithm runs ten iterations and the *selected-list* contains three instances in each iteration. Table 2.14 displays a performance comparison between GRASP and SS.

GRASP(1) achieves an optimality gap of 3.90% on average, with a running time of 24.30 seconds, and GRASP(2) has an average optimality gap of 1.59% with a running time of 36.99 seconds. Compared with SS, GRASP(1) is 22 times faster but suffers a slightly worse solution quality, meaning that we can save 95.50% of running time by sacrificing 0.98% of quality. On the other hand, GRASP(2) outperforms SS regarding both quality and efficiency. By choosing GRASP(2) rather than SS, we can decrease the optimality gap by 1.33% and save 93.14% of running time on average. Moreover, compared with CPLEX, both GRASP versions are at least five times faster.

Table 2.14: Overall Comparisons between GRASP and SS

| | Gap | | | Running Time | | |
|---|---|---|---|---|---|---|
| Data Set | GRASP(1) | GRASP(2) | SS | GRASP(1) | GRASP(2) | SS |
| 1 | 4.26% | 1.73% | 3.25% | 1.23 | 1.31 | 7.25 |
| 2 | 2.50% | 1.04% | 3.06% | 1.76 | 2.90 | 29.68 |
| 3 | 4.68% | 1.58% | 3.55% | 2.98 | 5.52 | 41.20 |
| 4 | 2.87% | 0.25% | 1.52% | 7.61 | 10.80 | 50.49 |
| 5 | 1.95% | 1.20% | 2.52% | 10.06 | 25.21 | 123.72 |
| 6 | 4.50% | 3.45% | 4.56% | 12.82 | 15.76 | 141.40 |
| 7 | 4.33% | 0.73% | 2.86% | 10.03 | 20.10 | 190.34 |
| 8 | 6.12% | 2.76% | 2.03% | 147.55 | 214.31 | 3727.26 |
| Average | 3.90% | 1.59% | 2.92% | 24.25 | 36.99 | 538.92 |

Tables 2.15 and 2.16 summarize the statistical results of the two GRASP algorithms. In both tables, we list the average, median, and maximum value of optimality gap and running time. The last column shows the number of optimal solutions we obtain for each set.

For GRASP(1), the range of optimality gap is relatively large, which is improved in GRASP(2). Although the maximum values of the gap are high in both algorithms, the median gaps are small, with an average of 2.41% in GRASP(1) and 0.96% in GRASP(2). Regarding the running time, we find a clear rising trend from small cases to large ones. However, the differences are much less than SA and SS algorithms, because in GRASP, we can only increase the candidate pool used to create the *selected-list*, but the size of *selected-list* does not change (m is still three). The extra time is spent in the two-swap local search.

Table 2.15: GRASP with One Swap Local Search

| | Gap | | | Running Time (s) | | | # Optimal |
|---|---|---|---|---|---|---|---|
| Data Set | average | median | max | average | median | max | Solutions |
| 1 | 4.26% | 2.63% | 15.59% | 1.23 | 0.72 | 3.58 | 1 |
| 2 | 2.50% | 2.74% | 5.55% | 1.76 | 1.34 | 4.38 | 1 |
| 3 | 4.31% | 2.72% | 15.99% | 3.33 | 1.73 | 10.55 | 0 |
| 4 | 2.87% | 2.33% | 6.06% | 7.61 | 3.54 | 20.41 | 1 |
| 5 | 1.95% | 1.67% | 5.68% | 10.06 | 5.80 | 39.04 | 0 |
| 6 | 4.50% | 2.79% | 20.33% | 12.82 | 5.56 | 58.81 | 0 |
| 7 | 4.33% | 1.53% | 16.64% | 10.03 | 8.14 | 26.89 | 0 |
| 8 | 6.12% | 2.87% | 19.96% | 147.55 | 91.68 | 747.96 | 0 |
| Average | 3.90% | 2.41% | 13.23% | 24.30 | 14.81 | 113.95 | 0.4 |

Table 2.16: GRASP with One and Two Swap Local Search

| | Gap | | | Running Time (s) | | | # Optimal |
|---|---|---|---|---|---|---|---|
| Data Set | average | median | max | average | median | max | Solutions |
| 1 | 1.73% | 1.50% | 4.50% | 1.31 | 0.70 | 3.81 | 4 |
| 2 | 1.04% | 1.02% | 2.46% | 3.06 | 1.88 | 12.25 | 2 |
| 3 | 1.58% | 1.39% | 2.49% | 5.79 | 2.65 | 33.30 | 0 |
| 4 | 0.25% | 0.00% | 1.07% | 10.80 | 6.88 | 41.06 | 6 |
| 5 | 1.20% | 0.41% | 6.18% | 25.21 | 8.33 | 113.00 | 2 |
| 6 | 3.45% | 1.64% | 17.41% | 15.76 | 7.02 | 64.52 | 0 |
| 7 | 0.73% | 0.59% | 1.64% | 20.10 | 8.79 | 92.25 | 0 |
| 8 | 2.76% | 1.10% | 17.48% | 214.31 | 125.73 | 549.45 | 0 |
| Average | 1.59% | 0.96% | 6.65% | 37.04 | 20.25 | 113.70 | 2 |

Figure 2.7 shows the overall average gap and running time for the five algorithms considered in the chapter. According to to the mean of optimality gap, the best choice is SA with three neighbors (i.e., SA(3)), followed by SA with two neighbors (SA(2)), GRASP with one and two swap local search (GRASP(2)), SS, and GRASP with one swap local search (GRASP(1)). When we look at the algorithm efficiency, GRASP with one swap local search (GRASP(1)) is the fastest, followed by GRASP with one and two swap local search (GRASP(2)), SA with three neighbors (SA(3)), SA with two neighbors (SA(2)), and SS. We suggest that for small problems, both SA(2) and SA(3) can quickly identify an excellent solution (gap is less than 1%); for large problems, GRASP(2) can solve the case in 4 minutes with a gap of around 3%.

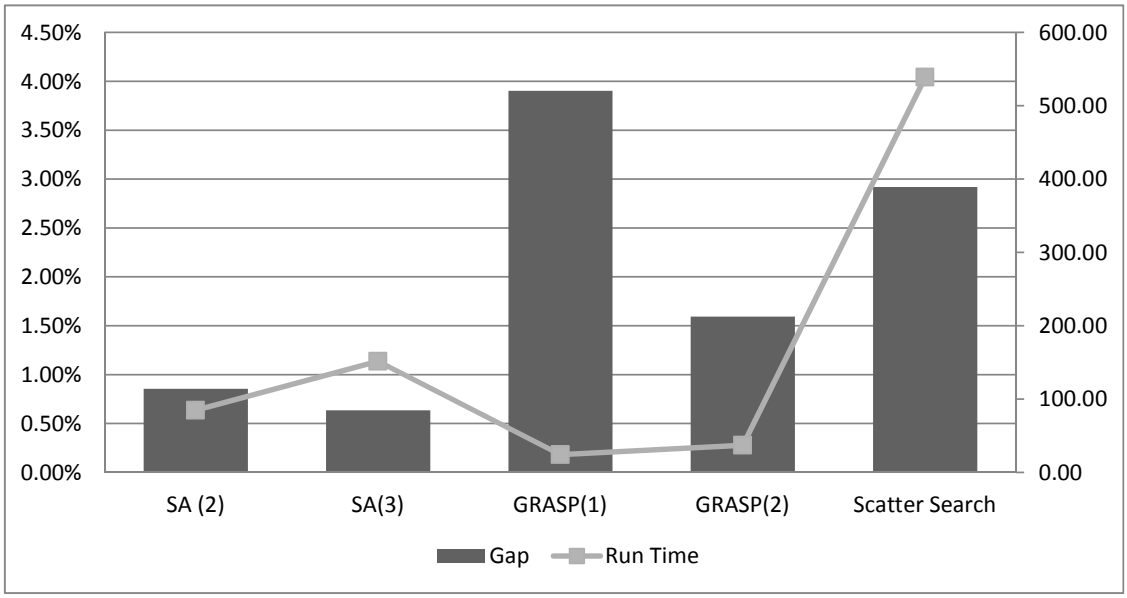



Figure 2.7: Optimality Gap and Running Time of All Algorithms

## 2.7. CONCLUSIONS

In this paper, we investigate meta-heuristic approaches for the PDSD problem. Although there have been a number of studies on PDSD problems, the meta-heuristic-based solutions to these problems have not been fully explored. This chapter presents two solution approaches based on SA and GRASP heuristics.

Inspired by Tabu Search, We create long and short memory lists to improve the searching efficiency in both SA and GRASP algorithms. A long-term memory list is used to avoid repetitive searching, and two short memory lists are refreshed at each iteration to keep track of the qualified candidates. The proposed hybrid algorithms are applicable to other p-median problems besides PDSD. Through a series of experiments, we find that both SA and GRASP outperform SS, our benchmarking algorithm. SA with three neighbors provides the smallest optimality gap of 0.64% and finishes in three minutes in average; GRASP with one-swap and two-swap achieves a optimality gap 0.96% within 38 seconds, which saves 80.72% of time compared with CPLEX.

One future plan includes a testing of the two algorithms in larger cases. In addition, as the GRASP algorithm can run in parallel on a multi-core system, we plan to study the three-stage PDSD problem, taking decisions on supplier location into account. However, for three-stage PDSD problems, the subproblems must be handled carefully due to the increased complexity of the transshipment problem. Finally, the proposed solution approaches are suitable even with generalized cost functions because the form of the cost function does not affect the searching method, although it may increase the difficulty of solving the subproblems.

## CHAPTER 2 REFERENCES


Ahmadi, S. and Osman, I. H. (2005). Greedy random adaptive memory programming search for the capacitated clustering problem. European Journal of Operational Research, 162(1):30–44. Logistics: From Theory to Application.

Arostegui, M. A., Kadipasaoglu, S. N., and Khumawala, B. M. (2006). An empirical comparison of tabu search, simulated annealing, and genetic algorithms for facilities location problems. International Journal of Production Economics, 103(2):742–754.

Brimberg, J. and Drezner, Z. (2013). A new heuristic for solving the p-median problem in the plane. Computers & Operations Research, 40:427–437.

Chiyoshi, F. and Galvao, R. D. (2000). A statistical analysis of simulated annealing applied to the p-median problem. Annals of Operations Research, 96:61–74.

Contreras, I. A. and Diaz, J. A. (2008). Scatter search for the single source capacitated facility location problem. Annals of Operations Research, 157:73–89.

Diaz, J. A. and Fernandez, E. (2006). Hybrid scatter search and path relinking for the capacitated p-median problem. European Journal of Operational Research, 169:570–585.

Fahimnia, B., Luong, L., and Marian, R. (2012). Genetic algorithm optimisation of an integrated aggregate production distribution plan in supply chains. International Journal of Production Research, 50(1):81–96.

Feo, T. A. and Resende, M. G. (1989). A probabilistic heuristic for a computationally difficult set covering problem. Operations research letters, 8(2):67–71.

Feo, T. A. and Resende, M. G. (1995). Greedy randomized adaptive search procedures. Journal of global optimization, 6(2):109–133.

Festa, P. and Resende, M. G. C. (2009). An annotated bibliography of grasp c part ii : Applications. International Transactions in Operational Research, 16:131–172.

Goetschalckx, M., Vidal, C. J., and Dogan, K. (2002). Modeling and design of global logistic systems: A review of integrated strategic and tactical models and design algorithms. European Journal of Operational Research, 143:1–18.

Jawahar, N. and Balaji, A. (2009). A genetic algorithm for the two-stage supply chain distribution problem associated with a fixed charge. European Journal of Operational Research, 194:496–537.

Kariv, O. and Hakimi, L. (1979). An algorithmic approach to network location problems. ii: The $p$-medians. SIAM Journal on Applied Mathematics, 37(3):539–560.

Keskin, B. B. and Üster, H. (2007a). Meta-heuristic approaches with memory and evolution for a multi-product production/distribution system design problem. European Journal of Operational Research, 182(2):663–682.

Keskin, B. B. and Üster, H. (2007b). A scatter search-based heuristic to locate capacitated transshipment points. Computers & Operations Research, 34(10):3112–3125.

Klrkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. Science, 220(4598):671–680.

Mladenovic, N., Brimberg, J., Hansen, P., and Moreno-Perez, J. A. (2007). The p-median problem: A survey of metaheuristic approachest. European Journal of Operational Research, 179:927–939.

Olhager, J., Pashaei, S., and Sternberg, H. (2015). Design of global production and distribution networks a literature review and research agenda. International Journal of Physical Distribution and Logistics Management, 45(1-2):138–158.

Pedrola, O., Ruiz, M., Velasco, L., Careglio, D., de Dios, O. G., and Comellas, J. (2013). A grasp with path-relinking heuristic for the survivable ip/mpls-over-wson multi-layer network optimization problem. Computers & Operations Research, 40:3174–3187.

Pessoa, L. S., Resende, M. G. C., and Ribeiro, C. C. (2013). A hybrid lagrangean heuristic with grasp and path-relinking for set k-covering. Computers & Operations Research, 40:3132–3146.

Resende, M. G. C. and Werneck, R. F. (2004). A hybrid heuristic for the p-median problem. Annals of Operations Research, 10(1):59–88.

Saez-Aguado, J. and Trandafir, P. C. (2012). Some heuristic methods for solving p-median problems with a coverage constraint. European Journal of Operational Research, 220:320–327.

Stefanello, F., de Araujo, O. C. B., and Müller, F. M. (2015). Matheuristics for the capacitated p-median problem. International Transactions in Operational Research, 22(1):149–167.

Tragantalerngsak, S., Holt, J., and Rönnqvist, M. (1997). Lagrangian heuristics for the two-echelon, single-source, capacitated facility location problem. European Journal of Operational Research, 102(3):611–625.

Tragantalerngsak, S., Holt, J., and Rönnqvist, M. (2000). An exact method for the two-echelon, single-source, capacitated facility location problem. European Journal of Operational Research, 123(3):473–489.

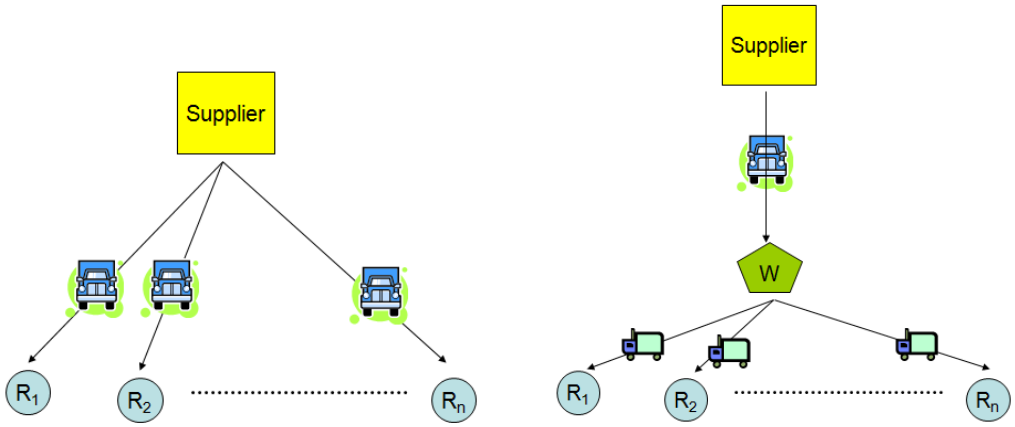
# CHAPTER 3

# INTEGRATED WAREHOUSE LOCATION AND INVENTORY DECISIONS IN A MULTI-LOCATION NEWSVENDOR PROBLEM

## 3.1. INTRODUCTION

The classical newsvendor problem is reflective of many real-life situations, and it is widely applicable for decision making in various areas such as fashion, sporting goods, and electronics industries, at both manufacturing and retail levels. The classical newsvendor problem determines the order quantity for a single product during a single period by maximizing the expected profit under probabilistic demand (Silver et al., 1998). The newsvendor model considers a shortage cost associated with each demand that cannot be fulfilled. Also, if there is any stock remaining in inventory at the end of the period, it is sold at a salvage value, or it is disposed of. Therefore, there is a trade-off between ordering/producing too little and too much due to the uncertainty of the demand.

The classical newsvendor problem may not apply to some modern businesses since currently many companies tend to adopt new expansion trends to make shopping more convenient and to elevate market share. To address this practical need, researchers have studied the multi-location newsvendor problem. Khouja (1999) provides an extensive overview that summarizes the variants of newsvendor problem. In this paper, we consider a multi-location newsvendor problem where the demand at each location is satisfied by the same supplier. More specifically, our model consists of a single supplier and multiple newsvendor locations, each of which observes a random demand for a seasonal (short-life time) product. In the traditional (decentralized) system, before the sales period, each retailer (newsvendor) places an order to meet the random demand, which is assumed to be independent of the demand




(a) A Decentralized System (b) A Centralized System

Figure 3.1: Decentralized and centralized systems

of other newsvendors. The retailer determines the order quantity in a way such that it maximizes its profits. On the other hand, upon receiving the orders, the supplier ships the order to each retailer in a timely manner. Therefore, each party (the supplier and the retailers) in the supply chain will maximize its gain. We refer to this system as the decentralized system (DS), as shown in Figure 3.1a.

Many researchers studied the multi-location newsvendor problem and noted that when demands from several retailers are consolidated, the expected holding and penalty costs of the consolidated system are lower than that of the decentralized system (Chang and Lin, 1991; Cherikh, 2000; Eppen, 1979; Lin et al., 2001). To take advantage of the centralized ordering and the economies of scale in transportation, the supplier may prefer each retailer to order through a distribution center (DC). Hence, the second supply chain system we consider is a centralized system where a DC exists to coordinate and consolidate the orders from the retailers. In the centralized system, the supplier oversees the whole supply chain and fulfills demands of retailers through the DC, as shown in Figure 3.1b. The DC receives a single consolidated shipment from the supplier and distributes the orders to the retailers when demand occurs. Many large retail industries and car manufacturers opt for large warehouses that distribute orders to local stores and dealerships across a region, rather than carrying

separate inventory at different locations. Moreover, the idea of centralizing stocks has been widely adopted in the fast-growing e-commerce industry.

The motivation for these models stems from recent supply chain management (SCM) practices. First of all, in the last decade, SCM strategies that transfer stock level decisions and stock-keeping responsibilities from downstream parties to the upstream have become popular. These new approaches include vendor managed inventory (VMI) (Cachon, 2001; Cetinkaya and Lee, 2000; Cetinkaya et al., 2006; Toptal and Cetinkaya, 2006), consignment vendor managed inventory, vendor hub (Lee and Chu, 2005), drop-shipping (Netessine and Rudi, 2004), and Click-and-Mortar (Lee and Whang, 2000). Under the VMI process, retailers can share demand information with a supplier who should then make the stock replenishment decisions for both itself and the retailers. A variant of VMI is consignment VMI, in which the supplier makes the stock replenishment decisions and owns the stock until it is sold. Currently, Walmart, one of the leading retailers, operates under a consignment VMI process. Vendor hub is an another popular method adopted by numerous electronics/computer manufacturers. These manufacturers request a supplier to keep a minimum stock in a hub, and they only pay the supplier when goods are pulled and consumed. The increasing popularity of these models drives us to wonder whether the gains obtained from the centralized system are superior to the ones from the decentralized system under different settings of transportation cost.

Another motivation comes from the utilization of transportation. In the decentralized system, the supplier ships the orders separately to each retailer and possibly over long distances. Moreover, these trucks may or may not be full depending on the order size, causing a potential waste which could be reduced by consolidation. For this purpose, the supplier may choose to outsource transportation to a third party logistics provider. Hence, explicit modeling of transportation costs for both decentralized and centralized systems not only provides insights for the supply chain managers, but also for the third party logistics companies.

The questions we aim to answer include:

- How do decentralized and centralized solutions differ?
- What would be the payoffs for each party in the supply chain? and if there are any savings, how would these savings be distributed in the supply chain?
- Under what circumstances it is preferred to implement a centralized system?
- How does the DC location impact the results?
- How do different transportation costs impact the results? In other words, how would the results change given economies of scale and/or economies of distance?

Although there have been studies on the impact of inventory centralization, no prior research has explicitly modeled the relation between inventory ordering and the DC location in a multi-location newsvendor problem. Moreover, the explicit modeling of the transportation cost provides further insights for the relation between the centralized and decentralized systems. We summarize the contributions of this work as follows:

- We develop two newsvendor models, including a decentralized system model (DSM) and a centralized system model (CSM) under a practical newsvendor setting. Both models help decision makers of a firm maximize the overall profit with estimated demand following a normal distribution. Besides, CSM determines the optimal DC location where a DC is suggested to build.
- We compare DSM and CSM regarding three modes of transportation cost, and develop solution approaches to these particular problems. In addition, we consider a change on CSM: instead of setting a DC at the optimal location, it may be cost-efficient to set it at the closest retailer to the optimal location. This option is referred to as Retailer-as-DC, and presents a high practical value.

- We conduct a suite of experiments to validate our models and solution approaches. The results demonstrate that 1) CSM outperforms DSM in large-scale network with more retailers, 2) the proposed models work well for simulated demands, with a result in line with the one obtained from expected demands, and 3) Retailer-as-DC shows great profitability and is highly recommended if the chosen retailer is sufficiently close to the optimal DC location.

The rest of this chapter is organized as follows. Section 3.2 provides a review of related work. Section 3.3 presents the assumptions and notations. In Section 3.4, we formulate the DSM and CSM models and develop solution approaches for each model. Section 3.5 is a comparison of DSM and CSM. We validate our models and solution approaches in Section 3.6, and conclude the chapter in Section 3.7.

## 3.2. LITERATURE REVIEW

We provide a review of related studies in two areas: multi-location newsvendor problems and location-inventory problems.

### 3.2.1. Multi-Location Newsvendor Problem

The single period newsvendor problem has been widely discussed in the literature. Khouja (1999) and Silver et al. (1998) provide comprehensive reviews on this subject. Eppen (1979) proposes a multi-location newsvendor problem with normal demand and linear holding and penalty cost functions at each location. This paper shows that the total cost in a decentralized model exceeds the one in a centralized system with a difference that depends on the correlation of demands. For uncorrelated and identically distributed demands, however, the expected cost of a centralized facility increases as the square root of the number of consolidated centers. Chang and Lin (1991) extend Eppen's model by taking transshipment cost among multiple retailers into account. The authors give the condition of cost savings on a centralized model compared with a decentralized model with certain functions of holding cost, penalty cost and transshipment cost, in which the transportation cost only includes the

transshipment of excess order quantities among retailers but ignores the transportation from a supplier to retailers. Slikker et al. (2005) also consider a multi-newsvendor problem with transshipment among retailers. Different from Chang and Lin (1991), the authors analyze the problem with a cooperative game and show that stable divisions of expected profits also exist if transshipment costs are positive. Furthermore, they show that given different retail and wholesale prices rather than anonymous prices, stable divisions of expected profits exist as well. Cherikh (2000) also recommends centralization with a condition that a portion or all of the excess demand at a location can be reallocated among other newsvendors. Lin et al. (2001) develop a single-period, single-product inventory model with several individual sources of demand. They prove that the Eppen's property still holds under a multi-source demand setting. Hartman and Dror (2005) consider this problem with normally distributed and correlated individual demand. They develop a greedy heuristic procedure to find the optimal centralized policy. Lee and Chu (2005) discuss whether it is profitable to move the stock-keeping responsibility to the upstream (supplier) in a newsvendor inventory control system. They find that the critical fractile used in a newsvendor problem to determine the optimal stock level can often be used as a yardstick to quickly assess the merits of adopting the new mechanism. Furthermore, they show that when the critical fractile of the supplier is greater than that of retailers, adopting the new mechanism always makes both members better off, provided that some risk-sharing rules can be adopted. Ji and Shao (2006) consider a model with fuzzy demands and quantity discounts in a hierarchical decision system, in which the supplier decides the wholesale prices of newspaper to maximize its profit, while the retailers decide the order size of newspaper to maximize their expected profits.

Although researchers have extended the multi-location newsvendor problem in various directions, the transportation cost has not been thoroughly considered, and the characteristics of the centralized DC and transportation modes have not discussed. Our research attempts to fill this gap and provide inventory-location joint models to analyze how the location and the capacity of a DC influence a centralized newsvendor system.

### 3.2.2. Location-Inventory Problem

In general, the location-inventory problems have two types of decisions: the DC location and the optimal order quantity of each retailer. In most prior studies, the locations of DCs are selected from a finite set of candidate locations, which makes a discrete location model. Our model is a continuous problem, as it decides the coordinates of a DC instead of choosing it from candidate locations. For the inventory decisions, many prior studies focus on periodic-review inventory policies such as the (r, q) policy (insert reference) or the approximation of this policy, which is essentially the classic economic order quantity (EOQ) model. It is reasonable to combine the long-term decision on a strategic location with the periodic inventory context, since both decisions are made for a finite long period. In our problem, we consider the location problem with a single period newsvendor setting, as the demand for seasonal goods in a selling season may boost quickly and drop sharply after the critical selling period. For example, the demand for eastern eggs increases dramatically in April, while there is no demand at all for the rest of a year. Retailers would like to increase the inventory of eastern eggs temperately and do not want it to influence the regular stock of other products. Thus, a temporary warehouse is a reasonable choice to hold the seasonal inventories, which motivates us to combine the location problem with a newsvendor structure.

Shen et al. (2003) propose a joint location-inventory model in which location, shipment, and nonlinear safety stock inventory costs are taken into account in the same model. They develop an integrated approach to determine the number of DCs to establish, the DC locations, and the magnitude of inventory to maintain at each DC. In their model, the potential DCs are built at existing retailers rather than setting new facilities as central DCs.

Tancrez et al. (2012) study a location-inventory problem in a three-level supply network. The authors formulate a nonlinear model including transportation and fixed holding costs, which decomposes into a closed-form equation and a linear program with fixed DC flows. They develop an iterative heuristic procedure that estimates the DC flows a priori, solves the linear program, and then improves the DC flow estimations.

## 3.3. GENERAL ASSUMPTIONS AND NOTATION

We consider a supplier that markets and delivers goods in a single selling season to $n$ retailers. This setting fits the supply chain for many seasonal products such as fashion goods, holiday items, and fast-moving electronic products. We assume that in both DSM and CSM, the locations of the supplier and the geographically dispersed retailers are known. The DC location in the CSM is continuous and needs to be decided. The selling season is relatively short and has a well-defined beginning and end. Therefore, retailers must decide how much to order before the start of the season. Each retailer faces a random demand, $x$, that is determined by a continuously differentiable probability density function $f_i(x)$ with a support on $[0, \infty)$. The cumulative distribution function (CDF) of $f_i(x)$ is denoted as $F_i(x)$. Every retailer experiences stochastic demand which is unknown at the moment of ordering.

In the DSM, retailers receive orders directly from the supplier, meaning that the supplier needs to solve $n$ newsvendor problems with transportation cost for all retailers. On the contrary, in the centralized model, the supplier observes the demand distribution of all retailers and then determines a total order quantity $Q_0$ shipped to the DC. Then, at the beginning of a selling season, after observing the actual demand at retailers, the DC will send the actual order $x_i$ to each retailer. If any inventory remains at the end of the period, a discount is used to sell it, or it is disposed of (i.e., salvage value). We assume that since there is only one kind of product, and the salvage value is the same for all retailers. If the order quantity at a retailer is lower than the actual demand at that retailer, a shortage cost occurs to penalize the lost profit.

Transportation cost is a critical component in our model. In the DSM, there is a transportation cost from the supplier to retailers. In the CSM, there are transportation costs between the supplier and the DC and also between the DC and retailers. Our model considers the transportation cost as a function of traveled distance and/or order size.

We define the following notations:

**Parameters:**

| | |
|---|---|
| $\mathcal{I}$ | set of retailers, $i = 1, \ldots, n$. |
| $L_0$ | location of the supplier, $L_0 = (a_0, b_0)$. |
| $L_i$ | location of retailer $i$, $L_i = (a_i, b_i)$, $\forall i \in \mathcal{I}$. |
| $x_i$ | random quantity demanded at retailer $i$, $\forall i \in \mathcal{I}$. |
| $\mu_i$ | the mean of demand at retailer $i$, $\forall i \in \mathcal{I}$. |
| $\sigma_i$ | the standard deviation of demand at retailer $i$, $\forall i \in \mathcal{I}$. |
| $f_i(x_i)$ | the probability density function of $x_i$ with support on $[0, \infty)$. |
| $F_i(x_i)$ | the CDF of $x_i$. |
| $x_0$ | total demand at DC, $x_0 = \sum_{i \in \mathcal{I}} x_i$ |
| $f_0(x_0)$ | the density function of the total demand $x_0$ with support on $[0, \infty)$. |
| $F_0(x_0)$ | the CDF of $x_0$. |
| $s$ | selling (retail) price per unit, $\forall i \in \mathcal{I}$. |
| $c$ | unit cost of supplier. |
| $v$ | salvage value per unit. |
| | It is assumed that $s > c > v$. |
| $b$ | shortage (out-of-stock) cost per unit at retailer $i$, $\forall i \in \mathcal{I}$. |
| $K_i$ | retailer's fixed cost of replenishment, $\forall i \in \mathcal{I}$. |
| $K_0$ | fixed replenishment cost for the supplier. |
| $ds_i$ | distance between the supplier at $L_0$ and retailer at $L_i$ |
| $TS_i(Q_i, ds_i)$ | transportation cost between the supplier/DC and retailer $i$, $\forall i \in \mathcal{I}$. |
| $T0(Q_0, d_0)$ | transportation cost between the supplier and the DC. |

**Decision variables:**

| | |
|---|---|
| $Q_i$ | order size of retailer $i$ from the warehouse, $\forall i \in \mathcal{I}$. |
| $Q_0$ | order size of the distribution center from the supplier. |
| $X$ | location of the DC, $X = (x, y)$. |
| $d_0$ | distance between the supplier at $L_0$ and the DC at $X$ |
| $d_i$ | distance between the DC at $X$ and retailer at $L_i$ |
| $\mathcal{D}$ | $\{d_0, d_1, \ldots, d_n\}$ |

## 3.4. MODELS AND SOLUTIONS

In this section, we formulate DSM and CSM. For each model, we consider three modes of transportation costs, which influence the problem complexity. Moreover, we develop the solution approach for each proposed model.

### 3.4.1. Model 1: Decentralized System Model (DSM)

We can decompose the DSM into multiple independent small problems, each of which consists of the suppler and a retailer and can be optimized individually. In particular, the problem at retailer $i$ is to determine the order quantity $Q_i > 0$ by maximizing the expected profit $\Pi_i(Q_i) = E[\pi_i(Q_i)]$, where $\forall i \in \mathcal{I}$, $\pi_i(Q_i)$ is:

$$\pi_i(Q_i) = \begin{cases} sx_i - wQ_i - b(x_i - Q_i) - K_i - TS_i(Q_i, ds_i) & \text{if } x_i \geq Q_i \\ sx_i + v(Q_i - x_i) - wQ_i - K_i - TS_i(Q_i, ds_i) & \text{if } x_i < Q_i \end{cases}$$

On the other hand, the supplier solves a problem to maximize its expected profit, $\Pi_{sp}(Q_1, Q_2, \ldots, Q_n)$:

$$\Pi_{sp}(Q_1, Q_2, \ldots, Q_n) = \sum_{i \in \mathcal{I}} (w - c)Q_i - K_s$$

Therefore, the total profit obtained in DSM is

$$\begin{aligned}\Pi^D(Q_1,\ldots,Q_n) &= \Pi_{sp}(Q_1,Q_2,\ldots,Q_n)+\sum_{i\in\mathcal{I}}\Pi_i(Q_i)\\ &= \sum_{i\in\mathcal{I}}(s\int_0^{\infty}x_i f(x_i)dx_i - b\int_{Q_i}^{\infty}(x_i-Q_i)f(x_i)dx_i\\ &\quad +v\int_0^{Q_i}(Q_i-x_i)f(x_i)dx_i - cQ_i - TS_i(Q_i,ds_i)-K_i)-K_0\end{aligned}$$

$$\textit{subject to} \qquad F_i(Q_i)\geq\gamma,\forall i\in\mathcal{I}$$

The transportation cost we consider in this model has both fixed and variable parts. We assume that transportation costs follow three different structures. Let $T_i(Q_i,d_i)$ be the transportation cost between the supplier and retailer $i$; meanwhile, we define $T_i$ as a function of quantity $Q_i$ and/or distance $d_i$. There are three cases.

**As a function of quantity shipped:**

$$\begin{aligned}TS(Q_i,ds_i) &= ps_i^q+rs_i^qQ_i \qquad (DSM)\\ T0(Q_0,d_0) &= p_0^q+r_0^qQ_0 \qquad (CSM)\\ TI(Q_i,d_i) &= p_i^q+r_i^qQ_i \qquad (CSM)\end{aligned}$$

**As a function of distance traveled:**

$$\begin{aligned}TS(Q_i,ds_i) &= ps_i^d+rs_i^dds_i \qquad (DSM)\\ T0(Q_0,d_0) &= p_0^d+r_0^dd_0 \qquad (CSM)\\ TI(Q_i,d_i) &= p_i^d+r_i^dd_i \qquad (CSM)\end{aligned}$$

**As a function of quantity and distance:**

$$\begin{aligned} TS(Q_i, ds_i) &= ps_i^{qd} + rs_i^{qd} Q_i ds_i \qquad (DSM) \\ T0(Q_0, d_0) &= p_0^{qd} + r_0^{qd} Q_0 d_0 \qquad (CSM) \\ TI(Q_i, d_i) &= p_i^{qd} + r_i^{qd} Q_i d_i \qquad (CSM) \end{aligned}$$

in which $ps_i^q$, $ps_i^{qd}$ and $ps_i^d$ are the bundling cost that is associated with one shipment. Meanwhile, $rs_i^q$ is transportation cost per unit product, $rs_i^{qd}$ is the transportation cost per mile per unit product, and $rs_i^d$ is the transportation cost per mile.

### 3.4.2. Solution of DSM

**Case 1:** $TS_i(Q_i, ds_i) = ps_i^q + rs_i^q Q_i$

When the transportation cost has a linear relation with the order quantity, we can consider the unit transportation cost $\alpha^q$ as a part of the unit variable cost. Therefore the optimal order quantity of each retailer $i$ has the traditional newsvendor problem solution structure. For each retailer we have:

$$\begin{aligned} \frac{d}{dQ_i}\Pi^{D1}(Q_i) &= s\mu_i - cQ_i - bE[x_i - Q_i]^+ + vE[Q_i - x_i]^+ \\ &= (s-c)\mu_i - (b-c)E[x_i - Q_i]^+ - (c-v)E[Q_i - x_i]^+ \\ &= (s-c)\mu_i - G(Q_i) \end{aligned}$$

Where

$$G(Q_i) = (b-c)E[x_i - Q_i]^+ + (c-v)E[Q_i - x_i]^+$$

Then maximizing the total profit is equivalent to minimizing $\sum_{i\in\mathcal{I}} G(Q_i)$. The optimal order quantity of each retailer is:

$$Q_i^* = F_i^{-1}(\frac{c_u}{c_u + c_o}), \forall i,$$

where $c_u = b - c - rs_i^q$, $c_o = c + rs_i^q - v$, and $F_i^{-1}$ is the CDF of $x_i$ following $(\mu_i, \sigma_i)$. The second derivative is $\frac{d^2}{dQ_i}\Pi^{D1}(Q_i) = -(b-v)f(Q_i) \leq 0$, implying a concave profit function is concave on $Q_i$. Also, since the constraint set $F(Q_i) \leq \gamma$ applies, we have the following solutions of DSM when the transportation cost is linearly dependent on $Q_i$:

$$Q_i^* = \max\{F_i^{-1}(\gamma), F_i^{-1}(\frac{b-c-rs_i^q}{b-v})\}$$

**Case 2:** $TS_i(Q_i, ds_i) = ps_i^d + rs_i^d ds_i$

When the transportation cost is linearly dependent on the distance between the two end points, the cost function is irrelevant to order size $Q_i$. Therefore we can solve the simple newsvendor model to obtain the optimal solution.

$$Q_i^* = F_i^{-1}(\frac{c_u}{c_u + c_o}), \quad \forall i \in \mathcal{I}$$

where $c_u = b - c$, $c_o = b - v$. Also,

$$Q_i^* = max\{F_i^{-1}(\gamma), F_i^{-1}(\frac{b-c}{b-v})\}, \quad \forall i \in \mathcal{I}$$

**Case 3:** $TS_i(Q_i, d_i) = ps_i^{qd} + rs_i^{qd} Q_i ds_i$

When the transportation cost is dependent on both order quantity and distance, we still have similar result as case 1 and 2, because in DSM locations of the supplier and retailers are given.

$$Q_i^* = F^{-1}(\frac{c_u}{c_u + c_o}), \forall i \in \mathcal{I}$$

where $c_u = b - c - rs_i^{qd}$, $c_o = c + rs_i^{qd} - v$. We can rewrite $Q_i^*$ as:

$$Q_i^* = max\{F_i^{-1}(\gamma), F_i^{-1}(\frac{b-c-rs_i^{qd}ds_i}{b-v})\}.$$

### 3.4.3. Model 2: Centralized System Model (CSM)

Different from DSM, the supply chain of CSM adds a DC that takes orders from all retailers. The DC location is a decision variable that determines the overall profit. Before a selling season starts, the supplier needs to decide the total order quantity $Q_0$ as well as the optimal DC location to maximize the expected profit of the whole system. When the season begins, each retailer will place an order based on its real demand. The DC ensures that each retailer will receive its full order, indicating a potential salvage or shortage. The total realized demand at DC is $x_0 = \sum_{i=1}^{N} x_i$. Assume demands at retailers are independent, therefore $x_0$ also follows the normal distribution with a mean $\mu_0 = \sum_{i=1}^{N} \mu_i$ and a standard deviation $\sigma_0 = \sqrt{\sum_{i=1}^{N} \sigma_i^2}$.

We formulate the overall expected profit, $\Pi^C(Q_0, \mathcal{X})$, as follows:

$$
\begin{aligned}
\Pi^C(Q_0, \mathcal{X}) &= s\int_0^{\infty} x_0 f(x_0)dx_0 - b\int_{Q_0}^{\infty}(x_0 - Q_0)f(x_0)dx_0 \\
&\quad + v\int_0^{Q_0}(Q_0 - x_0)f(x_0)dx_0 - cQ_0 - \sum_{i\in\mathcal{I}} K_i \\
&\quad - \sum_{i\in\mathcal{I}\cup\{0\}} TI_i(x_i, d_i) - T0(Q_0, d_0)
\end{aligned}
$$

subject to

$$
\begin{aligned}
Q_0 &\geq \sum_{i\in\mathcal{I}} Q_i, \\
F_i(Q_i) &\geq \gamma, \qquad \forall i \in \mathcal{I}, \\
d_0(x, y) &\geq 0, \\
d_i(x, y) &\geq 0, \qquad \forall i \in \mathcal{I}.
\end{aligned}
$$

In this formulation, the expected profit is a function of total order quantity $Q_0$ and the DC location $(X, y)$, which in return impacts the distances $d_i$, $i \in \mathcal{I}$ and $d_0$. The first part of the objective function captures the newsvendor related revenue and cost, including total revenue, shortage cost and salvage revenue, production cost of supplier, and fixed replenishment cost at each retailer. Again, we assume $K_i = 0$. The second part is the

transportation cost from supplier to DC and then DC to retailers. In CSM we assume that there is no transshipment among retailers. Since the shipments happened after receiving accurate demand information from retailers, the quantity shipped to each retailer is equal to the actual demand $x_i$. The distances between facilities can be calculated using squared Euclidean, Euclidean, or Rectangular distances. Similar to DSM, CSM also requires that the quantity planned to distribute to each retailer $i$, i.e., $Q_i$ needs to satisfy its minimal service level $F_i^{-1}(\gamma)$.

### 3.4.4. Solutions of CSM

In this section, we discuss the solution approaches of CSM under three types of transportation costs. Compared with DSM, there are gains from risk-pooling and centralization of the inventory. But the transportation cost could consume the risk-pooling benefit. We will compare DSM and CSM in next section.

Same as DSM, we consider three different transportation costs functions in CSM.

**Case 1:** $T0(Q_0, d_0) = p_0^q + r_0^q Q_0$**,** $TI_i(Q_i, d_i) = p_i^q + r_i^q Q_i$

For case 1, the DC location has no impact on the expected profit as it is not included in the model. The unit transportation cost $r_0$, however, will affect the optimal order quantity $Q_0$, and therefore, influence the total profit. We assume that the unit transportation costs $r_i^q$ from DC to retailer $i$ are the same for all retailers, i.e., $r_i^q = r^q$. Since the final quantity of products delivered to retailers equals to the actual demands, the transportation cost from the DC to retailers is unrelated to the optimal order quantity $Q_0$. Then we obtain the optimal central order quantity $Q_0^*$:

$$Q_0^* = \max\{\sum_i F_i^{-1}(\gamma), F_0^{-1}(\frac{b-c-r_0^q}{b-v})\}.$$

**Case 2:** $T0(Q_0, d_0) = p_0^d + r_0^d d_0$**,** $TI_i(Q_i, d_i) = p_i^d + r_i^d d_i$

When the transportation cost depends on the travel distance, we can separate the objective function to two independent parts. The first part is the newsvendor problem of DC which

only contains decision variable $Q_0$, and the second part is the location problem to determine the optimal DC location. We can easily find the optimal order quantity at DC is:

$$Q_0^* = max\{\sum_i F_i^{-1}(\gamma), F_0^{-1}(\frac{c_u}{c_u + c_o}) = F_0^{-1}(\frac{b-c}{b-v})\}.$$

For the location problem we want to minimize the total weighted travel distance of the entire supply chain network:

$$\text{minimize} \qquad \sum_{i\in\mathcal{I}} TI_i(Q_i, d_i) + T0(Q_0, d_0) = \sum_{i\in\mathcal{I}\cup\{0\}} (p_i^d + r_i^d d_i).$$

in which:

| | |
|---|---|
| $p_i^d$ | is the bundling cost that is associated with one shipment, |
| $r_i^d$ | is the unit transportation cost, |
| $X = (x, y)$ | is the location of DC, |
| $L_i = (a_i, b_i)$ | is the location of retailer $i$, |
| $L_0 = (a_0, b_0)$ | is the location of supplier, |
| $d_0$ | is the distance between supplier and DC, |
| $d_i$ | is the distance between DC and retailer $i$, $\forall i \in \mathcal{I}$, |
| $d_i =$ | $[(x - a_i)^2 + (y - b_i)^2 + \epsilon]^{1/2}$. |

Clearly, the center-of-gravity is the optimal solution for this subproblem. We have the optimal DC location:

$$x = \frac{\sum_{i\in\mathcal{I}\cup\{0\}} r_i^d a_i}{\sum_{i\in\mathcal{I}\cup\{0\}} r_i^d}, \qquad y = \frac{\sum_{i\in\mathcal{I}\cup\{0\}} r_i^d b_i}{\sum_{i\in\mathcal{I}\cup\{0\}} r_i^d}.$$

**Case 3:** $T0(Q_0, d_0) = p_0^{qd} + r_0^{qd} Q_0 d_0$**,** $TI_i(Q_i, d_i) = p_i^{qd} + r_i^{qd} Q_i d_i$

In this transportation mode, the total transportation cost is related to both $Q_0$ and $d_0$, which

can be represented as:

$$\sum_{i\in\mathcal{I}} TS_i(x_i,d_i)+T0(Q_0,d_0) = \sum_{i\in\mathcal{I}}[p_i^{qd}+r_i^{qd}x_i * [(x-a_i)^2+(y-b_i)^2+\epsilon]^{1/2}]. \\ +[p_0^{qd}+r_0^{qd}Q_0 * [(x-a_0)^2+(y-b_0)^2+\epsilon]^{1/2}]$$

The separation applied to case 2 is no longer feasible for case 3 due to the nonlinearity of this new form of transportation cost. To this end, we need to examine the concavity of the objective function to determine a feasible solution approach. Let $H(\Pi^C)$ be the Hessian Matrix of $\Pi^C$.

**Theorem 1.** *With the demand $x_0$ following normal distribution, $\Pi^C$ is NOT concave on all decision variables, because $H(\Pi^C)$ is NOT negative definite.*

*Proof.* $H(\Pi^C)$ is negative definite if and only if the scalar $z^TMz$ is negative for every non-zero column vector $z$ of real numbers. Therefore, to show that $H(\Pi^C)$ is NOT negative definite, we only need to find one particular instance of $z$ such that $z^THz$ is positive. To simplify the calculation, we consider a network with a supplier, a DC, and a retailer, i.e., $n=1$. Thus $\Pi^C$ becomes a function of three variables including $Q_0$, $x$, and $y$, and $H(\Pi^C)$ is a $3\times 3$ matrix. In addition, let $p=p_0=p_i$, $r=r_0=r_i$, we write $\Pi^{C1}$ as

$$\begin{aligned}\Pi^{C1} =& s\int_0^{\infty}\frac{x_0}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2\mu^2}(x_0-\mu)^2}\,dx_0 \\ &+ v\int_0^{Q_0}\frac{(Q_0-x_0)}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2\mu^2}(x_0-\mu)^2}\,dx_0 \\ &- b\int_{Q_0}^{\infty}\frac{(-Q_0+x_0)}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2\mu^2}(x_0-\mu)^2}\,dx_0 \\ &- Q_0c-2p-r(Q_0+\mu_1)\sqrt{(x-a_0)^2+(y-b_0)^2}\end{aligned}$$

The corresponding Hessian Matrix is

$$H(\Pi^C) = \begin{bmatrix} H_{00} & H_{01} & H_{02} \\ H_{10} & H_{11} & H_{12} \\ H_{20} & H_{21} & H_{22} \end{bmatrix}$$

in which

$$\begin{aligned} H_{00} =& \frac{(v-b)b}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\mu^2}(x_0-\mu)^2} \\ H_{01} =& -\frac{r\left(x-a_0\right)}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \\ H_{02} =& -\frac{r\left(y-b_0\right)}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \end{aligned}$$

$$\begin{aligned} H_{10} =& -\frac{r\left(x-a_0\right)}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \\ H_{11} =& -\frac{Q_0 r\left(-x+a_0\right)\left(x-a_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}} - \frac{Q_0 r}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \\ & -\frac{r x_1\left(-x+a_1\right)\left(x-a_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}} - \frac{r x_1}{\sqrt{\left(x-a_1\right)^2+\left(y-b_1\right)^2}} \\ H_{12} =& -\frac{Q_0 r\left(x-a_0\right)\left(-y+b_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}} - \frac{r x_1\left(x-a_1\right)\left(-y+b_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}} \end{aligned}$$

$$
\begin{aligned}
H_{20} =& -\frac{r\left(y-b_0\right)}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \\
H_{21} =& -\frac{Q_0 r\left(x-a_0\right)\left(-y+b_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}}-\frac{r x_1\left(x-a_1\right)\left(-y+b_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}} \\
H_{22} =& -\frac{Q_0 r\left(-y+b_0\right)\left(y-b_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}}-\frac{Q_0 r}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \\
& -\frac{r x_1\left(-y+b_1\right)\left(y-b_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}}-\frac{r x_1}{\sqrt{\left(x-a_1\right)^2+\left(y-b_1\right)^2}}
\end{aligned}
$$

Let $z=[z_0, z_1, z_2]^T$, we have

$$
\begin{aligned}
z^T H z =& -z_0 \frac{r z_1\left(x-a_0\right)+r z_2\left(y-b_0\right)}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}}+\frac{z_0^2(v-b) b}{\sigma \sqrt{2 \pi}} e^{-\frac{1}{2 \mu^2}\left(x_0-\mu\right)^2} \\
& -z_1 \frac{r z_0\left(x-a_0\right)}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}}-z_1^2 \frac{Q_0 r\left(-x+a_0\right)\left(x-a_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}} \\
& -z_1^2 \frac{Q_0 r}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}}-z_1^2 \frac{r x_1\left(-x+a_1\right)\left(x-a_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}} \\
& -z_1^2 \frac{r x_1}{\sqrt{\left(x-a_1\right)^2+\left(y-b_1\right)^2}}-z_1 z_2 \frac{Q_0 r\left(x-a_0\right)\left(-y+b_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}} \\
& -z_1 z_2 \frac{r x_1\left(x-a_1\right)\left(-y+b_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}}-z_2 \frac{r z_0\left(y-b_0\right)}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \\
& -z_1 z_2 \frac{Q_0 r\left(x-a_0\right)\left(-y+b_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}}-z_1 z_2 \frac{r x_1\left(x-a_1\right)\left(-y+b_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}} \\
& -z_2^2 \frac{Q_0 r\left(-y+b_0\right)\left(y-b_0\right)}{\left(\left(x-a_0\right)^2+\left(y-b_0\right)^2\right)^{\frac{3}{2}}}-z_2^2 \frac{Q_0 r}{\sqrt{\left(x-a_0\right)^2+\left(y-b_0\right)^2}} \\
& -z_2^2 \frac{r x_1\left(-y+b_1\right)\left(y-b_1\right)}{\left(\left(x-a_1\right)^2+\left(y-b_1\right)^2\right)^{\frac{3}{2}}}-z_2^2 \frac{r x_1}{\sqrt{\left(x-a_1\right)^2+\left(y-b_1\right)^2}}
\end{aligned}
$$

Now we are ready to feed some real numbers into it. Let $s=200$, $c=50$, $v=20$, $b=25$, $p=100$, $r=0.05$, $a_0=100$, $b_0=100$, $a_1=500$, $b_1=500$, $x=300$, $y=300$, $Q_0=1000$,

$\mu = 100$, $\sigma = 10$ $z = [-100, 1, 1]^T$. We have

$$z^T H z = -\frac{2500\sqrt{2}}{\sqrt{\pi}e^{4050}} + 10.0\sqrt{2} \approx 0 + 10.0\sqrt{2} > 0$$

Therefore, $H(\Pi^C)$ is not negative definite, and $\Pi^C$ is not concave. □

**Solution Approach: Q-search**

Because the profit is not concave on all decision variables, it is not viable to solve the problem with the augmented Lagrange multiplier method. To this end, we develop a custom solution algorithm named Q-search that can identify a **good** solution that is locally optimal. We find that when $Q_0$ is given, the problem is reduced to a simplified non-linear problem with the DC location $(x, y)$ as decision variables, which can be solved by the sequential least squares programming (SLSQP) algorithm. The proposed Q-search algorithm explores a wide range of possible $Q_0$s, evaluates the reduced non-linear problem via the SLSQP optimizer, stores the solution of each evaluation, and eventually outputs the one with highest profit as the solution we accept. In particular, the algorithm is described as follows:

**Step 1**: Determine the lower limit of $Q_0$, $Q_{lb} = \sum_{i \in \mathcal{I}} F_i^{-1}(\gamma)$, based on the service level constraint, and use it as a starting point of the search.

**Step 2**: Given the current $Q_0$, solve the simplified non-linear problem to obtain a profit, store tuple $(profit, Q_0, x, y)$ in the result set.

**Step 3**: Increment $Q_0$ by a pre-set step size, if $Q_0$ reaches $2Q_{lb}$, stop the search. Otherwise go to step 2.

**Step 4**: Return the entry with highest profit in the result set.

### 3.4.5. Choosing a Retailer as DC

Rather than building a new DC, an alternative is to set the DC at the closet retailer to the optimal DC location. Both options will incur additional cost, but in practice the cost of the retailer-as-DC option should be less expensive. We will explore this strategy via experiment in Section 3.6.

## 3.5. A COMPARISON OF DSM AND CSM

In this section, we provide a comparison of DSM and CSM under the first two types of transportation cost.

### 3.5.1. Case 1: quantity-dependent transportation cost

In Section 3.4.2 and Section 3.4.4, we have shown the optimal solutions for both DSM and CSM:

$$Q_i^* = \max\{F_i^{-1}(\gamma), F_i^{-1}(\frac{b-c-rs_i^q}{b-v})\} \qquad (DSM)$$
$$Q_0^* = \max\{\sum_i F_i^{-1}(\gamma), F_0^{-1}(\frac{b-c-r_0^q}{b-v})\} \qquad (CSM)$$

In DSM, the total optimal order quantity is $\sum_{i\in\mathcal{I}} Q_i$. Let $\beta_i = \frac{b-c-rs_i^q}{b-v}$, $\beta_0 = \frac{b-c-r_0^q}{b-v}$), then we can write the second expression of the max function in both solutions as $F_i^{-1}(\beta_i)$ and $F_0^{-1}(\beta_0)$. We first focus on the comparison when the optimal solutions are $F_i^{-1}(\beta_i)$ and $F_0^{-1}(\beta_0)$. This applies $\gamma \geq \beta_i$, and $\gamma \geq \beta_0$.

The normal distribution assumption is widely used in inventory theories when the demand comes from many different independent or weakly dependent retailers. In this chapter, we assume that the demand of each retailer follows normal distribution $N$ $(\mu_i, \sigma_i)$, then we can write the demand of retailer $i$ as $x_i = \mu_i + Z\sigma_i$ where Z is a standard normal random variable, and $\Phi(z) = Pr(Z < z)$ is the CDF of the standard normal random variable. Let $z_{\beta_i} = \Phi^{-1}(\beta_i)$ and $z_{\beta_0} = \Phi^{-1}(\beta_0)$, we can get

$$Q_i^* = \mu_i + z_{\beta_i}\sigma_i, Q_0^* = \mu_0 + z_{\beta_0}\sigma_0.$$

To simplify the objective function and let readers go through the results easily, we assume the unit transportation costs are the same for all retailers, $rs_i^q = rs^q, r_i^q = r^q \forall i \in \mathcal{I}$. With

the optimal order quantity, the profit gained from DSM is:

$$\Pi^D = (b - c - rs^q) \sum_{i \in \mathcal{I}} \mu_i - \sum_{i \in \mathcal{I}} G(Q_i^*)$$

where

$$\begin{aligned}
\sum_{i \in \mathcal{I}} G(Q_i^*) &= \sum_{i \in \mathcal{I}} [(c + rs^q - v) E[Q_i^* - x_i]^+ + (b - c - rs^q) E[x_i - Q_i^*]^+] \\
&= (c + rs^q - v) \sum_{i \in \mathcal{I}} Q_i^* - (c + rs^q - v) \sum_{i \in \mathcal{I}} \mu_i + (b - v) E \sum_{i \in \mathcal{I}} [x_i - Q_i^*]^+ \\
&= (c + rs^q - v) \sum_{i \in \mathcal{I}} Q_i^* - (c + rs^q - v) \sum_{i \in \mathcal{I}} \mu_i + (b - v) \sum_{i \in \mathcal{I}} \sigma_i R(z_{\beta_i})
\end{aligned}$$

We define $R(u) = Pr(z > u) = \int_u^\infty (x - u) \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx$, as the right-hand unit normal linear-loss integral.

Similarly, we can show the profit gained with CSM is:

$$\Pi^C = (b - c - r_0^q) \mu_0 - G(Q_0^*)$$

where

$$\begin{aligned}
G(Q_0^*) &= (c + r_0^q - v) E[Q_0^* - x_0]^+ + (b - c - r_0^q) E[x_0 - Q_0^*]^+ \\
&= (c + r_0^q - v) Q_0^* - (c + r_0^q - v) \mu_0 + (b - v) E[x_0 - Q_0^*]^+ \\
&= (c + r_0^q - v) Q_0^* - (c + r_0^q - v) \mu_0 + (b - v) \sigma_0 R(z_{\beta_0}).
\end{aligned}$$

For simplicity, assume that $\sigma_i$ is constant over all retailers, i.e., $\sigma_i = \sigma$, and $\sum_{i\in\mathcal{I}} Q_i^* = Q_0^*$. Then the difference of profit between CSM and DSM is:

$$\begin{aligned}
\Pi^C - \Pi^D &= (s-c-r_0^q-r^q)\mu_0 - G(Q_0^*) - (s-c-rs^q)\sum_{i\in\mathcal{I}}\mu_i + \sum_{i\in\mathcal{I}} G(Q_i^*) \\
&= (s-c-r_0^q-r^q)\mu_0 - (s-c-rs^q)\mu_0 - G(Q_0^*) + \sum_{i\in\mathcal{I}} G(Q_i^*) \\
&= (rs-r_0^q-r^q)\mu_0 + (c+rs^q-v)\sum_{i\in\mathcal{I}} Q_i^* - (c+rs^q-v)\mu_0 + (b-v)\sum_{i\in\mathcal{I}}\sigma_i R(z_{\beta_i}) \\
&\quad -(c+r_0^q-v)Q_0^* + (c+r_0^q-v)\mu_0 - (b-v)\sigma_0 R(z_{\beta_0}) \\
&= -r_0^q\mu_0 + (c-v)(\sum_{i\in\mathcal{I}} Q_i - Q_0) + rs^q\sum_{i\in\mathcal{I}} Q_i - r_0^q Q_0 \\
&\quad +(b-v)[\sum_{i\in\mathcal{I}}\sigma_i R(z_{\beta_i}) - \sigma_0 R(z_{\beta_0})] \\
&= (rs-r_0^q-r^q)\mu_0 + (c-v)(\sum_{i\in\mathcal{I}}\sigma_i z_{\beta_i} - \sigma_0 z_{\beta_0}) + rs^q\sum_{i\in\mathcal{I}}\sigma_i z_{\beta_i} - r_0^q\sigma_0 z_{\beta_0} \\
&\quad +(b-v)\sqrt{n}\sigma[\sqrt{n}R(z_{\beta_i}) - R(z_{\beta_0})]
\end{aligned}$$

Now, we can clearly see the impact of unit transportation cost on the profit. Based on the relationship between $rs^q$ and $r_0^q + r^q$, the benefit of using CSM may vary:

**Lemma 1.** *If $\delta = rs^q - r_0^q > 0$ and $\sqrt{n}z_{\beta_i} \geq z_{\beta_0}$, then the profit of CSM is always greater than the profit of DSM. The difference of $\Pi^C - \Pi^D$ is positive related to $\delta, n$, and $\sigma$.*

*Proof.* If $rs^q > r_0^q + r^q$: Let $\delta = rs^q - r_0^q - r^q > 0$, and recall that $\beta_i = \frac{b-c-rs^q}{b-v}$, and $\beta_0 = \frac{b-c-r_0^q}{b-v}$, we have

$$\begin{aligned}
&\beta_i < \beta_0 \\
\Rightarrow \quad &z_{\beta_i} < z_{\beta_0}, \\
\Rightarrow \quad &R(z_{\beta_i}) > R(z_{\beta_0}), \\
\Rightarrow \quad &(b-v)\sqrt{n}\sigma[\sqrt{n}R(z_{\beta_i}) - R(z_{\beta_0})].
\end{aligned}$$

If $\sqrt{n}z_{\beta_i} \geq z_{\beta_0}$, we can have

$$\begin{aligned} & \sqrt{n}\sigma_i(\sqrt{n}z_{\beta_i} - z_{\beta_0}) \geq 0, \\ \Rightarrow \quad & \textstyle\sum_{i\in\mathcal{I}} \sigma_i z_{\beta_i} - \sigma_0 z_{\beta_0} \geq 0. \end{aligned}$$

Therefore, when $\delta > 0$, and $\Pi^C - \Pi^D > 0$. The difference is related to the value of $\delta, n$, and $\sigma$. It implies that when the CSM is more profitable by 1) saving more from central transportation, 2) having more retailers, and 3) increasing demand variation at retailers. For a special case, when $\delta = 0$, meaning there is no difference on the unit transportation cost in two models, CSM is still a better choice because of the risk-pooling benefit. □

This case is similar to the previous work by Eppen and others (Cherikh, 2000; Eppen, 1979; Lin et al., 2001). However, when $\delta < 0$, CSM make more profit only if $(s - v)\sqrt{n}\sigma[\sqrt{n}R(z_{\beta_i}) - R(z_{\beta_0})] > (r_0^q + r^q - rs^q)\mu_0^*$, which can only be true when the number of retailers is very large.

When $\gamma > \beta_i$ and $\gamma > \beta_0$, we can find the optimal order quantity of DSM and CSM are:

$$Q_i^* = \mu_i + z_\gamma \sigma_i, Q_0^* = \sum_{i\in\mathcal{I}} Q_i.$$

Therefore, the difference of profit between CSM and DSM is

$$\begin{aligned} \Pi^C - \Pi^D &= (s - c - r_0^q - r^q)\mu_0 - G(Q_0^*) - (s - c - rs^q)\sum_{i\in\mathcal{I}} \mu_i + \sum_{i\in\mathcal{I}} G(Q_i^*) \\ &= -r^q\mu_0 + (rs^q - r_0^q)Q_0\sigma \\ &\quad +(b - v)\sqrt{n}\sigma R(z_\gamma)[\sqrt{n} - 1] \end{aligned}$$

When $rs^q > r_0^q + r^q$, clearly this difference is positive. It means CSM outperforms DSM.

### 3.5.2. Case 2: distance-dependent transportation cost

With distance-related transportation cost, we have shown the optimal solutions for DSM and CSM in the previous section.

$$Q_i^* = \max\{F_i^{-1}(\gamma), F_i^{-1}(\frac{b-c}{b-v})\} \qquad \forall i \in \mathcal{I} \qquad (DSM)$$
$$Q_0^* = \max\{\sum_i F_i^{-1}(\gamma), F_0^{-1}(\frac{c_u}{c_u+c_o}) = F_0^{-1}(\frac{b-c}{b-v})\} \qquad (CSM)$$

Letting $\frac{b-c}{b-v} = \beta, \sum_i \mu_i = \mu_0$, and $\sigma_i = \sigma, \forall i \in \mathcal{I}$, the difference of revenue and inventory costs between CSM and DSM is

$$\begin{aligned} CSM - DSM &= (s-c)\mu_0 - G(Q_0^*) - (s-c)\sum_{i\in\mathcal{I}}\mu_i + \sum_{i\in\mathcal{I}} G(Q_i^*) \\ &= \sum_{i\in\mathcal{I}} G(Q_i^*) - G(Q_0^*) \\ &= (c-v)(\sum_{i\in\mathcal{I}} Q_i - Q_0) + (b-v)[\sum_{i\in\mathcal{I}} \sigma_i R(z_i) - \sigma_0 R(z_0)]. \end{aligned}$$

If $Q_i^* = \sum_i F_i^{-1}(\gamma)$, $Q_0^* = \sum_i F_i^{-1}(\gamma)$, then $\sum_{i\in\mathcal{I}} Q_i = Q_0$. Because $\sigma_i$ is constant for all retailers, we have

$$\begin{aligned} \Pi^C - \Pi^D &= (b-v)[\sum_{i\in\mathcal{I}} \sigma_i R(z_i) - \sigma_0 R(z_0)] \\ &= (b-v)(n\sigma - \sqrt{n}\sigma)R(z) \\ &= (b-v)\sqrt{n}\sigma(\sqrt{n}-1)R(z) > 0 \end{aligned}$$

This shows that CSM benefits from risk pooling, and the profit is positively related to the variations of retailer demand. CSM also works better if the number of retailers increases. However the extra earnings from centralization may reduce if we consider the transportation costs. In case 2, the transportation cost depends on the total distance traveled. Using a central DC will incur longer distance compared with a direct shipping from the supplier.

The total transportation cost in CSM is

$$\sum_{i \in \mathcal{I}} (r_0^d d_0 + r_i^d d_i)$$

and the total transportation cost in DSM is

$$\sum_{i \in \mathcal{I}} (rs_i^d ds_i)$$

Since we want to maximize the total profit, the transportation part needs to be minimized. If we assume that $r_0^d = r_i^d = rs_i^d = r^d$, clearly DSM takes shorter distances than CSM, regardless of the DC location. The difference between CSM and DSM on transportation is

$$Trans(CSM - DSM) \quad = \quad \sum_{i \in \mathcal{I}} r^d (d_0 + d_i - ds_i) > 0.$$

It shows that in case 2, the performance of CSM may be worse than DSM. It depends on the difference of risk-pooling benefits and extra transportation cost.

### 3.5.3. Case 3: quantity-distance-dependent transportation cost

We will examine the performance of DSM and CSM under a quantity-distance-dependent transportation cost through computational experiments in next section.

## 3.6. COMPUTATIONAL EXPERIMENTS

In this section, we only focus on case 3 for the DSM and CSM, as we have compared the two models for case 1 and case 2 with numeric analysis in Section 3.5. We conduct all computational experiments on a workstation with a Intel Core(TM) i7-4770 CPU (3.40GHz), a 16GB RAM, and a 64-bit Windows 10 operation system. The employed solver is the optimization package of Scipy, an open-source Python library for scientific computation. In particular, we utilize the built-in SLSQP optimizer in Scipy as a building block of our solution approach. We implement both the data generation and the solution approach in Python.

Table 3.1: Data Specification

| Parameter | Notation | Setting |
|---|---|---|
| # Retailers | $n$ | [10, 100] by 10 |
| Selling Price | $s$ | \$200 |
| Unit Cost | $c$ | \$50 |
| Unit Salvage Value | $v$ | \$20 |
| Map Size | map_sz | $1000 \times 1000$ |
| Bundling Cost | $(p_0, p_i)$ | (\$200, \$100) |
| Unit Trans. Cost | $(r_0^{qd}, r_i^{qd})$ | (\$0.3, 0.5\$) |
| Demand Mean | $\mu_i$ | U[100, 200] |
| Demand Stdev | $\sigma_i$ | U[10, 20] |
| Pre-set Service Level | $\gamma$ | 0.3 |

### 3.6.1. Data Setting

We present the data setting of the experiment in Table 3.1, which includes ten instances that differ in network size measured by the number of retailers. The tested supply chain networks include 10, 20, up to 100 retailers, which can cover a wide range of real world situations. The demand of each retailer follows a normal distribution, with a mean in [100, 200] and a standard deviation in [10, 20]. We consider a square map with an area of $1000 \times 1000$ square miles; all location coordinates are randomly generated within the map. A fixed seed for the random number generator is used throughout the experiment for the purpose of consistency. In addition, we use $\gamma = 0.3$ for the tested ten instances.

### 3.6.2. DSM and CSM: a Comparison for Case 3

In this section ,we conduct a comparison study of case 3 for DSM and CSM. Letting $D_i$ denote the real demand retailer $i$ observes after the selling season starts, We define four metrics to measure the demand fulfillment rate:

$$\text{DSM:} \quad M_1 = \frac{1}{n}\sum_{i\in\mathcal{I}} \frac{Q_i}{\mu_i} \quad M_2 = \frac{1}{n}\sum_{i\in\mathcal{I}} \frac{Q_i}{D_i}$$

$$\text{CSM:} \quad M_3 = \frac{Q_0}{\sum_{i\in\mathcal{I}} \mu_i} \qquad M_4 = \frac{Q_0}{\sum_{i\in\mathcal{I}} D_i}$$

in which $M_1$ and $M_3$ reflect the ratio of order size to the expected demand, so they are referred to as expected fulfillment, while $M_2$ and $M_4$ measure the ratio of order size to the real demand, so we call them actual demand fulfillment.

Table 3.2: DSM Vs. CSM: Optimal Expected Profits

| | DSM | | CSM | | |
|---|---|---|---|---|---|
| n | $P_{DSM}$ (\$) | $M_1$ | $P_{CSM}$ (\$) | $M_3$ | $\Delta$ (\$) |
| 10 | 168879 | 94.6% | 170164 | 98.3% | 1285 |
| 20 | 354684 | 94.9% | 357101 | 99.3% | 2417 |
| 30 | 532537 | 94.8% | 536232 | 99.4% | 3695 |
| 40 | 688633 | 94.7% | 698628 | 99.1% | 9994 |
| 50 | 915006 | 94.6% | 924055 | 99.5% | 9049 |
| 60 | 1155430 | 94.6% | 1167504 | 99.7% | 12075 |
| 70 | 1274804 | 94.6% | 1294238 | 99.4% | 19434 |
| 80 | 1437268 | 94.5% | 1455582 | 99.5% | 18314 |
| 90 | 1612475 | 94.4% | 1629188 | 99.6% | 16713 |
| 100 | 1639821 | 94.3% | 1689876 | 99.2% | 50056 |

Table 3.2 presents the optimal profits, the service level indicator $M_1$, the running time for the CSM as well as the profit difference for both systems. In the table header, $P_{DSM}$ and $P_{CSM}$ refer to the optimal expected profits for DSM and CSM, respectively; also, we have $\Delta = P_{CSM} - P_{DSM}$, as shown in the last column. We find that for all ten instances, CSM outperforms DSM in terms of both profit and service level. Also, the profit gap $\Delta$ measures the additional profit the CSM earns compared to DSM. As setting a DC at a location incurs a cost, which can be taken into account at this point: if $\Delta$ is positive, and the amount is greater than the cost of adding a DC, then the CSM is a preferred choice. In this table, when the system size becomes larger, $\Delta$ is also larger, meaning that the CSM is more suitable for large supply chain networks. Meanwhile, the service level indicators, i.e., $M_1$ and $M_3$, of CSM are higher than those of DSM, meaning that the CSM system can deal with a large market.

Table 3.3 displays the profits calculated with randomly generated demands. As both the DSM and CSM are stochastic models that optimize expected profits, it is desirable to validate the models with random demand samples, which can simulate the demand observed at the beginning of the selling season. For our problem, we generate a random demand sample for each retailer using its density function, feed them into the profit function to replace $Q_i/\mu_i$,

Table 3.3: DSM Vs. CSM: Profits with Random Demand Samples

| | DSM | | CSM | | |
|---|---|---|---|---|---|
| n | $P_{DSM}$ ($) | $M_2$ | $P_{CSM}$ ($) | $M_4$ | $\Delta$ ($) |
| 10 | 172419 | 100.3% | 175578 | 102.9% | 3159 |
| 20 | 361815 | 100.7% | 366601 | 104.1% | 4786 |
| 30 | 542087 | 99.8% | 546837 | 103.3% | 4749 |
| 40 | 700733 | 99.4% | 711599 | 102.8% | 10866 |
| 50 | 931974 | 100.1% | 942193 | 103.8% | 10219 |
| 60 | 1174275 | 99.7% | 1183432 | 103.6% | 9157 |
| 70 | 1297803 | 99.5% | 1314126 | 103.2% | 16323 |
| 80 | 1459912 | 98.8% | 1477041 | 102.8% | 17128 |
| 90 | 1635625 | 98.3% | 1650692 | 102.6% | 15067 |
| 100 | 1669819 | 98.9% | 1719961 | 102.7% | 50142 |

and obtain the actual profit. We find that both CSM and DSM models perform as expected with these simulated demands. The results are in line with those in Table 3.2.

### 3.6.3. Retailer-as-DC: a Potentially Better Option

We discuss the option of choosing a retailer closest to the optimal DC location to serve as a temporary DC, which is more economic to build and maintain. We evaluate this retailer-as-DC option and compare it with the optimal solution. We study one instance (n=10) and plot the result in Figure 3.2, in which the horizontal axis shows the location of the optimal DC and locations of ten retailers, and the vertical axis refers to overall profit. In this case, as an alternative, we can build the DC at retailer 3, the one closest to the optimal DC, and live with a profit loss of $4,873. However, building a DC at the optimal location costs more than building it at a retailer. In practice, if this cost difference is over $4,873, retailer-as-DC wins. We expand this experiment to all ten instances. Table 3.4 shows the optimal DC location and the location of the closest retailer, and Table 3.5 displays a comparison of the two options in terms of expected and actual profits. Note that $P'_{CSM}$ refers to the profit obtained based on the location of the chosen retailer.

The first finding is that the system scale has no clear impact on the profit gap, which is mainly determined by the distance between the optimal DC and its closest retailer. Interestingly, for one instance (n=70), we obtain a negative profit gap, meaning that retailer-as-DC

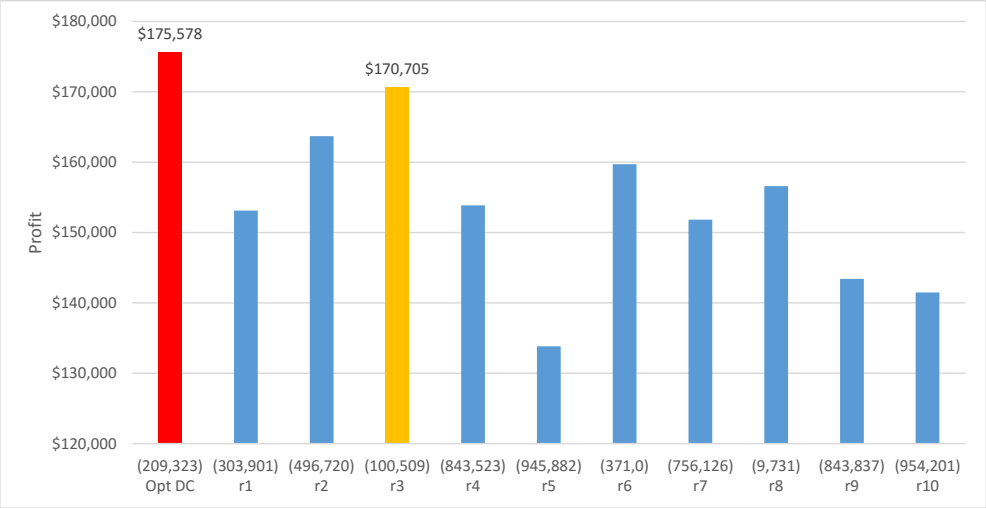



Figure 3.2: Optimal DC Vs. Retailer-as-DC on Instance n=10

Table 3.4: CSM Case 3: Optimal DC Vs. Retailer-as-DC

| n | $Q_0^*$ | $DC^*$ | Chosen Retailer | Dist. (mi) |
|---|---|---|---|---|
| 10 | 1467 | (209.6, 323.0) | (100, 509) | 215.9 |
| 20 | 2935 | (121.0, 144.1) | (90, 62) | 87.8 |
| 30 | 4421 | (496.7, 915.7) | (451, 905) | 46.9 |
| 40 | 5898 | (250.6, 205.0) | (206, 197) | 45.3 |
| 50 | 7435 | (847.2, 359.5) | (854, 345) | 16.0 |
| 60 | 9019 | (269.0, 360.0) | (308, 398) | 54.5 |
| 70 | 10576 | (633.8, 177.0) | (650, 144) | 36.8 |
| 80 | 11990 | (186.7, 702.3) | (200, 653) | 51.1 |
| 90 | 13272 | (735.0, 773.5) | (725, 792) | 21.0 |
| 100 | 14493 | (237.3, 228.5) | (251, 233) | 14.4 |

makes more actual profit with the given demand samples. This is because the solver uses expected demands to compute an optimal expected profit and a DC location, while the actual profit relies on simulated demands. Therefore, our models can provide valuable guidance but cannot guarantee an optimal decision provided with random demand samples, because of the nature of the problem. Overall, we highly recommend retailer-as-DC if the chosen retailer is close to the optimal DC location.

Table 3.5: CSM Profit Comparison: Optimal DC Vs. Retailer-as-DC

| | Expected Profit | | | Actual Profit | | |
|---|---|---|---|---|---|---|
| n | $P_{CSM}$ ($) | $P'_{CSM}$ ($) | $\Delta$ ($) | $P_{CSM}$ ($) | $P'_{CSM}$ ($) | $\Delta$ ($) |
| 10 | 170164 | 164893 | 5271 | 175578 | 170705 | 4872 |
| 20 | 357101 | 349242 | 7859 | 366601 | 359161 | 7441 |
| 30 | 536232 | 533343 | 2889 | 546837 | 544110 | 2727 |
| 40 | 698628 | 697833 | 794 | 711599 | 711230 | 369 |
| 50 | 924055 | 923623 | 432 | 942193 | 941918 | 275 |
| 60 | 1167504 | 1159312 | 8193 | 1183432 | 1174974 | 8458 |
| 70 | 1294238 | 1293777 | 461 | 1314126 | 1314151 | -26 |
| 80 | 1455582 | 1453261 | 2321 | 1477041 | 1474350 | 2690 |
| 90 | 1629188 | 1627938 | 1250 | 1650692 | 1649627 | 1066 |
| 100 | 1689876 | 1689753 | 123 | 1719961 | 1719627 | 334 |

Table 3.6: Sensitivity Analysis

| Parameter | Notation | Setting |
|---|---|---|
| Service Level Limit | $\gamma$ | [0.1, 0.9] by 0.1 |
| Map Size | map_sz | [200, 400, 800, 1000] |
| Unit Trans. Cost | $(r_0^{qd}, r_i^{qd})$ | [0.03,0.05], [0.05,0.05] |

### 3.6.4. Parameter Sensitivity Analysis

For sensitivity analysis, we select a mid-size instance with 40 retailers from the ten instances that are previously generated. For each experiment we only vary one parameter and keep the rest parameters unchanged. Table 3.6 displays three parameters including the pre-set service level, the map size, and the unit transportation cost, as well as their values in the experiment.

**Service Level**

Parameter $\gamma$ sets a lower limit for the system service level, i.e., $F_i(Q_i) \geq \gamma$; in other words, $Q_i$ takes a value that has a probability of $\gamma$ to satisfy a random demand with a CDF of $F_i$. Clearly, the higher $\gamma$ is, the larger $Q_i$ has to be. To figure out the impact of $\gamma$, we vary its value from 0.1 to 0.9 by 0.1, and display the result in Figure 3.3, from which we see a clear rising trend of service level and a descending trend of profit, for both DSM and CSM, as $\gamma$ grows. First, A higher $\gamma$ leads to a higher order quantity, which fulfills more demands that are unchanged during this experiment. This explains the rising of the expected demand

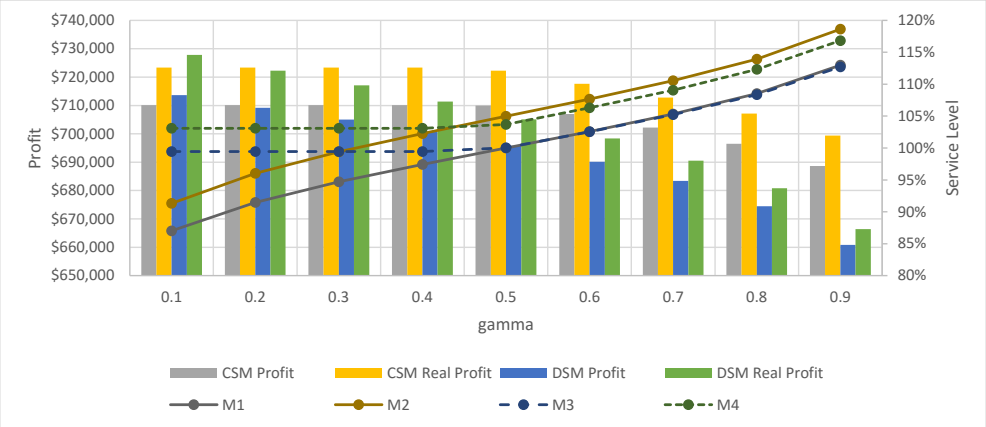



Figure 3.3: Impact of $\gamma$

fulfillment $M_1$ and $M_3$, and the actual demand fulfillment $M_2$ and $M_4$. Also, the overall profit declines in all scenarios, implying a counter-effect of $\gamma$. The reason is that when the order size exceeds demands due to a high $\gamma$, salvage occurs, lowering the profit.

We also find that CSM makes more profit than DSM when $\gamma$ is greater than 0.2, and the higher $\gamma$ is, the more benefit CSM creates, as the profit gap of CSM and DSM is larger. Meanwhile, we see that CSM presents a higher service level than DSM, for both expected and actual, even with a low $\gamma$. Overall, the pre-set service level does not make a positive effect for CSM, which already achieves a great performance with a $\gamma$ being 0.1; however, for DSM, a low $\gamma$ brings down both expected and actual demand fulfillment, which only boost up to 100% as $\gamma$ rises to 0.5.

**Unit Transportation Cost**

Our models assume that $r_0^{qd} \leq r_i^{qd}$; in other words, it is more cost-efficient to ship products to one destination (the DC) than to multiple destinations (retailers). We study the profit difference, the revenue difference, and the transportation cost difference of CSM and DSM with two modes of unit transportation cost: $(r_0^{pd}, r_i^{pd}) = (0.03, 0.05)$ and $(r_0^{pd}, r_i^{pd}) = (0.05, 0.05)$. Table 3.7 displays these differences under both expected and actual demand. With the first mode, when $r_0^{qd} < r_i^{qd}$, CSM incurs a higher expected transportation cost (see column three in top table) than DSM for seven out of ten instances; and this cost becomes

Table 3.7: Difference of CSM and DSM

| $(r_0^{pd}, r_i^{pd}) = (0.03, 0.05)$ | | | | | | |
|---|---|---|---|---|---|---|
| n | $\Delta_{rev}$ | $\Delta_{trans}$ | $\Delta_{real_rev}$ | $\Delta_{real_trans}$ | $\Delta_{profit}$ | $\Delta_{real_profit}$ |
| 10 | 2061 | 776 | 1641 | -1518 | 1285 | 3159 |
| 20 | 4391 | 1974 | 1861 | -2925 | 2417 | 4786 |
| 30 | 7055 | 3360 | 3143 | -1606 | 3695 | 4749 |
| 40 | 9879 | -115 | 5414 | -5452 | 9994 | 10866 |
| 50 | 12958 | 3909 | 5477 | -4742 | 9049 | 10219 |
| 60 | 15786 | 3712 | 6981 | -2176 | 12075 | 9157 |
| 70 | 18666 | -768 | 8108 | -8214 | 19434 | 16323 |
| 80 | 21542 | 3228 | 11234 | -5895 | 18314 | 17128 |
| 90 | 24542 | 7829 | 13639 | -1428 | 16713 | 15067 |
| 100 | 27153 | -22903 | 14262 | -35880 | 50056 | 50142 |
| $(r_0^{pd}, r_i^{pd}) = (0.05, 0.05)$ | | | | | | |
| n | $\Delta_{rev}$ | $\Delta_{trans}$ | $\Delta_{real_rev}$ | $\Delta_{real_trans}$ | $\Delta_{profit}$ | $\Delta_{real_profit}$ |
| 10 | 2096 | 1996 | 1221 | -682 | 101 | 1903 |
| 20 | 4394 | 2202 | 1711 | -2944 | 2192 | 4655 |
| 30 | 7057 | 3670 | 2993 | -1521 | 3388 | 4514 |
| 40 | 9951 | 6570 | 4604 | -13 | 3380 | 4617 |
| 50 | 12968 | 5379 | 5177 | -3844 | 7589 | 9021 |
| 60 | 15786 | 3712 | 6981 | -2176 | 12075 | 9157 |
| 70 | 18746 | 9094 | 7118 | -503 | 9652 | 7621 |
| 80 | 21596 | 10820 | 10394 | 222 | 10776 | 10171 |
| 90 | 24560 | 10809 | 13129 | 581 | 13751 | 12548 |
| 100 | 27556 | 19080 | 11772 | 279 | 8476 | 11493 |

higher when $r_0^{pd}$ is as high as $r_i^{pd}$, which can be seen from column three of the bottom table. However, the high revenue of CSM well compensates its high expense of transportation, leading to a higher profit compared to DSM, as shown in the last two columns. In addition, we should notice that CSM is less profitable with a high $r_0^{qd}$, which is illustrated by narrowed profit gaps in the bottom table.

**Map Size**

Intuitively, map size has a negative impact on transportation cost, since a larger map places the same number of facilities in a more scattered way, leading to increased distances among facilities. In this experiment, we raise the map from an area of $200 \times 200$ to $1000 \times 1000$.

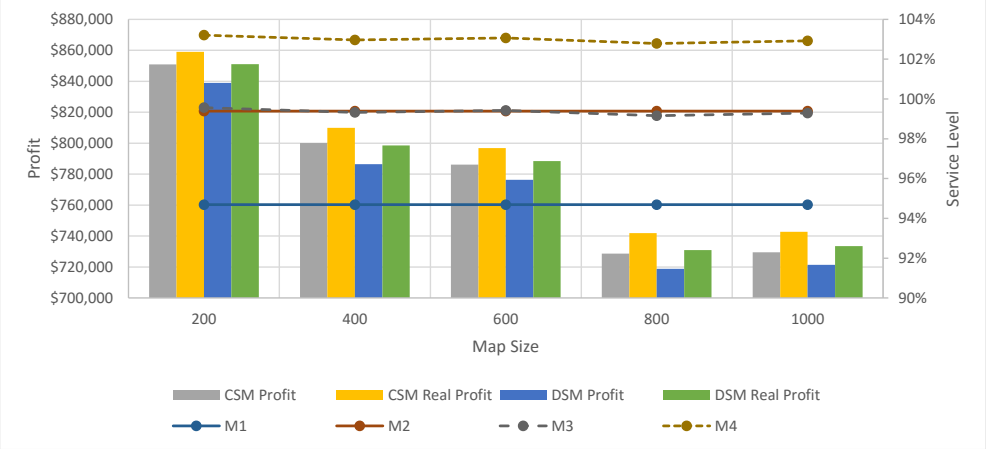



Figure 3.4: Impact of Map Size

The result shown in Figure 3.4 verifies this intuition. Another straightforward observation is that map size is unrelated to the service level.

## 3.7. CONCLUSION

In this chapter, we consider a supply chain network where the sole supplier has the option of centralizing the ordering of a product for a number of retailers whose demand fluctuates randomly. In this network, each retailer operates in a newsvendor setting, i.e., selling a product to its customers in a single period. We discuss two models including DSM and CSM with three modes of transportation cost. For both models, the goal is to maximize the overall system profit. The DSM essentially consists of $n$ independent newsvendor problems, while the CSM introduces a DC that takes orders from all retailers. It is thus critical to determine the location of the DC, since all orders will be firstly shipped to the DC, and then dispatched to retailers. When the transport cost is only dependent on the distance between two end points or the order size, we provide closed-form solutions for both models. However, when the transportation cost depends on both distance and order size, the CSM problem becomes a non-linear programming model, for which a closed-form optimal solution does not exist. We propose a Q-search algorithm to identify a high-quality solution which is practically acceptable. We validate our models through a series of experiments with the summarized

key findings as follows. First, a comparison of DSM and CSM for case 3 shows that CSM is better for large networks in terms of both profit and service level. Also, our models are competent with simulated demands, which demonstrates their values in practice. Besides, the option of retailer-as-DC presents a great performance and is thus highly recommended if the chosen retailer is close to the optimal DC. Lastly, our sensitivity analysis shows that 1) a pre-set service level limit does not pose a positive effect on CSM; 2) the unit shipping cost ($r_0^{pd}, r_i^{pd}$) also affects the profitability of CSM, and a lower $r_0^{pd}$ helps CSM generate more profit due to more savings on shipping cost; 3) our models handles problems with various map sizes quite well. In all, CSM presents a number of advantages over DSM, and is applicable to more practical scenarios.

## CHAPTER 3 REFERENCES


Cachon, G. (2001). Stock wars: Inventory competition in a two-echelon supply chain with multiple retailers. Operations Research, 49(5):658–674.

Cetinkaya, S. and Lee, C.-Y. (2000). Stock replenishment and shipment scheduling for vendor managed inventory systems. Management Science, 46:217–232.

Cetinkaya, S., Mutlu, F., and Lee, C.-Y. (2006). A comparison of outbound dispatch policies for vendor-managed inventory systems. European Journal of Operational Research, 171:1094–1112.

Chang, P. and Lin, C. (1991). On the effect of centralization on expected costs in a multi-location newsboy problem. Journal of Operational Research Society, 42:1025–1030.

Cherikh, M. (2000). On the effect of centralization on expected profits in a multi-location newsboy problem. Journal of Operational Research Society, 51:755–761.

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

Hartman, B. C. and Dror, M. (2005). Allocation of gains from inventory centralization in newsvendor environments. IIE Transactions, 37:93–107.

Ji, X. and Shao, Z. (2006). Model and algorithm for bilevel newsboy problem with fuzzy demands and discounts. Applied Mathematics and Computation, 172:163–174.

Khouja, M. (1999). The single-period (newsvendor) problem: literature review and suggestions for future research. Omega: The International Journal of Management Science,

27:537–553.

Lee, C. C. and Chu, W. H. J. (2005). Who should control inventory in a supply chain? European Journal of Operational Research, 164:158–172.

Lee, H. L. and Whang, S. (2000). Information sharing in a supply chain. International Journal of Manufacturing Technology and Management, 1:79–83.

Lin, C., Chen, C., and Hsieh, H. (2001). Effects of centralization on expected profits in a multi-location newsboy problem. Journal of Operational Research Society, 52:839–841.

Netessine, S. and Rudi, N. (2004). Supply chain structures on the internet. In Simchi-Levi, D., Wu, S. D., and Shen, Z. M., editors, Handbook of Quantitative Supply Chain Analysis: Modelling in the E-Business Era. Springer.

Shen, Z.-J., Coullard, C., and Daskin, M. S. (2003). A joint location-inventory model. Transportation Scinece, 47:40–55.

Silver, E. A., Pyke, D. F., and Peterson, R. (1998). Inventory management and Production Planning and Scheduling. John Wiley and Sons, Inc.

Slikker, M., Fransoo, J., and Wouters, M. (2005). Cooperation between multiple newsvendors with transshipments. European Journal of Operational Research, 167:370–380.

Tancrez, J., Lange, J., and Semal, P. (2012). A location-inventory model for large three-level supply chains. Transportation Research Part E, 48:485–502.

Toptal, A. and Cetinkaya, S. (2006). Contractual agreements for coordination and vendor-managed delivery. accepted Naval Research Logistics.

# CHAPTER 4

# SELECTIVE NEWSVENDOR PROBLEM WITH DEPENDENT LEADTIME AND JOINT MARKETING DECISIONS

## 4.1. INTRODUCTION

Competitions in global markets, the introduction of products with shorter life cycles, and heightened customer expectation have driven enterprises to pay more attention to supply chain operations. Enterprises combine and coordinate different departments such as operations, marketing, and sales to be more responsive and cost-efficient. The development of coordination between operations and marketing has been witnessed by both researchers and practitioners, since the early 2000s. Explanatory, empirical and analytical studies between the two business functions are expected to provide the foundation for new frameworks and theories.

In this chapter, we investigate a joint management effort between operations and marketing. We consider a newsvendor-like context including a three-level network with two suppliers, a company (buyer) and geographically distributed retailers/customers. We assume that the distant supplier (the primary supplier) has unlimited capacity, and there are no defects or damages during shipping, meaning that the buyer can receive 100% of its orders from the supplier. The second supplier, referred to as the emergency local supplier, can also provide the same product but with a higher production cost. The company has a certain number of sales agents in the marketing department. The abilities of agents to promote products are different. Also, each sales agent can only handle a certain number of customers due to limited work hours and energy. We assume that customer demands follow a Poisson distribution. The actual demands are unknown until the selling season begins. After placing

an order, each customer has an expected time frame of order delivery, referred to as the waiting time; in other words, the waiting time is the number of days a customer is willing to wait. Based on the different ordering quantities, the waiting of customers can be varied even for the same product. It often happens in the selective distribution network that the manufacturer can choose the most appropriate or best-performing outlets and focus effort on them. And in an exclusive distribution system, companies usually launch products, such as electronic devices and lines of clothing, via exclusive sale or limited edition at selected retailers as test markets. For example, Apple first launched iPhone 6s in United States, UK, France, Germany, Canada, Australia, Japan, Hong Kong, and Singapore on September 9th, 2015. The second launch wave started on October 9th, and the device was available in more than 40 additional countries.

At the beginning of a selling season, the company needs to make two decisions: (i) which customers should be selected to serve and how to assign them to sales agents; (ii) how many products need to order from the supplier. Traditionally, a company makes two decisions separately for marketing and operations. However, conflicts between the two departments will arise when the "supply" of operations does not meet the "demand" of marketing. Specifically, conventional theories of aggregate planning assume that the demand forecast is a mission of marketing and this information flows to operations department to make production and inventory decisions. On the other hand, sales agents in marketing will push for as many orders as they can, without worrying about the capacity, inventory level, and other inside information from the operations. As a consequence, if demands are higher than the capacity of inventory and production, the company may lose profit and customer satisfaction may become worse. In this chapter, we propose to integrate decisions of the two functional departments and study how this joint decision-making pattern benefits a company.

Another factor we take into account is lead time. Instead of using a fixed lead time, we consider a lead time that is linearly dependent on the order quantity: the more the company orders, the longer it has to wait. This is a reasonable constraint due to a capacitated supplier.

Meanwhile, with a larger order size, the company can serve more customers, but the lead time increases accordingly, which causes the company to lose some customers. In other words, some customers are willing to wait longer for the product delivery, whereas others would take their business elsewhere if the lead time exceeds their waiting times. As such, the company needs to make a trade-off to balance customer satisfaction and profit.

We summarize the contributions of this chapter as follows:

- We define two novel models, including all-or-nothing selective newsvendor problem (AON-SNP) and selective newsvendor problem with dependent lead time and price related demands (DL-SNP). We investigate two types of demand patterns, associated with dependent lead time and service level considerations. The proposed models are applicable to numerous areas such as the fashion, furniture, and electronic industries.

- We develop an efficient solution algorithm, referred to as R-Search, to identify an optimal solution for the DL-SNP model. The algorithm employs optimization strategies that are mathematically proven for performance improvement.

- We conduct a series of experiments to validate the DL-SNP model and the R-search algorithm. The results demonstrate that our model well explains various behavior for all involved parties in the network, and provide guidance on intelligent decision making for a company.

The rest of this chapter is structured as follows. We deliver a literature review in Section 4.2, and provide model formulations and related solution approaches in Section 4.3. In Section 4.4, we present the results of a series of computational experiments and the marginal insights. We conclude the chapter with some key findings and future directions in Section 4.5.

## 4.2. LITERATURE REVIEW

In this section, we mainly review prior studies in three areas, including selective newsvendor models, joint decision models of marketing and inventory, and the role of service level constraint in newsvendor problems.

### 4.2.1. Selective Newsvendor Models

It is common to assume that excess demand is backordered in classic inventory models. However, Gruen et al. (2002) show that only 15% of the customers still decide to place an order for a product that is known to be temporarily out of stock. Thus, most of the excess demand become lost sales rather than backordered in a retail environment. Many prior studies have contributed to the inventory theory regarding lost sales, and we recommend Bijvank and Vis (2011) as a review. Besides, Tsay et al. (1999) provide a review of supply chain contracts and competitive inventory management in a single-period newsvendor setting. In most of the inventory theory papers, the lead time is assumed to be a fixed or a random integral multiple of the review period. Unfortunately, these assumptions are not satisfied in many practical settings. Bijvank and Johansen (2012) consider a periodic review lost sales inventory problem with Poisson demand and constant lead times of any length. Taaffe et al. (2008) consider a newsvendor problem with $n$ potential markets. They develop a model that maximizes the profit, and provide an analysis of the influence of marketing efforts (promotions and advertisements) on the mean and variance of demand. Their model assumes given lead times and normally distributed demand, and also defines a binary variable to capture the market selection decision. Based on the prior model, Chahar and Taaffe (2009) solve a demand selection problem with all-or-nothing orders. They use a conditional value-at-risk approach that allows a decision maker to control the number of demands considered in the overall procurement policy. Also, they formulate a model with the flexibility provided to a firm that wants to balance the risk tolerance versus achievable profits.

In our problem, we use a profit maximization approach based on a newsvendor-type model, in which customers are selected and assigned to sale agents. We consider two demand

patterns combined with service level constraints. Besides, we also examine lead time as a linear function of order quantity, which is more reasonable for real world cases. The cost of lost sales (or cost of second local supplier) is also discussed in the model formulation.

### 4.2.2. Marketing and Inventory Joint Decision Models

Most marketing and inventory integrated models consider price as the primary factor that affects demand quantity and therefore influences order/production quantity or purchasing policies. Researchers have been studying the issue of joint production and pricing decision since the late 1950s. In the literature, most joint production and pricing models focus on the mathematical analysis to determine the optimal pricing and production(or replenishment) planning decisions for a single product over a single period. We recommend (Tang, 2010) as a review of marketing-operations interface models. Note that all previous models assume that the pricing decision can be made after demand (or supply) is realized. Recently, researchers consider other factors that will influence demand such as booking discounts and trade credits. Ouyang et al. (2009) propose an integrated supplier-buyer inventory model with the assumption that market demand is sensitive to the retail price and the payment delay period is dependent on the order quantity. Through an analysis of the total channel profit function, they develop a solution algorithm to determine the optimal order quantity, retail price and the number of shipments from the supplier to the customer. In this study, the supply chain includes a single supplier and a single customer, while our problem considers a single supplier and multiple customers to serve. Carr and Lovejoy (2000) present an inverse newsvendor problem to determine the optimal market selection that maximizes the expected profit, where the order quantity is normally distributed with a known mean and standard deviation. In our paper, both customer selection and order quantity are decision variables.

The sales force possess marketing knowledge that is critical for a broad range of decisions. However, researchers have not paid enough attention in the field of marketing-operation joint decisions until recently. Chen (2005) focuses on how incentives affect the sales force and pro-

duction/inventory decisions under the setting of a single supplier and a single agent. The objective of their model is to maximize a newsvendor profit by determining the level of incentives and order quantity. In their model, customer demands are influenced by three factors: the agent's selling effort, the marketing condition, and random noise. A key assumption is that the firm must make its production decision (order quantity) before observing the total sales. In our problem, we consider the customer selection and assignment based on different sales capabilities of agents, and the two decisions, customer selection and order quantity, are determined at the same time. Also, since the lead time of ordering is linearly dependent on the total quantity, the more the company orders from its supplier, the longer customers have to wait. Customers will leave if the lead time is longer than their waiting times. This characteristic makes our newsvendor model more difficult, which calls for a new solution approach.

### 4.2.3. Service Level Constraint

In practical situations, it is difficult to measure the shortage cost caused by the loss of potential future customers or orders. Instead of explicitly expressing the shortage cost in the objective function, some studies choose to introduce a service level constraint to reduce the chance of shortage. Aardal et al. (1989) introduce two metrics of service level: the fraction of demand covered from stock and the average number of shortage occurrences per year. They study the relation between shortage cost and service level based on a continuous review inventory system. Chen and Krass (2001) discuss a minimal service level constraint which requires the company to meet a certain service level in each period. Chen and Chuang (2000) propose an extended newsvendor problem by considering an average shortage-level constraint. Their results show that the greater the demand, the earlier an order should be placed. In our model, we utilize the fraction of selected customers and the capacity of each agent as factors to define the service level. Our choice is reasonable since it occurs in practice that only a fraction of customers can be served. Thus, the company can make strategic analyses regarding its profit, reputation, and customer relationship.

## 4.3. PROBLEM DESCRIPTION AND MODEL FORMULATION

In this section, we formulate two selective newsvendor models based on two demand patterns with the following notations.

**Notation**

**Sets and Indices:**

$\mathcal{I}$: set of sales agents, $\mathcal{I} = \{i : 1, 2, ...I\}$

$\mathcal{J}$: set of customers, $\mathcal{J} = \{j : 1, 2, ...J\}$

**Properties of Customer:**

$\mu_j$: expected demand quantity of customer $j$, $j \in \mathcal{J}$

$w_j$: waiting time of customer $j$, $j \in \mathcal{J}$

**Properties of Sales Agent:**

$p_{ij}$: effort of agent $i$ successfully getting an order from customer $j$ , $i \in \mathcal{I}$, $j \in \mathcal{J}$

$g_i$: maximum number of customers which agent $i$, $i \in \mathcal{I}$ can handle

**Time and Costs:**

$a$: unit production time of supplier

$b$: fixed shipping time from the supplier to the company

$c$: production cost

$e$: salvage price, assume $e < c$

$s$: shortage cost/second supplier production cost, assume $s > c$

$\lambda$: sensitivity of customer demand to price reduction

$r$: base price (suggested price)

$\alpha$: minimum service level

**Decision Variables:**

$Q$: order quantity

$R$: final selling price

$Y_j$: number of products sold to customer $j$, $j \in \mathcal{J}$

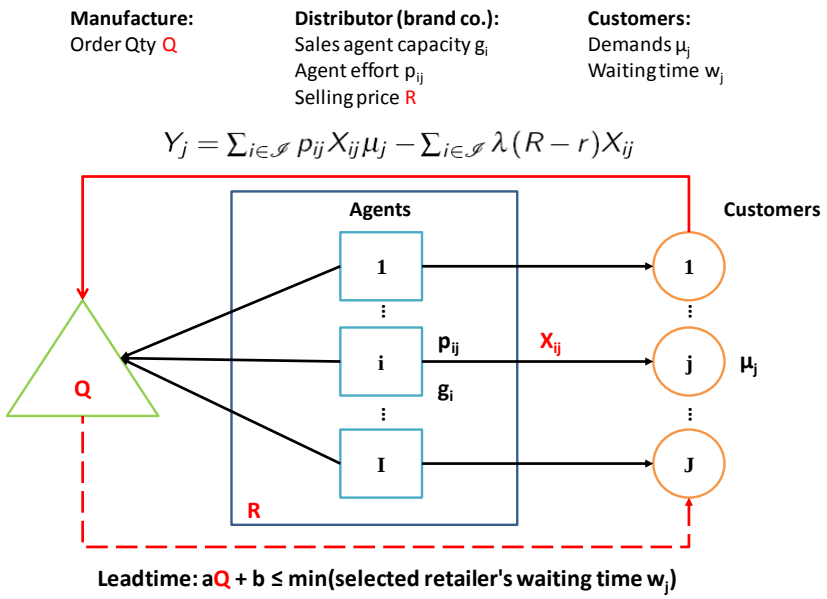



Figure 4.1: Big Picture of the Selective Newsvendor Network

$$X_{ij} = \begin{cases} 1, & \text{if customer } j \text{ is assigned to salesman } i, \\ 0, & \text{otherwise.} \end{cases} \qquad i \in \mathcal{I}, j \in \mathcal{J}$$

### 4.3.1. All-or-Nothing Selective Newsvendor Problem (AON-SNP)

We consider a joint decision model for a two-stage supply chain network, including a supplier, a company, and its customers. The buyer company has a marketing department with a group of sales agents that have different abilities and skills of product promotion. Whether the company can secure customer orders depends on the marketing abilities of agents, which are measured by the success rate to obtain orders. Besides, each agent can only handle a certain number of customers due to limited working hours and personal experience. We assume that customer demands follow a Poisson distribution, and are uncertain until a selling season starts.

Sometimes customers are willing to accept partial orders. However, there are many situations in real life that customers do not accept partial orders, and they simply take the business away if their demands cannot be fully satisfied. This phenomenon usually happens in the retail industry, in particular for the highly customized products, like large conference/wedding purchasing and school uniforms. We refer to this type of customer demand as all-or-nothing (AON) demand. In other words, an order needs to be satisfied at the expected level or not at all.

At the beginning of a selling season, the company desires to maximize the expected profit by making a series of decisions including the strategy of customer selection, a proper selling price, and order quantity from the oversea supplier. A customer has two attributes: the expected demand $\mu_j$, i.e., mean of the Poisson distribution and waiting time $w_j$. Each agent $i$ in the company has influence on order quantities of customers, denoted by $(p_{ij}) \in [0.8, 1.2]$. Also, every agent also has a limited number of customers that he/she can handle, denoted by $g_i$. For the AON-SNP model, the company has two decisions to make: (i) the assignment $X_{ij}$ of customers to each agent, which equals 1 if customer $j$ is assigned to agent $i$ and 0 otherwise; (ii) the order quantity $Q$. The question we attempt to answer is how does this demand pattern influence decision making and the expected profit. As such, we formulate the AON-SNP model as follows.

**Objective (AON-SNP):**

$$\max \quad f(X_{ij}, Q) = r \sum_{j \in \mathcal{J}} \sum_{i \in \mathcal{I}} p_{ij} X_{ij} \mu_j - cQ + e(Q - \sum_{j \in \mathcal{J}} \sum_{i \in \mathcal{I}} p_{ij} X_{ij} \mu_j)$$

**Subject to:**

$$\textbf{Agent Capacity} \qquad \sum_{j\in\mathcal{J}} X_{ij} \leq g_i, \qquad \forall i \in \mathcal{I}. \tag{4.1}$$

$$\textbf{Customer Assignment} \qquad \sum_{i\in\mathcal{I}} X_{ij} \leq 1, \qquad \forall j \in \mathcal{J}. \tag{4.2}$$

$$\textbf{Production sold} \qquad \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} p_{ij} X_{ij} \mu_j \leq Q, \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}. \tag{4.3}$$

$$\textbf{Integrality} \qquad Q > 0, X_{ij} \in \{0, 1\}. \tag{4.4}$$

Constraint set 4.1 specifies the capacity for each agent. Constraint set 4.2 ensures that each customer can either be assigned by one agent or not served at all. In particular, with the AON demand, the number of products sold to a selected customer $j$ equals the demand of $j$. If the company does not have enough inventory as required by customer $j$, he/she will not be served. This constraint leads to a problem that the total fulfilled demand will never exceed the order quantity $Q$, as shown in Constraint 4.3. Therefore, shortage never occurs. The formulated model is a mixed integer linear programming (MILP) problem, which can be solved by existing MILP solvers such as IBM CPLEX.

### 4.3.2. Selective Newsvendor Problem with Dependent Lead Time and Price Related Demands(DL-SNP)

The AON-SNP model considers one kind of demand pattern in the real world. However, sometimes customers are flexible to accept partial orders, or companies can use a second supplier to fill the shortage with higher costs. Choosing a second or emergency provider is costly but worthwhile if it can help the company keep loyalty customers. As opposed to the AON-SNP model, this section explores a model that allows the main supplier to take partial order quantities of customers. Moreover, we try to capture the influence of selling price on demands. In particular, we formulate a selective newsvendor model with dependent lead time and price-related demand, as shown in Figure 4.1. Concretely, the product quantity sold to a customer $j$, i.e., $Y_j$, is negatively affected by the selling price $R$, and a price scale

$\lambda$ is introduced to measure the sensitivity of customers to price reduction. For instance, $\lambda = 1.0$ means that if the price rises/reduces by \$1, the demand will decrease/increase by 1 unit, accordingly. Each customer $j$ has an expectation of delivery time $w_j$. When the waiting time is greater than $w_j$, the customer will leave with nothing bought. The waiting time is determined by the production time at supplier, $a \times Q$, and the time consumed on transportation $b$. Therefore, the company needs to make three kinds of decisions to achieve a maximal profit, including 1) the selling price $R$, 2) the order quantity $Q$ from its oversea supplier, and 3) the assignment of a selected customer $j$ to an agent $i$, i.e., $X_{ij}$. To formulate the new model, we first define $Y_j$:

$$Y_j = \sum_{i \in \mathcal{I}} p_{ij} X_{ij} \mu_j - \sum_{i \in \mathcal{I}} \lambda (R - r) X_{ij} \qquad \forall j \in \mathcal{J}$$

We then formulate the DL-SNP model:

**Objective (DL-SNP):**

$$\max \quad f(X_{ij}, Q, R) = \sum_{j \in \mathcal{J}} R Y_j - cQ + e(Q - \sum_{j \in \mathcal{J}} Y_j)^+ - s(\sum_{j \in \mathcal{J}} Y_j - Q)^+ \tag{4.5}$$

**Subject to:**

**Customer Service Level**
$$J - \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} X_{ij} \leq J(1-\alpha), \tag{4.6}$$

**Sales Agent Capacity**
$$\sum_{j \in \mathcal{J}} X_{ij} \leq g_i, \qquad \forall i \in \mathcal{I} \tag{4.7}$$

**Lead Time**
$$aQ + b \leq \sum_{i \in \mathcal{I}} w_j X_{ij} + M(1 - \sum_{i \in \mathcal{I}} X_{ij}), \qquad \forall j \in \mathcal{J} \tag{4.8}$$

**Customer Assignment**
$$\sum_{i \in \mathcal{I}} X_{ij} \leq 1, \qquad \forall j \in \mathcal{J} \tag{4.9}$$

**Production Sold**
$$Y_j = \sum_{i \in \mathcal{I}} p_{ij} X_{ij} \mu_j - \sum_{i \in \mathcal{I}} \lambda(R-r) X_{ij} \qquad \forall j \in \mathcal{J} \tag{4.10}$$

**Integrality**
$$X_{ij} \in \{0, 1\}, \tag{4.11}$$

**Non-negativity**
$$Q, R, Y_j \geq 0. \tag{4.12}$$

As opposed to AON-SNP, we add a few more constraints to meet the needs of the new model. First, we introduce Constraint 4.6 of customer service level to ensure that a certain fraction of customers will be selected. A minimum service level $\alpha$ is given to enforce the constraint. Second, we add a constraint set 4.8 to ensure that the lead time should not exceed the expected waiting times of all selected customers. We employ the big-M method to properly formulate this constraint: if customer $j$ is selected, then $\sum_{i \in \mathcal{I}} X_{ij} = 1$, and the constraint becomes $aQ + b \leq \sum_{i \in \mathcal{I}} w_j X_{ij}$, which sets an effective bar for the lead time; otherwise $\sum_{i \in \mathcal{I}} X_{ij} = 0$, and the constraint becomes $aQ + b \leq M$, which holds anyway. The product quantity sold to a customer $j$, $Y_j$, is another key difference. Besides the expected demand and the efforts of a sales agent, we integrate the factors of selling price, base price, and customers' sensitivity to price reduction into the formulation of $Y_j$. In particular, the company can sell more products to a selected customer if 1) the customer has a high demand in the past, indicating a high demand mean, or 2) the assigned agent does a good job to convince the customer to buy more. On the contrary, the customer will order less with a higher selling price $R$.

The DL-SNP model is a mixed integer non-linear problem, which can not be solved by existing solvers such as CPLEX. We develop a solution approach named R-search, which can effectively and efficiently identify an optimal solution.

**Solution Approach: R-search Method**

The proposed R-search method is based on a fact that when $R$ is given, the objective function $f(X_{ij}, Q, R)$ becomes $f(X_{ij}, Q)$, which is a MILP problem solvable by CPLEX. The key is to determine a proper searching strategy to reduce the running time of the algorithm. Clearly, the larger the searching range is, the more MILP problems need to be solved, which hurts the performance.

To facilitate the presentation of the upcoming lemma and its proof, we define a list of quantities denoted by $K^{max} = \{\max\{p_{ij}\mu_j \mid i \in \mathcal{I}\} \mid j \in \mathcal{J}\}$. Essentially, $K^{max}$ can be obtained by selecting the largest value of $p_{ij}\mu_j$ for each customer $j$ and add it into the list. Now let $K^{max}_{sorted}$ denote a sorted $K^{max}$ in ascending order, and $K^{max}_{sorted}[i]$ refer to the $i$th element in $K^{max}_{sorted}$.

**Lemma 2.** *The upper bound of $R$ can be determined as follows:*

$$R_{ub} = \frac{1}{\lambda} K^{max}_{sorted}[J - \lceil J\alpha \rceil + 1] + r$$

*Proof.* If a customer $j$ is not selected, then $\sum_{i\in\mathcal{I}} X_{ij} = 0$, and $Y_j = 0$, meaning that $j$ has no impact on $R$, as no products are sold to this customer. If $j$ is selected, then $\sum_{i\in\mathcal{I}} X_{ij} = 1$, and we can rewrite the Constraint 4.10 as:

$$\begin{aligned} R &\leq \frac{\sum_{i\in\mathcal{I}} p_{ij} X_{ij} \mu_j}{\lambda \sum_{i\in\mathcal{I}} X_{ij}} + r \\ &= \frac{1}{\lambda} \sum_{i\in\mathcal{I}} p_{ij} X_{ij} \mu_j + r \\ &\leq \frac{1}{\lambda} \max\{p_{ij}\mu_j \mid i \in \mathcal{I}\} + r \end{aligned}$$

Note that $\max\{p_{ij}\mu_j \mid i \in \mathcal{I}\}$ has been pre-stored in $K^{max}_{sorted}$. To satisfy the service level constraint, at least $\lceil J\alpha \rceil$ customers will be selected. To ensure that Constraint 4.10 holds for each selected $j$, $R$ is required to be less than the smallest $p_{ij}\mu_j$ among all selected customers. Clearly, that would be the $(J - \lceil J\alpha \rceil + 1)$th element in $K^{max}_{sorted}$. Therefore, we have $R \leq R_{ub} = \frac{1}{\lambda} K^{max}_{sorted}[J - \lceil J\alpha \rceil + 1] + r$. □

Lemma 2 identifies the upper bound of the selling price $R$, which serves as an accurate starting point for the search. The lemma shows that any price beyond that point is a violation of Constraint 4.10.

**Lemma 3.** *Without any constraints, the problem can achieve the optimal solution if and only if there is neither a salvage cost nor a shortage cost, i.e.,* $Q = \sum_{j\in\mathcal{J}} Y_j$.

*Proof.* Assume $R$ is given, let $\sum_{j\in\mathcal{J}} Y_j = D$ to represent the total product quantity sold, then the objective function becomes

$$f(D, Q) = RD - cQ + e(Q - D)^+ - s(D - Q)^+$$

Since $\min(Q, D) = D - (D - Q)^+ = Q - (Q - D)^+$, we can transform $f(D, Q)$ as:

$$\begin{aligned} f(D, Q) &= RD - c\left(\min(Q, D) + (Q - D)^+\right) + e(Q - D)^+ - s(D - Q)^+ \\ &= RD - c\left(D - (D - Q)^+ + (Q - D)^+\right) + e(Q - D)^+ - s(D - Q)^+ \\ &= (R - c)D - G(D, Q) \end{aligned}$$

in which $G(D, Q) = (s - c)(D - Q)^+ + (c - e)(Q - D)^+$. Now we show that $f$ is maximized if and only if $Q = D$; as such, we have $G(D, Q) = 0$, and $f_{max} = (R - c)D$. There are two cases:

**Case I**: $D > Q$. Without loss of generality, let $D = Q + \Delta$, $\Delta > 0$ then $f_{max} = (R - c)D = (R - c)(Q + \Delta)$, and $f(Q + \Delta, Q) = (R - c)(Q + \Delta) - (s - c)\Delta > 0$. Therefore, $f_{max} - f(Q + \Delta, Q) = (s - c)\Delta$. In the real world, the shortage cost should be greater

than the production cost, i.e., $s > c$, because once shortage occurs, the company will find another supplier to fill the additional demand, which increases the cost. Therefore, instead of spending more money working with another supplier, it is a better to satisfy the additional demand $\Delta$ by the original supplier, which then brings up the order quantity $Q$ by $\Delta$ units so that $D = Q$ once again.

**Case II**: $D < Q$. Similarly, let $D = Q - \Delta$, $\Delta > 0$ then $f_{max} = (R-c)D = (R-c)(Q-\Delta)$, and $f(Q+\Delta, Q) = (R-c)(Q-\Delta)-(c-e)\Delta$. Therefore, $f_{max} - f(Q-\Delta, Q) = (c-e)\Delta > 0$, which also makes sense in the real world, as the salvage price is lower than the production cost, i.e., $e < c$. It is clearly better to reduce the order quantity $Q$ by $\Delta$ units if possible.

In conclusion, the optimality condition for the profit function is that $Q = D$, meaning that neither shortage nor salvage will occur. □

Lemma 3 shows the optimality condition of the profit function without any constraint. In other words, $Q$ and $D$ are free to take any positive values. However, for our problem, $Q$ is limited by the lead time, i.e., $aQ+b$ is less than the shortest waiting time among all selected customers. It is thus possible that when $Q$ reaches its limit, $D$ can still be increased to raise the overall profit, leading to a shortage. To achieve that, $R$ needs to be further reduced to increase the demand. In particular, we formulate the upper limit of $Q$:

**Lemma 4.** *The upper limit of $Q$ is: $Q_{ub} = \frac{\min\{w_j | j \in \mathcal{J}\} - b}{a}$*

*Proof.* To ensure that Constraint 4.8 is always satisfied, $aQ + b$ is supposed to be less than or equal to the shortest waiting time for all customers, i.e., $aQ + b \leq \min\{w_j \mid j \in \mathcal{J}\}$. We can then obtain $Q$'s upper limit. □

**Lemma 5.** *When $X_{ij}$ is given, and $Q$ reaches its upper limit, the profit function is concave on $R$.*

*Proof.* There are two cases to discuss:

**Case I**: if $Q = D$, neither shortage nor salvage will occur, and the profit function becomes

$$\begin{aligned} f(R) &= R\sum_{j\in\mathcal{J}} Y_j - cQ \\ &= R\sum_{j\in\mathcal{J}}\left(\sum_{i\in\mathcal{I}} p_{ij}X_{ij}\mu_j - \sum_{i\in\mathcal{I}}\lambda(R-r)X_{ij}\right) - cQ \\ &= R\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} p_{ij}X_{ij}\mu_j - R^2\lambda\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij} + R\lambda r\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij} - cQ \end{aligned}$$

**Case II**: if $Q < D$, there will be a shortage. Similarly, we have

$$\begin{aligned} f(R) &= R\sum_{j\in\mathcal{J}} Y_j - cQ - s\sum_{j\in\mathcal{J}} Y_j + sQ \\ &= R\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} p_{ij}X_{ij}\mu_j - R^2\lambda\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij} + R(\lambda r + \lambda s)\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij} \\ &\quad - s\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} p_{ij}X_{ij}\mu_j - s\lambda r\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij} + (s-c)Q \end{aligned}$$

For both case, we have $f''(R) = -2\lambda < 0$, implying a concave $f$ on $R$. In addition, for both cases, $f_{max}$ is achieved if and only if $f'(R^*) = 0$.

For **case I**: $R^* = \frac{\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} p_{ij}X_{ij}\mu_j}{2\lambda\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij}} + \frac{r}{2}$

For **case II**: $R^* = \frac{\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} p_{ij}X_{ij}\mu_j}{2\lambda\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij}} + \frac{r+s}{2}$ □

Lemma 5 is a key to reduce the number of evaluations of the MILP prolblem. However, to use it in the solution approach, we need additional theory support.

**Lemma 6.** *If (1) $R$ is given, (2) $Q$ reaches its upper limit, and (3) $Q < D$, then maximizing the profit is equivalent to maximizing $\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} F_{ij}X_{ij}$, in which $F_{ij} = p_{ij}\mu_j - \lambda(R-r)$.*

*Proof.* If both $R$ and $Q$ are given, and there is a shortage. Then our profit function becomes

$$\begin{aligned} f(X_{ij}) &= (R-s)\sum_{j\in\mathcal{J}} Y_j + (s-c)Q \\ &= (R-s)\sum_{j\in\mathcal{J}}\left(\sum_{i\in\mathcal{I}} p_{ij}X_{ij}\mu_j - \sum_{i\in\mathcal{I}}\lambda(R-r)X_{ij}\right) + (s-c)Q \\ &= (R-s)\sum_{j\in\mathcal{J}}\left(\sum_{i\in\mathcal{I}}\left(p_{ij}\mu_j - \lambda(R-r)\right)X_{ij}\right) + (s-c)Q \\ &= (R-s)\sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{I}} F_{ij}X_{ij} + (s-c)Q \end{aligned}$$

Since $R > s$, and $Q = Q_{ub}$, maximizing $f$ is equivalent to maximizing $\sum_{j\in\mathcal{J}}\sum_{i\in\mathcal{I}} F_{ij}X_{ij}$. □

Lemma 6 explains the rationality of choosing new customers: If $F_{ij} > 0$, then customer $j$ should be selected, i.e., $\sum_{i\in\mathcal{I}} X_{ij} = 1$, because a positive $\sum_{i\in\mathcal{I}} F_{ij}X_{ij}$ will contribute to the profit. If, however, $F_{ij} \leq 0$, then customer $j$ will not be selected; thus, $\sum_{i\in\mathcal{I}} F_{ij}X_{ij} = 0$, which does not contribute to the profit. Therefore, the optimizer will select every customer $j$ with a positive $F_{ij}$, as long as there exists an agent who can take new customers.

**Lemma 7.** *During the search, $X_{ij}$ remains unchanged when the number of selected customers does not change.*

*Proof.* Based on Lemma 6, the optimizer will select all customers with a positive $F_{ij}$. In other words, if a customer $j$ is selected, then $F_{ij} = p_{ij}\mu_j - \lambda(R-r) > 0$, i.e., $R < \frac{p_{ij}\mu_j}{\lambda} + r$. Meanwhile, an available agent $i$ with the highest effort $p_{ij}$ will be selected to handle the customer. In conclusion, a customer $j$ will be selected and assigned to an agent $i$ if 1) $R < \frac{p_{ij}\mu_j}{\lambda} + r$, 2) agent $i$ is not overloaded, i.e., $\sum_{j\in\mathcal{J}} X_{ij} \leq g_i$, and 3) $F_{ij}$ is the highest among all agents that can take new customers. However, if there is another agent $i'$ who is also available and has the same effort as agent $i$, the optimizer will break the tie by choosing the agent with a lower agent ID. Compared to Lemma 6, we take one step further. Let us keep reducing $R$, before the next new customer can be selected, $X_{ij}$ will not change. First, all selected customers will remain the same, because unselecting a selected customer will

decrease the profit, and given the current $R$, no new customers can be selected. Second, for a selected customer $j$, the optimizer will not assign $j$ to a different agent, say $i'$, with a lower effort $p_{i'j} < p_{ij}$, which lowers the overall expected profit. Therefore, $X_{ij}$ will remain unchanged as long as the number of selected customers does not change. □

Lemma 7 gives an effective indicator of the status of $X_{ij}$. During the search, if the number of selected customers remains unchanged, $X_{ij}$ remains the same, which provides a condition of applying Lemma 5.

Based on the above analysis, we describe the R-search algorithm as follows:

**Step 1**: Determine $R_{ub}$ based on Lemma 2, and use it as the starting point of the search: $R \leftarrow R_{ub}$. Also, let $R_{lb} \leftarrow s$. In the real world, the selling price should be higher than the shortage cost $s$.

**Step 2**: Solve the reduced MILP problem via CPLEX; if the solution is optimal, add tuple ($R$, profit, $Q$, $X_{ij}$) to the result set.

**Step 3**: If $Q$ has not reached its upper limit, i.e., $Q < Q_{ub}$, then $R \leftarrow R - step_sz$. Go to step 2.

**Step 4**: If $Q$ has reached its upper limit, i.e., $Q = Q_{ub}$:

- If no shortage occurs, i.e., $Q = \sum_{j \in \mathcal{J}} Y_j$. We can calculate a $R'$ based on case I of Lemma 5.
- If shortage occurs, we can calculate a $R'$ based on case II of Lemma 5.
- If $R' > R$, meaning that keep searching downwards will not yield a better solution so the search stops. Go to step 5.
- If the *stop* condition is set, break the search and go to step 5.
- If $R' < R$, then $R \leftarrow R'$. Meanwhile, if all customers are selected, or all agents are working in full capacity, a *stop* condition will be set; go to step 2.

**Step 5**: Return the result with the highest profit in the result set.

In addition, we provide the pseudo code of R-search in Algorithm 4.

**Algorithm 4** R-search

**Input:** Parameter list in Table 4.1, $K^{max}_{sorted}$, step_size

rst_set $\leftarrow \emptyset$, $R_{lb} \leftarrow s$, *stop* $\leftarrow$ False
$R_{ub} \leftarrow \frac{1}{\lambda} K^{max}_{sorted} [J - \lceil J\alpha \rceil + 1] + r$ (see Lemma 2)
$Q_{ub} \leftarrow \frac{\min\{w_j | j \in \mathcal{J}\} - b}{a}$ (see Lemma 4)
$R \leftarrow R_{ub}$
**while** $R > R_{lb}$ **do**
  $rst \leftarrow Cplex.solve$(reduced DL-SNP)
  **if** $rst$ is an optimal solution **then**
    Extract tuple (profit, $Q$, $X_{ij}$) from $rst$
    Add tuple (profit, $R$, $Q$, $X_{ij}$) to rst_set
    Compute $\{Y_j \mid j \in \mathcal{J}\}$
    $D \leftarrow \sum_{j \in \mathcal{J}} Y_j$
    **if** $Q < Q_{ub}$ **then**
      $R \leftarrow R$ - step_size
    **else if** $Q = Q_{ub}$ **then**
      **if** $Q = D$ **then**
        $R' \leftarrow \frac{\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} p_{ij} X_{ij} \mu_j}{2\lambda \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} X_{ij}} + \frac{r}{2}$ (see Lemma 5)
      **else if** $Q < D$ **then**
        $R' \leftarrow \frac{\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} p_{ij} X_{ij} \mu_j}{2\lambda \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} X_{ij}} + \frac{r+s}{2}$ (see Lemma 5)
      **end if**
      **if** $R' > R$ **or** $stop = true$ **then**
        break
      **else**
        $R \leftarrow R'$
        **if** All customers are selected, **or** all agents are in full capacity **then**
          Set the stop flag $stop \leftarrow true$
        **end if**
      **end if**
    **end if**
  **end if**
**end while**
Output the result with highest profit in rst_set.

## 4.4. EVALUATION

We conduct all computational experiments on a workstation with a Intel Core(TM) i7-4770 CPU (3.40GHz), a 16GB RAM, and a 64-bit Windows 10 operation system. The

Table 4.1: Data Specification

| **Parameter** | **Notation** | **Setting** |
|---|---|---|
| # Customers | $I$ | small - [4,6,8,10]; large - [12,14,16,18] |
| # Sales Agents | $J$ | small - [50, 100] by 10; large - [200, 300] by 20 |
| Waiting Time | $w_j$ | U[90, 120] |
| Agent Capacity | $g_i$ | U[20, 40] |
| Mean Demand | $\mu_j$ | U[10, 20] |
| Sales Effort | $p_{ij}$ | U[0.8, 1.2] |
| Unit Prod. Time | $a$ | small - 0.1 day; large - 0.02 day |
| Fixed Shipping Time | $b$ | 3 days |
| Salvage Price | $e$ | \$50 |
| Shortage Price | $s$ | \$90 |
| Price Scale | $\lambda$ | 1.0 |
| Base Price | $r$ | \$100 |
| Production Cost | $c$ | \$70 |
| Service Level Limit | $\alpha$ | 0.8 |

employed solver is IBM CPLEX 12.6. We implement the solution algorithm in C++. In addition, we develop a Python program for data generation.

### 4.4.1. Data Setting

We present the data setting of the experiment in Table 4.1. Based on the instance size, we generate two groups of data, each of which consists of 24 distinct instances. Besides the data size, the two data sets only differ in unit production time, as the demand for the large data set is significantly larger than the small set, so we apply a shorter unit production time for the large set. In other words, the company improves its production efficiency to handle more demands.

### 4.4.2. Results

We display the results for the small data set and the large data set in table 4.2 and table 4.3, respectively. For each instance, we report the optimal profit, the optimal decision variables $R^*$ and $Q^*$, the running time in seconds (denoted by $T$ in the table), the pre-calculated $R_{ub}$, the total actual demand $D = \sum_{j \in \mathcal{J}} Y_j$, and the difference between $Q$ and $D$, denoted by $\Delta_{Q^*,D}$. In addition, to evaluate the demand fulfillment rate, we define three

Table 4.2: Results for the Small Data Set

| ID | I | J | Profit | $R^*$ | $Q^*$ | T | $R_{ub}$ | $D$ | $\Delta_{Q^*,D}$ | $M_1$ | $M_2$ | $M_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4 | 50 | 24621 | 98.5 | 863.9 | 2.7 | 116 | 863.9 | 0.0 | 125% | 124% | 100% |
| 2 | 4 | 60 | 29380 | 103.5 | 870 | 2.0 | 118 | 887.4 | -17.4 | 95% | 94% | 100% |
| 3 | 4 | 70 | 31816 | 106.0 | 870 | 1.7 | 118 | 898.5 | -28.5 | 82% | 81% | 100% |
| 4 | 4 | 80 | 32843 | 105.5 | 870 | 1.5 | 116 | 994.2 | -124.2 | 82% | 73% | 100% |
| 5 | 4 | 90 | 35257 | 105.7 | 870 | 1.3 | 116 | 1133.9 | -263.9 | 82% | 64% | 100% |
| 6 | 4 | 100 | 37411 | 105.8 | 870 | 1.2 | 117 | 1265.3 | -395.3 | 82% | 57% | 100% |
| 7 | 6 | 50 | 26164 | 100.0 | 870 | 2.2 | 116 | 876.4 | -6.4 | 118% | 117% | 100% |
| 8 | 6 | 60 | 29967 | 104.5 | 868.6 | 2.0 | 118 | 868.6 | 0.0 | 93% | 94% | 100% |
| 9 | 6 | 70 | 31333 | 106.0 | 870 | 1.9 | 117 | 870.8 | -0.8 | 83% | 84% | 100% |
| 10 | 6 | 80 | 35139 | 106.6 | 870 | 1.8 | 119 | 1065.5 | -195.5 | 83% | 69% | 100% |
| 11 | 6 | 90 | 36145 | 106.1 | 870 | 1.6 | 118 | 1161.8 | -291.8 | 84% | 64% | 100% |
| 12 | 6 | 100 | 37647 | 105.8 | 880 | 1.2 | 116 | 1266.4 | -386.4 | 84% | 59% | 100% |
| 13 | 8 | 50 | 27292 | 101.5 | 866.4 | 2.7 | 119 | 866.4 | 0.0 | 110% | 110% | 100% |
| 14 | 8 | 60 | 29417 | 104.0 | 865.2 | 2.3 | 118 | 865.2 | 0.0 | 96% | 97% | 100% |
| 15 | 8 | 70 | 31583 | 105.9 | 870 | 1.7 | 116 | 891.2 | -21.2 | 84% | 84% | 100% |
| 16 | 8 | 80 | 33119 | 105.7 | 870 | 1.6 | 115 | 1003.0 | -133.0 | 85% | 75% | 100% |
| 17 | 8 | 90 | 36033 | 106.1 | 870 | 1.4 | 116 | 1158.3 | -288.3 | 84% | 64% | 100% |
| 18 | 8 | 100 | 37229 | 105.7 | 870 | 1.4 | 116 | 1259.5 | -389.5 | 85% | 60% | 100% |
| 19 | 10 | 50 | 26294 | 100.5 | 862.1 | 2.6 | 116 | 862.1 | 0.0 | 116% | 116% | 100% |
| 20 | 10 | 60 | 29960 | 104.5 | 868.4 | 2.5 | 118 | 868.4 | 0.0 | 94% | 95% | 100% |
| 21 | 10 | 70 | 31575 | 105.9 | 870 | 2.0 | 116 | 891.0 | -21.0 | 85% | 84% | 100% |
| 22 | 10 | 80 | 34421 | 106.3 | 870 | 2.1 | 118 | 1043.7 | -173.7 | 84% | 71% | 100% |
| 23 | 10 | 90 | 36052 | 106.1 | 870 | 2.1 | 118 | 1158.9 | -288.9 | 85% | 65% | 100% |
| 24 | 10 | 100 | 38246 | 106.1 | 870 | 1.5 | 116 | 1291.4 | -421.4 | 84% | 58% | 100% |

metrics:

$$M_1 = \frac{\sum_{j\in\mathcal{J}} \frac{Y_j}{\mu_j}}{\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij}} \times 100\% \qquad M_2 = \frac{Q}{\sum_{j\in\mathcal{J}} \mu_j} \times 100\% \qquad M_3 = \frac{\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} X_{ij}}{J} \times 100\%$$

in which $M_1$ computes an average ratio of the actual demand and the expected demand for each individual customer, $M_2$ is a ratio of order quantity fulfilled by the company and total expected demand of the market, and $M_3$ is the actual service level the company achieves. $M_1$ and $M_2$ show the capability of the company fulfilling the market needs from two different aspects.

Table 4.3: Results for the Large Date Set

| ID | I | J | Profit | $R^*$ | $Q^*$ | T | $R_{ub}$ | $D$ | $\Delta_{Q^*,D}$ | $M_1$ | $M_2$ | $M_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 12 | 200 | 109283 | 96 | 4203 | 7.5 | 118 | 4203 | 0 | 141% | 140% | 100% |
| 2 | 12 | 220 | 120086 | 98 | 4289 | 6.6 | 116 | 4289 | 0 | 130% | 130% | 100% |
| 3 | 12 | 240 | 126885 | 99.5 | 4301 | 6.3 | 117 | 4301 | 0 | 122% | 122% | 100% |
| 4 | 12 | 260 | 138413 | 102 | 4325 | 6.6 | 118 | 4325 | 0 | 108% | 109% | 100% |
| 5 | 12 | 280 | 142468 | 103 | 4317 | 5.5 | 116 | 4317 | 0 | 102% | 103% | 100% |
| 6 | 12 | 300 | 148082 | 104 | 4350 | 5.4 | 116 | 4363 | -13 | 97% | 98% | 100% |
| 7 | 14 | 200 | 109330 | 96 | 4205 | 7.2 | 116 | 4205 | 0 | 142% | 141% | 100% |
| 8 | 14 | 220 | 119641 | 97.5 | 4350 | 6.5 | 116 | 4352 | -2 | 134% | 133% | 100% |
| 9 | 14 | 240 | 128517 | 99.5 | 4350 | 6.9 | 116 | 4370 | -20 | 122% | 122% | 100% |
| 10 | 14 | 260 | 137093 | 101.5 | 4350 | 6.8 | 118 | 4356 | -6 | 111% | 111% | 100% |
| 11 | 14 | 280 | 145249 | 103.5 | 4336 | 6.4 | 118 | 4336 | 0 | 100% | 101% | 100% |
| 12 | 14 | 300 | 149143 | 104 | 4350 | 6.1 | 118 | 4439 | -89 | 97% | 96% | 100% |
| 13 | 16 | 200 | 110677 | 96.5 | 4177 | 8.4 | 118 | 4177 | 0 | 139% | 138% | 100% |
| 14 | 16 | 220 | 120576 | 98 | 4306 | 7.8 | 118 | 4306 | 0 | 131% | 131% | 100% |
| 15 | 16 | 240 | 128584 | 99.5 | 4350 | 7.0 | 116 | 4377 | -27 | 123% | 122% | 100% |
| 16 | 16 | 260 | 137155 | 101.5 | 4350 | 7.1 | 118 | 4361 | -11 | 112% | 112% | 100% |
| 17 | 16 | 280 | 146077 | 103.5 | 4350 | 7.1 | 118 | 4376 | -26 | 101% | 101% | 100% |
| 18 | 16 | 300 | 149012 | 104 | 4350 | 6.9 | 118 | 4429 | -79 | 97% | 97% | 100% |
| 19 | 18 | 200 | 112188 | 96.5 | 4234 | 8.5 | 118 | 4234 | 0 | 139% | 138% | 100% |
| 20 | 18 | 220 | 119787 | 98 | 4278 | 8.3 | 118 | 4278 | 0 | 131% | 131% | 100% |
| 21 | 18 | 240 | 130672 | 100 | 4350 | 8.3 | 118 | 4367 | -17 | 120% | 119% | 100% |
| 22 | 18 | 260 | 140199 | 102.5 | 4314 | 7.3 | 118 | 4314 | 0 | 106% | 107% | 100% |
| 23 | 18 | 280 | 144165 | 103 | 4350 | 7.2 | 118 | 4397 | -47 | 103% | 103% | 100% |
| 24 | 18 | 300 | 147949 | 104 | 4350 | 7.1 | 118 | 4354 | -4 | 97% | 98% | 100% |

**Solution:** For both data sets, we obtain the optimal solution for each tested instance. Shortage occurs for 28 out of 48 instances, meaning that in most circumstances the company is unable to fulfill all demands from its dedicated supplier, and has to seek help from a local supplier. We will discuss the potential strategies later in the section of sensitivity analysis. In addition, the achieved service level is 100% for every instance. This means that with our data setting, as long as agents are not in full capacity, the more customers are selected, the more profit the company earns.

**Decision Variables:** For both sets, the optimal selling price $R^*$ is in [\$90, \$110]; depending on particular instances, $R^*$ could be higher or lower than the suggested selling price $r$. Also, $Q^*$ reaches its upper limit for 32 out of 48 instances.

**Performance:** For the small set, the average running time is 1.87 seconds, while it is 7.03 seconds for the large set. The largest instance we test is (I=18,J=300) which includes 5402 decision variables in total, and it only takes 7.1 seconds to finish the search. This result validates the effectiveness and efficiency of the R-search algorithm. We also compare the R-search method with a naive sequential search method in the next subsection.

**Market Satisfaction:** $M_1$ and $M_2$ measure the company's ability to satisfy the market. The difference is that $M_1$ takes into account the additional products from a local supplier when shortage occurs, while $M_2$ only considers $Q$ which is entirely produced by the company's dedicated oversea supplier.

### 4.4.3. Testing the R-search Algorithm

The proposed R-search algorithm is validated through the experiment. From the small data set, we choose a typical instance (I=4, J=100) that demonstrates how the search works. We compare the R-search algorithm to a naive sequential search algorithm, which will search the entire range of $[s, R_{ub}]$ with a step size of 0.5. The results displayed in Figure 4.2 and 4.3 show the searching performance for the R-search and the naive search, respectively. The R-search algorithm finds the optimal solution ($R^* = 105.8$, profit=34711) in 1.23 seconds, while the naive search algorithm takes 13.85 seconds to find a solution ($R^* = 106$, profit=34708). In terms of the number of MILP problems, the naive search evaluates 54 MILP problems, while the R-search only evaluates 15 MILP problems, showing a 72% improvement.

The results are also in alignment with our analysis. R-search does a sequential search when there is no shortage, i.e., $Q - D = 0$. Once shortage occurs, and $Q$ has reached its upper limit, R-search will compute a $R^*$ based on lemma 5, which can significantly reduce the number of MILP problems to be evaluated. For this instance, R-search identifies the optimal price \$105.8 immediately after the algorithm detects a shortage for the first time ($R = \$110.5$). On the other hand, the naive sequential search evaluates all MILP problems

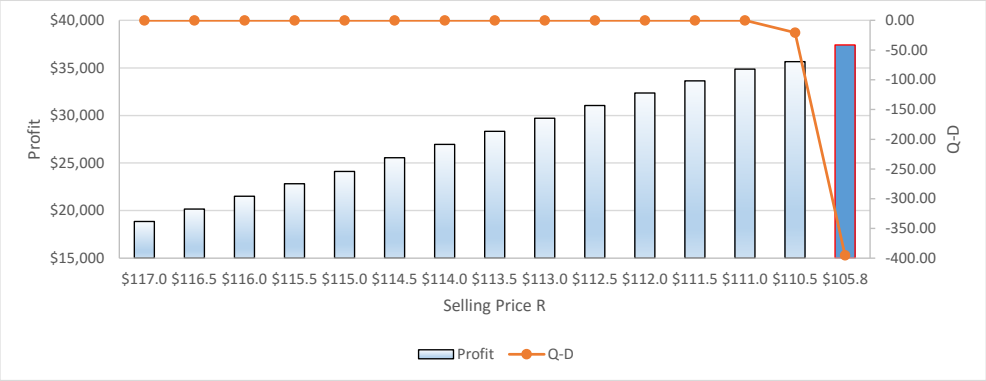



Figure 4.2: R-search for Instance (I=4, J=100) with a running time of 1.2 second

Figure 4.3: Naive Search for Instance (I=4, J=100) with a running time of 13.9 seconds

in the searching range with a fixed step size of 0.5. Figure 4.3 shows a concave curve for the profit function, although strictly speaking, the profit is only concave on $R$ beyond the point of $R = 110.5$, when shortage occurs and that $X_{ij}$ will not change since all customers are selected.

### 4.4.4. Parameter Sensitivity Analysis

For the sensitivity analysis on each parameter, we take one instance from the small data set, and only vary the parameter under study, while keeping all other parameters untouched. We list the tested parameters and their values in Table 4.4.

Table 4.4: Parameter Sensitivity Analysis

| **Parameter** | **Notation** | **Setting** |
|---|---|---|
| Market size | $\mu_j$ | U[8,12], U[10,20], U[15,25], U[30,50] |
| Agent Capacity | $g_i$ | U[5,10], U[10,15], U[15,20], U[20,30] |
| Price Scale | $\lambda$ | [0.1, 1.4] by 0.1 |
| Base Price | $r$ | [\$95, \$150] by \$5 |
| Production Cost | $c$ | [\$50, \$80] by \$5 |
| Unit Production Time | $a$ | [0.1, 0.5] by 0.1 day |
| Service Level Limit | $\alpha$ | [0, 1] by 0.01 |

Table 4.5: Impact of Market Size on Instance ($I = 4$, $J = 50$)

| Size | $\mu_j$ | Profit | $R^*$ | $Q^*$ | T | $R_{ub}$ | $D$ | $\Delta_{Q^*,D}$ | $M_1$ | $M_2$ | $M_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| XL | U[30, 50] | 62982.8 | 123.8 | 870 | 2.4 | 148 | 1350 | -480 | 66% | 44% | 100% |
| L | U[15, 25] | 32576.3 | 107.5 | 869 | 2.5 | 124 | 869 | 0 | 86% | 87% | 100% |
| M | U[10, 20] | 25682.1 | 99.5 | 870 | 2.4 | 116 | 872 | -2 | 119% | 119% | 100% |
| S | U[8, 12] | 19307.3 | 92.5 | 858 | 2.8 | 112 | 858 | 0 | 176% | 174% | 100% |

**Market Size**

We evaluate four markets, referred to as extra large (XL), large (L), medium (M), and small (S), in terms of the market size. For our problem, the size of a market is determined by the expected demand $\mu_j$. Since the four instances only differ in the market size, the upper limit of $Q$ will be the same, which is an indicator of the production capacity of the company's oversea supplier. The results shown in Table 4.5 demonstrate that there is a large shortage for the XL market, while for the rest three market no shortage occurs. This is rational because a larger market will force the company to work with the local supplier to fulfill the shortage without extending the deadline of order delivery. However, we notice that even with shortage, the company in the XL market decides to only fulfill two thirds of expected demands, i.e., $M_1 = 66.33\%$, because to match the amount of overall expected demands, the company has to lower the selling price to attract more orders, which might hurt the profit. Our model demonstrates the ability to decide the optimal selling price, striving to fulfill as much demand as possible without losing profit.

Table 4.6: Impact of Agent Capacity on Instance ($I = 4$, $J = 100$)

| ID | $g_i$ | Profit | $R^*$ | $Q^*$ | T | $R_{ub}$ | $D$ | $\Delta_{Q^*,D}$ | $M_1$ | $M_2$ | $M_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | U[5,10] | 16239 | 98.5 | 569.8 | 5.4 | 127 | 569.8 | 0 | 121.6% | 38.0% | 25% |
| 2 | U[10,15] | 30432 | 104.5 | 880 | 3.4 | 128 | 885 | -5 | 94.8% | 57.9% | 52% |
| 3 | U[15,20] | 33066 | 107 | 890 | 3.2 | 127 | 898 | -8 | 82.9% | 57.4% | 62% |
| 4 | U[20,30] | 36175 | 105.6 | 870 | 3.0 | 128 | 1207.1 | -337 | 83.7% | 60.0% | 97% |

**Agent Capacity**

Agent capacity is another factor that determines a company's ability to fulfill demands of a market. To assess its impact, we specify four different levels of agent capacity, including U[5,10], U[10,15], U[15,20], and U[20, 30] and test the setting on instance (I=4, J=100). In addition, we eliminate the service level constraint by setting $\alpha = 0$, as agent capacity is a hard constraint which might conflict with a pre-set service level, leading to an infeasible solution. We present the results in Table 4.6. Also, the randomly generated agent capacities for the four instances are [6, 5, 8, 6], [15, 14, 10, 13], [15, 16, 15, 16], and [22, 28, 25, 22]. From $M_3$, we find that for each tested instance, all agents are working in full capacity, which can be validated through $\sum_{i\in\mathcal{I}} g_i = M_3 J$. Therefore, a low $M_3$ due to agent capacity implies that the marketing department is understaffed.

**Base Price**

We vary the base price $r$ in [\$95, \$150] by \$5, and generate 12 instances. Based on the result in Figure 4.4, the impact of base price on the system is straightforward. First, a higher base price will bring more profit to the company, as well as increase the selling price $R$. Also, $R$ is less than $r$ for all cases; because of that, the overall demand will increase as the actual price is lower than the suggested price, which attracts more orders. In practice, this marketing strategy is frequently used, as customers are willing to spend money on a discounted product, and ignore the suggested price that is significantly higher.

**Unit Production Time**

Figure 4.5 shows how unit production time $a$ makes an impact on profit, $Q$, and $D$. Clearly, both profit and $Q$ decrease as we raise $a$ from 0.1 to 0.5. After all, a higher $a$

Figure 4.4: Impact of Base Price on Instance ($I = 4$, $J = 50$)

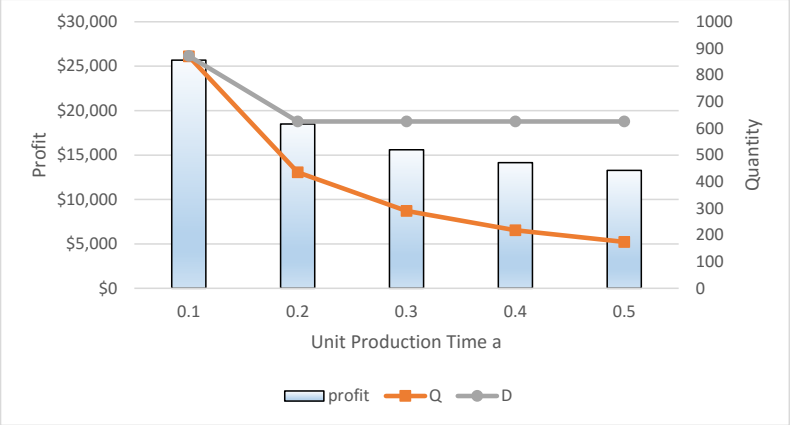



Figure 4.5: Impact of Unit Production Time on Instance ($I = 4$, $J = 50$)

indicates a lower productivity and capacity of the company, meaning that more demands can not be fulfilled, and that more shortage will occur, as indicated by a larger gap between $D$ and $Q$.

**Price Scale**

The price scale $\lambda$ determines how much impact the price $R$ has on a customer's demand. In particular, $\lambda = 1$ means that the price increase/decrease by \$1, the demand will decrease/increase by 1 accordingly. The lower $\lambda$ is, the less impact it has on demand. Based on the data setting, we vary $\lambda$ from 0.1 to 1.4 by 0.1, and present the results in Table 4.7. As shown, when $\lambda$ is low, demand is insensitive to the selling price. Thus, the company can

Table 4.7: Impact of Price Scale $\lambda$ on Instance ($I = 4$, $J = 50$)

| ID | $\lambda$ | Profit | $R^*$ | $Q^*$ | Time | $R_{ub}$ | $D$ | $\Delta_{Q^*,D}$ | $M_1$ | $M_2$ | $M_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.1 | 50180 | 170 | 501.8 | 19.0 | 229 | 501.8 | 0 | 66.4% | 68.6% | 100% |
| 2 | 0.2 | 33166 | 127.5 | 576.8 | 9.7 | 164 | 576.8 | 0 | 77.2% | 78.8% | 100% |
| 3 | 0.3 | 28244.6 | 113.5 | 649.3 | 6.0 | 143 | 649.3 | 0 | 87.6% | 88.7% | 100% |
| 4 | 0.4 | 26345.7 | 106.5 | 721.8 | 4.6 | 132 | 721.8 | 0 | 98.0% | 98.6% | 100% |
| 5 | 0.5 | 25657.6 | 102 | 801.8 | 3.6 | 126 | 801.8 | 0 | 109.4% | 109.5% | 100% |
| 6 | 0.6 | 25570.6 | 99.5 | 866.8 | 3.2 | 121 | 866.8 | 0 | 118.8% | 118.4% | 100% |
| 7 | 0.7 | 25644.4 | 99.5 | 869.3 | 2.5 | 118 | 869.3 | 0 | 119.1% | 118.8% | 100% |
| 8 | 0.8 | 25682.1 | 99.5 | 870 | 2.5 | 116 | 871.8 | -1.8 | 119.5% | 118.9% | 100% |
| 9 | 0.9 | 25705.8 | 99.5 | 870 | 2.2 | 114 | 874.3 | -4.3 | 119.8% | 118.9% | 100% |
| 10 | 1 | 25729.6 | 99.5 | 870 | 2.1 | 113 | 876.8 | -6.8 | 120.2% | 118.9% | 100% |
| 11 | 1.1 | 25753.3 | 99.5 | 870 | 1.7 | 111 | 879.3 | -9.3 | 120.5% | 118.9% | 100% |
| 12 | 1.2 | 25777.1 | 99.5 | 870 | 1.6 | 110 | 881.8 | -11.8 | 120.9% | 118.9% | 100% |
| 13 | 1.3 | 25800.8 | 99.5 | 870 | 1.6 | 110 | 884.3 | -14.3 | 121.3% | 118.9% | 100% |
| 14 | 1.4 | 25824.6 | 99.5 | 870 | 1.4 | 109 | 886.8 | -16.8 | 121.6% | 118.9% | 100% |

Table 4.8: Impact of Waiting Time on Instance ($I = 4$, $J = 50$)

| $w_j$ Range | Profit | $R^*$ | $Q^*$ | T | $R_{ub}$ | $D$ | $\Delta_{Q^*,D}$ | $M_1$ | $M_2$ | $M_3$ |
|---|---|---|---|---|---|---|---|---|---|---|
| (20,40) | 12871 | 105 | 170 | 13.0 | 128 | 631.4 | -461.4 | 84% | 23% | 100% |
| (20,60) | 19090 | 112 | 310 | 26.6 | 140 | 585.9 | -275.9 | 73% | 31% | 68% |
| (20,80) | 29904 | 120 | 510 | 26.7 | 154 | 656.8 | -146.8 | 68% | 39% | 56% |
| (20,100) | 38540 | 121 | 470 | 29.7 | 167 | 940 | -470 | 65% | 29% | 78% |

maximize its profit with a higher selling price. On the other hand, when $\lambda$ becomes higher, demand also becomes more sensitive to the price, forcing the company to reduce the selling price to gain a larger profit. In addition, we did not observe significant shortage in this data set, but a larger shortage is anticipated when $\lambda$ is higher, because the overall demand grows in a faster way.

**Waiting Time** $w_j$

Table 4.8 demonstrates the impact of waiting time. For this experiment, we assume that waiting time is linearly related to expected demand. Letting $w_j = 2\mu_j$, we generate four instances by choosing $\mu_j$ randomly from [10, 20], [10, 30], [10, 40], and [10, 50], respectively. Thus, the ranges of $w_j$ are in [20, 40], [20, 60], [20, 80], and [20, 100] for the four tested instances. Also, to eliminate the influence of the service level constraint, we let $\alpha = 0$ for this

experiment. The result shows that as the range of waiting times becomes larger, the solver eventually decides not to serve certain customers with short waiting times or low expected demand, because a short waiting time places a strong constraint on the total order quantity; thus it is likely that the company would rather not serve this customer, and choose other customers with longer waiting times and higher expected demand to generate more profit. In addition, we find that our solver takes a longer time (an average of 24 seconds) to finish an instance, as CPLEX takes a longer time to solve each MILP problem that needs more computation on customer selection. Column $M_3$ displays the ratio of served customers, in which three out of four instances have an actual service level lower than 80%. If the pre-set service level is 0.8, the company has to give up some customers to meet the required service level by sacrificing some profit.

**Unit Production Cost** $c$

The impact of unit cost $c$ on the problem is obvious; as it only affects the total production cost in the whole problem, a higher unit cost will only reduce the profit by lowering the total production cost, i.e., the $cQ$ term in the objective function. There is no other impact of $c$ on the optimal solution.

## 4.5. CONCLUSION

In this paper, we define two new models, AON-SNP and DL-SNP, for the selective newsvendor problem. The first model considers an all-or-nothing demand pattern, while the second one takes into account a dependent lead time and a demand that can be affected by marketing factors like the selling price and customers' sensitivity to the price. The proposed two models address two different demand patterns which can both occur in the real world.

We develop the R-search algorithm to find an optimal solution for the DL-SNP problem. The R-search algorithm is based off a sequential searching on the selling price $R$, combined with mathematically proven optimizing strategies for performance improvement.

We conduct a series of experiments to evaluate the model. The result shows a premium performance of the R-search algorithm, in comparison with a naive sequential searching. In addition, we examine various responses of the system through parameter sensitivity analysis. Some key findings include:

- The selling price $R$ plays a crucial role in the model. First, it directly affects the demand. Second, it determines whether a customer should be selected to for service or not. It is not a trivial decision to decide a selling price and related marketing strategy such as the best discount rate. Our model can identify the best price that benefits the company most.

- The order quantity $Q$ could also be a tricky decision to make. Our model facilitates this decision. First, the dedicated supplier of the company has an upper limit of order quantity due to dependent lead time. Our model proves that if the total demand is lower than the upper limit of the order quantity, the best strategy for $Q$ is to match that demand; otherwise, if the market increases, more demand comes in, leading to a shortage, and forcing the company to find other local suppliers to fill the additional demand. However, it might not be a good idea to fill as much demand as possible. Our model can find the best shortage quantity that maximizes the profit.

- Customer satisfaction reflects the quality of service of a company. For our model, it is jointly affected by two metrics: $M_1$ and $M_3$. $M_1$ depends on the expected demand. The higher the expected demand is, the larger the potential market is. If the expected demand is accurate, the company has two strategies to respond: the company can either increase its productivity by lowering the unit production time, or outsource partial orders to another supplier. On the other hand, $M_3$ measures the fraction of selected customers over all customers. Clearly, $M_3$ should be as high as possible since denying a customer might hurt the reputation of a company.

## CHAPTER 4 REFERENCES


Aardal, K., Jonsson, O., and Jonsson, H. (1989). Optimal inventory policies with service-level constraints. The Journal of the Operational Research Society, 40:65–73.

Bijvank, M. and Johansen, S. G. (2012). Perodic review lost-sales inventory models with compund possion demand and constant lead times of any length. European Journal of Operational Research, 220:106–114.

Bijvank, M. and Vis, I. (2011). Lost-sales inventory theory: A review. European Journal of Operation Research, 215:1–13.

Carr, S. and Lovejoy, W. (2000). The inverse newsvendor problem: choosing an demand portfolio for capacitated resources. Management Science, 46:912–927.

Chahar, K. and Taaffe, K. (2009). Risk averse demand selection with all-or-nothing orders. Omega, 37:996–1006.

Chen, F. (2005). Salesforce incentives, market information and production/inventory planning. Management Science, 51:60–75.

Chen, F. Y. and Krass, D. (2001). Inventory models with minimal service level constraints. European Journal of Operational Research, 134:120–140.

Chen, M. S. and Chuang, C. C. (2000). An extended newsboy problem with shortage-level constraints. Int. J. Production Economics, 67:269–277.

Gruen, T. W., Corsten, D., and Bharadwaj, S. (2002). Retail out-of-stocks: A worldwid examination of extent acuses and consumer responses. Technical report, Grocery Manufactureres of America.

Ouyang, L., Ho, C., and Su, C. (2009). An optimization approach for joint pricing and ordering problem in an integrated inventory system with order-size dependent trade credit. Computers and Industrial Engineering, 57:920–930.

Taaffe, K., Geunes, J., and Romeijn, H. (2008). Target market selection and marketing effort under uncertainty: The selective newsvendor. European Journal of Operational Research, 189:987–1003.

Tang, C. S. (2010). A review of marketingcoperations interface models: From co-existence to coordination and collaboration. Int. J. Production Economics, 125:22–40.

Tsay, A., Nahmias, S., and Agrawal, N. (1999). Modeling supply chain contracts: A review Quantitative Models for Supply Chain Management. Kluwer Academic Publisher, Boston, MA.

# CHAPTER 5

# CONCLUSION

Recent advances in operations management have witnessed the development and popularity of integrated supply chain management. The notion of the integration of various parties in a modern supply chain including suppliers, distributors, and retailers has been widely studied in academia and also applied by practitioners across numerous industrial fields such as fashion goods, car manufacturers, and the electronic industry. This dissertation studies three integrated supply chain problems, each of which explores a set of unique decision-making policies under customized settings of the studied supply chain network. To conclude this dissertation, we summarize our findings, insights, and future research directions in this section.

In Chapter 2, we propose to solve the PDSD problem using SA and GRASP metaheuristic procedures. We formulate a mixed integer linear programming model which minimizes the total cost of the network with optimally opened DCs from a set of candidate DC locations. The proposed SA and GRASP algorithms employ both long and short memory lists to avoid repetitive instance evaluation, as well as to keep high-quality solution candidates. Compared to the Scatter Search and Tabu Search used in a prior study, both SA and GRASP have shown superior performance in terms of running time and solution quality. We also find that SA and GRASP perform especially well for large instances. Compared with the CPLEX solver, which always yields an optimal solution, our algorithms have saved 80.72% in running time in average. Future research plans include the consideration of a

three-stage PDSD problem that involves plant location decisions, as well as taking a more generic cost function into account.

In chapter 3, we consider a company that procures a product from a distant supplier to serve geographically distributed customers in a single selling season. Different from typical multi-customer newsvendor problems, we assume the firm is not guaranteed to secure the orders from its customers. Given its sales force, the company has to strategically select a fraction of customers to serve. Additionally, due to a limited production capacity of the supplier, the company may not be able to meet all demands. Finally, the lead time from the external supplier is dependent on the order quantity. Hence, the company may lose some of its customers or have to pay penalties for long wait times due to the quantity-dependent lead time. For this problem, we combine the market strategy of assigning customers to each sales agent and inventory procurement policy together to maximize the total expected profit. We build a mixed integer non-linear programming (NPL) model with a linear-quantity-dependent lead time and customer priorities. We propose a solution approach named R-search that looks for the optimal solution among a collection of reduced MILP problem. We assess the model through a series of experiments and validate effectiveness and efficiency of the proposed R-search method. Besides, we investigate the impacts of various parameters such as suggested base price, agent capacity, market size, production cost, and service level. Our model provides guidance on the company's ordering policies and marketing strategies, and thus serves as a promising interface between marketing and operations management.

In Chapter 4, we explore the option of having a supplier (vendor) to centralize a single product ordering for a number of retailers whose demand fluctuates randomly. Each retailer operates in a newsvendor setting, i.e., selling a product to customers in a single period. For a decentralized system, retailers are responsible for order replenishment and stock keeping so as to satisfy customer demands in the most cost-effective way as well as maximize profits. On the other hand, the supplier is responsible for transporting the ordered quantity to the retailers before selling season starts. As opposed to the decentralized system, a centralized

system employs a distribution center (DC) to coordinate and consolidate orders from the retailers. This added DC, whose location should be optimally determined, is able to utilize risk-pooling advantages and save from economies-of-scale in the transportation costs.