UNIVERSITY OF CALGARY

Dynamic Pricing and Scheduling for the Coordination of a One-of-a-kind Production Supply Chain

by

Jian Zhang

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL AND MANUFACTURING ENGINEERING

CALGARY, ALBERTA
SEPTEMBER, 2013

© Jian Zhang 2013
Abstract

In this thesis work, we study the dynamic pricing strategy (DPS) for different cases and its influences on supply chain coordination. First, we study a DPS for a one-of-a-kind production (OKP) firm with two classes of orders (due-date guaranteed and due-date unguaranteed). We model the DPS using Bellman equation and compare it with a static pricing strategy (SPS). Second, we study the pricing problem for a third-party-logistics (3PL) provider that provides warehousing and less-than-truckload (LTL) transportation services. We develop a stochastic-nonlinear-programming (SNLP) model which computes the optimal freight rates for different delivery dates incorporating the 3PL provider’s current holding cost and available transportation capacity. We develop an adjusted multinomial logit (MNL) function to predict customer choices so that our SNLP model can obtain near optimal freight rate settings. Finally, we study dynamic pricing based on a practical OKP firm which is currently employing a SPS. For the three cases, we show the increase of the price-setting firm’s profit, customer and social welfare when DPS is employed through simulation, and consequently show the DPS’s influence on the performance of the supply chain. We also develop a scheduling method for a manufacturer whose suppliers offer different delivery times at different prices. We abstract the problem to a one-machine scheduling problem which is featured by: (a) the release date of each job is compressible and stochastic, (b) each job has to be delivered before its due date (deadline) and (c) the manufacturer can expedite the production with costly overtime. The target is to minimize the total cost including the compressing cost and the overtime production cost. We coin a concept of a job’s late-release-impact factor (LRIF) and we propose a LRIF based heuristic algorithm. Through the numerical test, we shows that the LRIF based algorithm can obtain a better schedule comparing to the ones that are commonly used in practice. By this thesis work, we are trying to integrate an OKP supply chain, which is critical to reduce the cost of OKP companies.
Acknowledgements

I would like to thank my advisor Dr. Tu for all the support, encouragement and guidance during the course of my research work. It has been a truly rewarding experience working with him. I also thank Dr. Nault for sharing so much expertise and valuable experience by which I can benefit for my entire career. I would also extend my sincerest thanks to Dr. Xue, Dr. Liang and Dr. Jabr for agreeing to serve on my committee.

Finally, I would like to thank everyone who is still with me after all these years - my friends, my mother. Thank you all for loving me all the time without a reason.
My Contributions for the Co-authored Papers

The thesis work includes four co-authored papers (two published, one under review and another to be submitted). I am the lead author on all papers and they are as follows:


   I composed the initial draft of the manuscript including all sections. I also composed the response sheet to address the reviewers’ comments based on Dr. Tu and Dr. Nault’s advice.

2. Chapter 4 is mainly based on the paper: “A Dynamic Pricing Strategy for a 3PL Provider with Heterogeneous Customers” by Jian Zhang, Barrie R. Nault and Yiliu Tu, submitted to *Production and Operations Management*.

   I composed the initial draft of the manuscript including all sections.

3. Chapter 5 is mainly based on the paper: “Dynamic Pricing and Its Influences on the Performance of a One-of-a-kind Supply Chain” by Jian Zhang, Barrie R. Nault and Yiliu Tu, to be submitted.

   I composed the initial draft of the manuscript including all sections.


   I composed the initial draft of the manuscript including all sections. I also composed the response sheet to address the reviewers’ comments based on Dr. Tu’s advice.
# Table of Contents

Abstract .......................................................... i
Acknowledgements .................................................. ii
My Contributions for the Co-authored Papers .................. iii
Table of Contents .................................................. iv
List of Tables ....................................................... vi
List of Figures ....................................................... vii
List of Symbols ..................................................... viii

1 Introduction ..................................................... 1

2 Literature Review ................................................ 5
    2.1 Decentralized Coordination .................................... 5
    2.2 Supply Chain Scheduling ....................................... 7
    2.3 Pricing/Queuing Model ......................................... 8
    2.4 The Scheduling with Release-date and Due-date Constraints 11

3 A Dynamic Pricing Strategy for an OKP Firm with Two Classes of Orders 13
    3.1 Background .................................................. 13
    3.2 Notation and Assumptions .................................... 16
    3.3 Dynamic Pricing Strategy (DPS) ............................... 19
    3.4 Static Pricing Strategy (SPS) ................................. 23
    3.5 Welfare Analysis ............................................. 27
        3.5.1 The firm’s profit in DPS and SPS ......................... 27
        3.5.2 The customer’s net welfare obtained from the price-setting firm in DPS and SPS 27
        3.5.3 The customers’ absolute welfare obtained from the price-setting firm in DPS and SPS 31
        3.5.4 The relationship between pricing interval and customer welfare .................... 33
    3.6 Parametric Estimation for Dynamic Pricing .................. 34
        3.6.1 Estimate available capacity ($m$) and future arrivals ($n$) .................. 34
        3.6.2 Learning the distribution of $r$ and $v$ ................. 38
    3.7 Chapter Conclusions ......................................... 41

4 A Dynamic Pricing Strategy for a 3PL Provider with Heterogeneous Customers 43
    4.1 Background .................................................. 43
    4.2 Problem Description .......................................... 49
    4.3 Pricing with Deterministic $N$ and $q$ ....................... 51
        4.3.1 Computing $P_r(p)$ using MLE+MNL based Method ............. 52
        4.3.2 The SNLP Model with Deterministic $N$ and $q$. .......... 56
        4.3.3 Simulation with Deterministic $N$ and $q$ ............... 57
    4.4 Pricing with Random $N$ and $q$ .............................. 62
        4.4.1 The SNLP Model with Random $N$ and $q$ ............... 63
        4.4.2 Simulation with Random $N$ and $q$ .................. 65
    4.5 Compare the Dynamic Pricing Strategy with the Static Pricing Strategy ......... 66
        4.5.1 Compare the 3PL Provider’s Expected Profit, Customer Welfare and Social Welfare 67
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5.2</td>
<td>Compare the Shadow Prices of Reducing Holding Cost and Increasing Transportation Capacity</td>
<td>68</td>
</tr>
<tr>
<td>4.6</td>
<td>Chapter Conclusions</td>
<td>73</td>
</tr>
<tr>
<td>5</td>
<td>A Dynamic Pricing Strategy Incorporating Overtime</td>
<td>75</td>
</tr>
<tr>
<td>5.1</td>
<td>Background</td>
<td>75</td>
</tr>
<tr>
<td>5.2</td>
<td>Notation and assumptions</td>
<td>78</td>
</tr>
<tr>
<td>5.3</td>
<td>Models of the Three Pricing Strategies</td>
<td>82</td>
</tr>
<tr>
<td>5.4</td>
<td>Main Results</td>
<td>87</td>
</tr>
<tr>
<td>5.5</td>
<td>Case Study</td>
<td>90</td>
</tr>
<tr>
<td>5.5.1</td>
<td>The settings for the simulation case study</td>
<td>93</td>
</tr>
<tr>
<td>5.5.2</td>
<td>The simulation for a single-echelon supply chain</td>
<td>95</td>
</tr>
<tr>
<td>5.5.3</td>
<td>The Simulation of a two-echelon Supply Chain</td>
<td>97</td>
</tr>
<tr>
<td>5.5.4</td>
<td>Discussion</td>
<td>102</td>
</tr>
<tr>
<td>5.6</td>
<td>Chapter Conclusions</td>
<td>103</td>
</tr>
<tr>
<td>6</td>
<td>Scheduling with Compressible and Stochastic Release Dates</td>
<td>104</td>
</tr>
<tr>
<td>6.1</td>
<td>Background</td>
<td>104</td>
</tr>
<tr>
<td>6.2</td>
<td>Problem Description and Notations</td>
<td>105</td>
</tr>
<tr>
<td>6.3</td>
<td>Scheduling with Compressible and Deterministic Release Dates</td>
<td>109</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Heuristic for CDR-S</td>
<td>110</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Solving CDR</td>
<td>115</td>
</tr>
<tr>
<td>6.4</td>
<td>Scheduling with Compressible and Stochastic Release Dates</td>
<td>118</td>
</tr>
<tr>
<td>6.4.1</td>
<td>The optimistic policy, the pessimistic policy and the neutral policy</td>
<td>118</td>
</tr>
<tr>
<td>6.4.2</td>
<td>Converting policy with concerns of the impact of late release</td>
<td>120</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Comparing H-CSR(O), H-CSR(P), H-CSR(N) and H-CSR(L)</td>
<td>123</td>
</tr>
<tr>
<td>6.5</td>
<td>Chapter Conclusions</td>
<td>124</td>
</tr>
<tr>
<td>7</td>
<td>Conclusions</td>
<td>126</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>129</td>
</tr>
<tr>
<td>A</td>
<td>Proofs of Theorems</td>
<td>142</td>
</tr>
<tr>
<td>A.1</td>
<td>Proof of Theorem 5.1</td>
<td>142</td>
</tr>
<tr>
<td>A.2</td>
<td>Proof of Theorem 5.2</td>
<td>143</td>
</tr>
<tr>
<td>A.3</td>
<td>Proof of Theorem 5.3</td>
<td>148</td>
</tr>
</tbody>
</table>
### List of Tables

3.1 An example of (1, 2) case .................................................. 31
3.2 \( RMSE \) of MLE based regression and NN based regression ........................................... 40

4.1 \( \vec{a}^S \) and \( \vec{a}^S \) estimated by MLE with standard MNL ........................................... 59
4.2 The regressed \( r_i(p_t) \) for each delivery date option ........................................... 60
4.3 The freight quotes computed by SNLP-D(S) and SNLP-D(A) ........................................... 61
4.4 Comparing the accuracies of standard MNL function and the adjusted MNL function in predicting the probabilities of customer choices† ........................................... 62
4.5 Comparing the 3PL provider’s expected profits under the freight quotes computed from SNLP-D(S) and SNLP-D(A)† ........................................... 63
4.6 The freight quotes for different delivery date options when \( N \) and \( q \) are random ........................................... 65
4.7 Comparing the 3PL provider’s expected profits under the freight quotes computed from SNLP-R(S) and SNLP-R(A)† ........................................... 66
4.8 Comparing the 3PL’s expected profit obtained from the two pricing strategies ........................................... 68
4.9 Comparing customer welfare and social welfare under two pricing strategies ........................................... 68

5.1 The content of each node in Figure 5.3 ........................................... 95
5.2 The result of the simulation for the two-echelon supply chain ........................................... 102

6.1 Notation .................................................. 106
6.2 Comparing different methods for solving CDR-S ........................................... 115
6.3 Comparing H-CDR(RS) and H-CDR(HS) ........................................... 117
6.4 Example: four possible incurred overtimes with different realization of \( r_1 \) and \( r_2 \) ........................................... 121
6.5 The compressing prices quoted by the component suppliers ........................................... 123
6.6 Comparing different scheduling method for CSR problem ........................................... 124
List of Figures and Illustrations

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Problem description</td>
<td>17</td>
</tr>
<tr>
<td>3.2</td>
<td>The adaptive control process</td>
<td>18</td>
</tr>
<tr>
<td>3.3</td>
<td>Solving $\pi_S^*$</td>
<td>23</td>
</tr>
<tr>
<td>3.4</td>
<td>Distributions of $r$ and $v$</td>
<td>25</td>
</tr>
<tr>
<td>3.5</td>
<td>Price quotes obtained from DPS and SPS in different cases</td>
<td>26</td>
</tr>
<tr>
<td>3.6</td>
<td>The gap of the firm’s expected profit under different pricing strategies in different cases</td>
<td>28</td>
</tr>
<tr>
<td>3.7</td>
<td>The gap of customers’ net welfare under different pricing strategies in different cases</td>
<td>30</td>
</tr>
<tr>
<td>3.8</td>
<td>$f(r, v)$ when $r = R - S$, $s \sim U(0, 5)$, $R \sim U(5, 10)$ and $v \sim U(0, 10)$</td>
<td>32</td>
</tr>
<tr>
<td>3.9</td>
<td>The gap of customers’ absolute welfare under in DPS and SPS</td>
<td>33</td>
</tr>
<tr>
<td>3.10</td>
<td>The gap of customers’ net welfare under PPS and SPS for different setting of $l$</td>
<td>34</td>
</tr>
<tr>
<td>3.11</td>
<td>An AND-OR tree for modeling configuration variations of windows</td>
<td>37</td>
</tr>
<tr>
<td>3.12</td>
<td>The outputs of true probability functions and regressed probability functions</td>
<td>41</td>
</tr>
<tr>
<td>4.1</td>
<td>The curves of regressed $r(p)$ in different cases</td>
<td>56</td>
</tr>
<tr>
<td>4.2</td>
<td>The profit ($\pi^{100}$) and the shadow price of reducing holding cost ($\gamma^H$) when the current unit holding cost ($h$) is at different values in DPS and SPS</td>
<td>70</td>
</tr>
<tr>
<td>4.3</td>
<td>The profit ($\pi^{100}$) and the shadow price of increasing transportation capacity ($\gamma^C$) when the current daily transportation capacity ($c$) is at different values in DPS and SPS</td>
<td>72</td>
</tr>
<tr>
<td>5.1</td>
<td>A multi-echelon OKP supply chain</td>
<td>76</td>
</tr>
<tr>
<td>5.2</td>
<td>An example production line for vinyl window</td>
<td>92</td>
</tr>
<tr>
<td>5.3</td>
<td>Flowchart of the simulation program for each working day</td>
<td>94</td>
</tr>
<tr>
<td>5.4</td>
<td>The curves of optimal $x^D$ and $x^P$ under different settings of $p^D$ and $p^P$ in DDP and PDP, respectively</td>
<td>96</td>
</tr>
<tr>
<td>5.5</td>
<td>The curves of the firm’s profits under different settings of $p^S$, $p^D$ and $p^P$ in SP, DDP and PDP, respectively</td>
<td>97</td>
</tr>
<tr>
<td>5.6</td>
<td>The curves of customers’ profit under different settings of $p^S$, $p^D$ and $p^P$ in SP, DDP and PDP, respectively</td>
<td>98</td>
</tr>
<tr>
<td>5.7</td>
<td>The curves of the supply chain’s profit under different settings of $p^S$, $p^D$ and $p^P$ in SP, DDP and PDP, respectively</td>
<td>98</td>
</tr>
<tr>
<td>5.8</td>
<td>A two-echelon MTO supply chain</td>
<td>99</td>
</tr>
<tr>
<td>5.9</td>
<td>The production schedules of M1 and M2 under different pricing strategies</td>
<td>101</td>
</tr>
<tr>
<td>6.1</td>
<td>An example of HS</td>
<td>113</td>
</tr>
<tr>
<td>6.2</td>
<td>An example to the superiority of LPT rule when multiple jobs have the same deadline</td>
<td>117</td>
</tr>
</tbody>
</table>
# List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>3PL</td>
<td>third-party-logistics</td>
</tr>
<tr>
<td>CDR problem</td>
<td>the scheduling problem when all the jobs have compressible and deterministic release dates</td>
</tr>
<tr>
<td>CDR-S</td>
<td>the CDR problem when the job sequence is given</td>
</tr>
<tr>
<td>CSR problem</td>
<td>the scheduling problem when all the jobs have compressible and stochastic release dates</td>
</tr>
<tr>
<td>G-order</td>
<td>due-date-guaranteed order</td>
</tr>
<tr>
<td>LRIF</td>
<td>late-release-impact factor</td>
</tr>
<tr>
<td>MLE</td>
<td>maximum-likelihood estimation</td>
</tr>
<tr>
<td>MNL</td>
<td>multinomial logit</td>
</tr>
<tr>
<td>MTO</td>
<td>make-to-order</td>
</tr>
<tr>
<td>MTS</td>
<td>make-to-stock</td>
</tr>
<tr>
<td>DDP</td>
<td>discount-based dynamic pricing strategy</td>
</tr>
<tr>
<td>DPS</td>
<td>dynamic pricing strategy</td>
</tr>
<tr>
<td>JIT</td>
<td>just-in-time</td>
</tr>
<tr>
<td>LTL</td>
<td>less-than-truckload</td>
</tr>
<tr>
<td>OKP</td>
<td>one-of-a-kind production</td>
</tr>
<tr>
<td>PDP</td>
<td>premium-based dynamic pricing strategy</td>
</tr>
<tr>
<td>PPS</td>
<td>periodic pricing strategy</td>
</tr>
<tr>
<td>SME</td>
<td>small- or medium-sized enterprise</td>
</tr>
<tr>
<td>SNLP</td>
<td>stochastic nonlinear programming</td>
</tr>
<tr>
<td>SPS</td>
<td>static pricing strategy</td>
</tr>
<tr>
<td>U-order</td>
<td>due-date-unguaranteed order</td>
</tr>
</tbody>
</table>
WTP willingness to pay

† Abbreviations are sorted in alphabetical order.
Chapter 1

Introduction

A generic supply chain in various industries may include the following business functions: customer, manufacturing, distribution, retail, Transportation, inventory planning, forecasting and supply planning. The main concern of supply chain management (SCM) is to improve competitiveness of the whole supply chain by means of maximizing the customer satisfaction. A customer can be an end consumer (consumer) or a firm that issues orders to other firms. The customer satisfaction can be determined by many factors, e.g., price, quality, brand reputation, leadtime and customization. The weight of each factor varies from customer to customer, which forms customers’ heterogeneity. This research focuses on the development of methodologies and algorithms for optimizing the performance of supply chain considering the heterogeneity of customers’ preferences in the customizations and delivery dates of their orders.

With the explosive growth of internet and e-Commerce, the cost for customers to acquire information about seller prices, product and service offerings is getting cheap and the competition among the firms is increasing (Bakos 1997). As a result, the firms are forced to add value to their products and services to maintain customer loyalty. Many firms increased the customization level of their products to add to their product differentiation in competition. With the advances in manufacturing and information technologies, “mass customization” has been extensively employed by practitioners to produce individually customized products at a cost which is on par with the cost of mass production (Tseng and Jiao 2001), e.g., Dell, National Bicycle, etc. However, in some industries where customers require even more complex customizations that may vary not only in combinations of parts, but also in detailed dimensions, shapes or other calibrations, mass customization is not enough to maintain the
firms’ competitive power and the firms often employ other market strategies for one-of-a-kind production (OKP). OKP is a strategy that allows local firms to provide highly customized products at an effective production rate. OKP features a “once” successful approach for product development and production according to specific customer requirements, and accordingly no prototype or specimen is made (Tu et al. 2000). The firm usually does not keep inventory of some inputs (e.g., parts) and only makes orders on demand. As a consequence, this requires that the entire supply chain provides fast and reliable delivery. Constrained by global competition in prices and costs, the large firms have limited incentives to provide high-variety customization (Cavusoglu et al. 2007). Thus, OKP firms are usually small- or medium-sized enterprises (SMEs). Because the SMEs can meet the specific customer requirements in a far extend, the customization or flexibility is the unique competitive advantages of these SMEs or OKP companies against the large manufacturers.

Because the order from the SMEs to their suppliers is usually discrete, diversified, and with a much smaller batch size due to the high customization in OKP, from the point of the entire supply chain’s view, the thriving of the OKP firms substantially increases the difficulty to coordinate the supply chain because the varying customization requirements make it hard to precisely promise the delivery date for the orders. When a firm schedules the production of orders from both large firms and SMEs, the orders from SMEs are often dispatched with lower priorities due to SMEs’ limited bargaining power and consequently they have to bear a longer leadtime which reduces the efficiency of the entire supply chain. When there are a large number of OKP firms within a supply chain, this inefficiency will cause a sever loss in the supply chain’s profit (or social welfare).

Because of the features of OKP, it is usually hard to employ the traditional strategies, i.e. information and revenue sharing, to solve the coordination problem for the supply chain which contains a large number of OKP firms. Especially in a multi-echelon OKP supply chain where the customers of an OKP firm are also OKP firms, the supply chain coordination is
further complicated (Giannoccaro and Pontrandolfo 2004, Disney et al. 2008, Kelle et al. 2009). The requirements for a more flexible and easy-to-implement strategies motivate this thesis work that focuses on a supply chain’s decentralized coordination.

This thesis work studies the dynamic pricing strategy (DPS) and its influence on an OKP supply chain’s coordination. Based on the features of different requirements within OKP firms, we study the DPS under three different cases:

**Case 1:** The price-setting firm receives two classes of orders: due-date-guaranteed orders and due-date-unguaranteed orders. In this case, the price quote dynamically changes by orders and the firm cannot temporarily increase the production capacity.

**Case 2:** The price-setting firm provides just-in-time delivery service with any number of delivery time options, each of which is priced differentially. We study the DPS based on a case of a third-party-logistics (3PL) provider that provides comprehensive services including warehousing and less-than-truckload (LTL) transportation. In this case, the price quote does not change during each day. The optimal price quote is computed accounting for the 3PL provider’s capacity and inventory holding cost. The 3PL provider can temporarily increase its transportation capacity at an extra cost.

**Case 3:** The price-setting firm provides just-in-time delivery service and keeps no stock for finished products. The firm has long-term customers and spot-market customers. Two types of DPS are studied accounting for the different difficulties of implementing each DPS.

The three cases studied in the present thesis work are motivated by a practical problem faced by a local customized-widow manufacturer. Because the customized-window manufacturer experiences different production conditions in different production lines and different
production period, it is hard to form a universal pricing model which accounts for the manufacturer’s pricing problem in all conditions. Instead, we form different pricing models for different cases so that a manufacturer can choose a proper model for a particular case. Through the study of the DPS in each case described above, we mainly answer two questions: what is the optimal price setting in a DPS and how a DPS will influences the supply chain coordination. For each of the three cases, we proposed the dynamic pricing model by considering customers’ heterogeneity and the stochasticity nature of the problem. Then we develop algorithms to solve the proposed models. At last, through analytical analysis or simulation experiment, we compare the price-setting firm’s expected profit, the customer and social welfare under different pricing strategies. Because the social welfare actually measures the supply chain’s profit, we can finally obtain how the DPS influences the supply chain coordination by comparing the social welfare under different pricing strategies.

As a complement to the proposed DPS, we also develop a scheduling strategy for an OKP firm accounting for the case where release time of each job is stochastic and compressible. That is, when the OKP firm orders parts from its supplier(s), the delivery date of each part is stochastic, but the distribution of the delivery date depends on the price paid by the firm.
Chapter 2

Literature Review

2.1 Decentralized Coordination

For the past decade, the coordination of a MTO supply chain has attracted much attention from academics and industry (Gunasekaran and Ngai 2005). The coordination of a multi-echelon OKP supply chain, as a special form of a MTO supply chain, can be categorized into two types: the centralized coordination and the decentralized coordination. Centralized coordination is often referred as ‘integration’ where there is a sole decision maker and all the firms within the supply chain follow the decisions from the decision maker. With perfect information centralized supply chains can achieve the first-best schedules and thus are more efficient than the decentralized ones. However, in a supply chain which is composed by large amount of firms, a proper decentralized coordination strategy will be important to coordinate the supply chain because the “person on the spot” has the intimate knowledge of his or her immediate surroundings (Hayek 1945). In this thesis work, we study the pricing strategies employed by OKP firms as a market-based solution for the decentralized coordination. When pricing strategies are employed for supply chain coordination, the upstream and downstream firms in the same supply chain are not bonded through contracts. Thus, a multi-echelon OKP supply chain can be easily decoupled into multiple single-echelon ones, each of which includes only a sole price-setting firm and its direct customers (Wouters 1991). Hence, if we can find a pricing strategy which increases the profit of a decoupled MTO supply chain, we can increase the profit of a multi-echelon MTO supply chain by implementing the pricing strategy in the decoupled ones at all the echelons.

A literature review on the decentralized supply chain coordination can be found in the work by Li and Wang (2007) and Kanda et al. (2008). The existing research on decentralized
supply chain coordination has been focusing on devising mechanisms that are not only able to coordinate the activities but also able to align the objectives of independent supply chain members (Chen et al. 2000). The main strategies for decentralized supply chain coordination studied in the literature can be categorized into contract-based and market-based strategies. In the contract-based strategies, a supplier and its customer within a decoupled supply chain share the coordination benefit under contracts. Because we study a supply chain where large amount of spot-market customers are involved, it is not realistic for the firm to work with each customer under contract, and so the contract-based strategies are not applicable. For market-based strategies, some work can be found studying the quantity discount models which adjust the demand flows between each tier within a supply chain, e.g. Dolan (1987), Parlar and Wang (1994), Wang (2002), etc. These papers analyzed the conditions to achieve system optimization through market-based coordination strategies and how different pricing strategies affect the supply chain and each of its member’s profit. There are also some researchers studying the compound pricing strategies. Parlar and Wang (1994), Weng (1995) and Chen et al. (2001) studied the pricing strategy incorporating the quantity discount and franchise fees. Wang (2005) studied pricing strategy where the discount is mutually determined by buyers’ individual lot sizes and annual volume. Cachon and Lariviere (2005) compare different pricing strategies with the revenue sharing strategies in their influences on supply chain coordination. However, according to the best of our knowledge, the research on decentralized supply chain coordination mainly focuses on the make-to-stock supply chain, where the coordination of inventory system is the core problem. In this work, we focus on the coordination of an OKP supply chain, which is seldom studied from the perspective of decentralized coordination.
2.2 Supply Chain Scheduling

The coordination of a MTO supply chain has been studied extensively under the topic of supply chain scheduling which was first defined by Hall and Potts (2000). They made two propositions for a manufacturer to obtain the cooperation from its supplier, i.e., offering an incentive to compensate the supplier for the increased cost, or offering to share holding costs for completed orders at the supplier. Afterwards, the scheduling for a two-echelon supply chain has been studied for different cases. Agnetis et al. (2006) studied the coordination of a supply chain including one supplier and several manufacturers. In their setting, both the supplier and each manufacturer have an ideal schedule, and an ‘interchange cost’ will be incurred by the supplier or a manufacturer if its realized schedule deviates from the ideal one. They proposed an algorithm to minimize the total interchange cost of a supply chain composed by the supplier and a manufacturer. Sawik (2009) studied the supply chain scheduling with the objective of minimizing the total supply chain inventory holding cost, the production line start-up cost and parts shipping cost. Liao et al. (2009) studied the supply chain scheduling with the objective of minimizing the total setup cost incurred by the supply chain. Yeung et al. (2010) studied the supply chain scheduling with the objective of maximize its profit which is a function of the storage time, storage quantity, order sequence dependent weighted storage costs, and idle time. Qi (2011) studied three outsourcing models where the subcontractors are responsible for different parts of the production. Yeung et al. (2011) studied the scheduling of a supply chain with multiple delivery modes, and they developed algorithms to minimize the storage cost and idle time. Manoj et al. (2012) studied the scheduling of a supply chain with the objective of minimizing total cost which includes buffer cost, tardiness cost and resequencing cost. In the papers cited above, the authors focus on algorithm design to obtain coordinated schedule across the supply chain. They did not consider the online scheduling problem and the mechanism design problem, which are the focus of the present thesis work.
The research of supply chain scheduling focusing on the online scheduling can be found from the work by Averbakh and Xue (2007), Averbakh (2010) and Averbakh and Baysan (2012), which studied the online supply chain scheduling which minimizes the total flow time and delivery cost. The research of supply chain scheduling focusing on the mechanism design can be found from the work by Chang and Lee (2003), Li and Xiao (2004) and Chen (2007), Zou et al. (2008), etc.

In the literature of supply chain scheduling, the coordination of manufacturers’ schedules is realized through centralized or decentralized optimization with information and profit sharing. However, as described in Chapter 1, because the OKP supply chain studied in the thesis present work contains large number of OKP firms that make only small-batch and discrete order, it is hard for them to form a relationship strong enough for information and profit sharing. In the present thesis work, because pricing strategies are employed for supply chain coordination, the upstream and downstream firms in the same supply chain are not bonded through contracts. Thus, a multi-echelon OKP supply chain can be easily decoupled into multiple single-echelon ones, each of which includes only a sole price-setting firm and its direct customers (Wouters 1991).

2.3 Pricing/Queuing Model

Because an OKP firm can be treated as a server of production, the literature usually uses pricing/queuing model to model the dynamic pricing strategy which incorporates the order’s price with the delivery time. We can categorize the literature of pricing/queuing model into two streams.

In the first stream, the customers’ heterogeneity in their “impatience factors”, which can be defined as the cost incurred by the customer for each increased leadtime, is ignored. In this stream of literature, pricing is used as a method for controlling the arrival rate of customers. By dynamically adjusting the price quoted, the firm controls the leadtime for each order and
optimizes the utilization of the server. According to the best of our knowledge, Naor (1969) conducted the earliest research on pricing/queuing model in this stream. Later, Naor’s work is extended by considering different settings of the server or customers. Stidham (1985) studied the setting where the server can control the queue length by either charging a fee or closing the queue. So and Song (1998) incorporated the decisions on capacity settings in their pricing model to guarantee the leadtime to customers. Maglaras and Zeevi (2003) studied a setting where multiple customers can share the resource of a single server. Ata and Olsen (2009), Feng et al. (2011) studied a setting where customers’ delay cost follows special forms.

In the second stream of the pricing/queuing literature, the research focuses on the priority pricing in a priority queuing system. To the best of our knowledge, Kleinrock (1967) was the first to study priority queues in which the priorities are associated with prices paid by customers. He showed that their pricing model can generate socially optimal results. Later, Balachandran (1972) studied the stability of the pricing strategies in priority queues, and he showed that it is also possible for a stable pricing strategy to generate non-optimal results. Well-cited research on priority pricing in priority queues can be found in Dolan (1978), Mendelson and Whang (1990) and Rao and Petersen (1998). In these articles, the authors propose a priority pricing mechanism based on self-revelation theory so that each arriving customer is dispatched with the proper priority, and the target of the proposed priority pricing mechanism is to maximize either social welfare (Dolan 1978, Mendelson and Whang 1990) or the firm’s profit (Rao and Petersen 1998). Van Mieghem (2000) extended the study of pricing/queuing model by dynamically changing each sub-queue’s priority ranking according to the status of the waiting jobs in each sub-queue. In the research on priority pricing in priority queues, the firm does not guarantee the delivery dates, and customers make decisions only based on the expected leadtime. Because there is no delivery date guarantee, the firm does not need to control the number of customer choices for each delivery date. Thus,
the pricing methods proposed by the work on priority pricing are only applicable in the case where there is no need to induce customer choices or to control the arrival rate. Hence the priority pricing mechanism cannot be applied for the 3PL providers whose customers specify their delivery date requirements and their freight has to be delivered on specified delivery dates.

With the increased demand for guaranteed delivery dates in business-to-business (B2B) e-commerce, the literature has started to focus on problems with leadtime guarantees, where the leadtime for processing an order cannot exceed the maximum leadtime specified by the customer. Plambeck (2004) studied the pricing for different leadtime options when the firm faces customers that are heterogeneous both in price sensitivity and leadtime sensitivity. In her work, the distribution of customer choices for all leadtime options are induced by the prices. Due to the problem complexity, she only studied the case where the firm faces two classes of customers and offers two different leadtime options. Çelik and Maglaras (2008) and Akan et al. (2012) both studied the leadtime pricing problem for an arbitrary number of customer classes where they guarantee the promised leadtime by expediting as an extra cost. According to our analysis, the pricing method developed in the cited papers cannot be employed because the delivery-date-guarantee problem is more complex than the leadtime-guarantee problem when inventory cost is considered.

Because we focus on the pricing strategy for the firms with customers that are heterogeneous in their impatience factor as defined by Kleinrock (1967), the present thesis work is closely related to the second stream of pricing/queuing literature reviewed above. However, the literature of pricing/queuing models based on priority queues is featured by the following limitations. First, the pricing/queuing models assume that the firm knows the distribution of customers’ valuation and impatience factors. However, in practice, the customers’ profit function, the basis of its valuation for an order, can be arbitrary and hard to estimate. Second, the pricing/queuing models assume that all the customer prefer the shorter leadtime.
Thus, in this thesis work we developed new dynamic pricing strategies which complement the literature of dynamic pricing for OKP firms.

2.4 The Scheduling with Release-date and Due-date Constraints

When studying the scheduling problem of a supply chain where the manufacturers provide delivery-date guaranteed services, a scheduling scheme is often required that accounts for the varying release-date and due-date requirement for each order (or job). Our research of scheduling policy with compressible and stochastic release dates relates to the literature on the scheduling of jobs with release-date and due-date constraints. There is now a substantial literature that studies the scheduling problem with the jobs constrained by the fixed and deterministic release dates, and a detailed survey of the research on this problem setting can be found in Tian (2003). Here we focus on the literature which studies the compressible and stochastic release dates.

A number of papers can be found studying the scheduling with compressible release dates. Cheng and Shakhlevich (2003) studied the single-machine scheduling of unit-time jobs with controllable release dates, and they developed an algorithm with polynomial complexity to find the integer Pareto optimal points that minimize the makespan and the total compression cost. Later, Janiak (1998) generalized the condition of unit-time jobs. Several papers can be found studying the case where both the jobs’ release dates and processing times are compressible, i.e. Cheng et al. (2006a,b), Kaspi and Shabtay (2006), Shakhlevich and Strusevich (2006), Choi et al. (2007b). For more particular case, Shakhlevich and Strusevich (2006) also considered the controllable processing speed of the server in their work. We can find some papers in the literature using the term “resource dependent release date” to refer to the release date which can be compressed under a certain cost. Choi et al. (2007a) and Li et al. (2011) studied the single-machine scheduling problem with resource dependent release dates to minimize the total resource consumption. They do not include the makespan in the
objective function, but constrain the makespan or the total job completion time to be under a given limit. In all the papers cited above, the authors do not consider the due-date constraint of each job. Seldom research has studied the scheduling problem with the concerns of both compressible release dates and due dates. Janiak (1991) and Ventura et al. (2002) both studied single-machine scheduling problem with common due date and resource-dependent release dates. According to our best knowledge, there is no research so far studying the general scheduling problem with jobs constrained by compressible release dates and varying due dates.

For the literature relating to the scheduling with stochastic release dates, we only find Pinedo (1983) in which the author assumes the release date for each job follows an arbitrary joint distribution, but he only studied the case where the target is to minimize the sum of weighted job completion times. The research in the literature of scholastic scheduling mainly focus on the case where the processing time of each job is random, e.g., Rothkopf (1966), or the case where the scheduling is online and the scheduler expects an arrival stream of new jobs, e.g., Megow et al. (2006).

For the literature on due-date-guaranteed scheduling with production expediting, Bratley et al. (1971) first studied the scheduling problem in which the due date of each job is a constraint. They developed an algorithm which finds the optimal schedule that minimizes the elapsed time if a feasible solution exists. A number of papers can be found that include expediting such as Bradley (2004, 2005), Plambeck and Ward (2007), Celik and Maglaras (2008), but none of the papers has considered the constraints of the jobs’ release dates.
Chapter 3

A Dynamic Pricing Strategy for an OKP Firm with Two Classes of Orders

3.1 Background

Because it is usually the case that an OKP firm receives discrete orders which arrive sequentially and are heterogeneous in leadtime requirements, the firm needs a proper strategy to solve the problem of how to allocate capacity to customers from different priority classes. That is, whether to allocate capacity to the current customer or save it for future arrivals that might be from higher priority classes and hence generate higher profits (Keskinocak and Tayur 2004). In the literature, a well-accepted method to solve this problem is price differentiation between priority classes. However, this simple price differentiation usually cannot promise an exact delivery date. Rather, at best each type of order is promised an expected leadtime which is computed based on the theory of priority queues. Because no exact delivery date is promised, this leadtime pricing strategy can only be applied in a business-to-customer (B2C) environment where the customers are usually not very strict about the due-date delivery. In a business-to-business (B2B) context, the firm has to be able to promise the delivery date to some customers that are not flexible in delivery dates. Without a proper leadtime management strategy, these customers might suffer great loss when the firm fails to deliver on their required delivery dates.

In this chapter, we focus on the case where some customers have strict requirements on leadtime guarantee while the others do not. In this case the firm offers two types of orders: due-date-guaranteed orders (G-orders) and due-date-unguaranteed orders (U-orders), and only the G-orders are promised due-date delivery. We compare two pricing strategies: a
dynamic pricing strategy (DPS) and a static pricing strategy (SPS). G-orders and U-orders are priced differently, but in DPS the firm dynamically changes the price for each type of order to maximize its profit. That is, the price quotes to a customer are also determined by its arrival time. We analyze how the supply chain benefits when each firm prices each order accounting for the order’s arrival time together with the firm’s production capacity and schedule. To match the DPS for larger-scale problems, we also introduce a periodic pricing strategy (PPS), which is a compromise between the features of DPS and SPS.

We design numerical experiment to test how the proposed DPS influence the performance of the supply chain. Our results suggest a promising approach to integrate (or coordinate) a two-tier supply chain with no dominant tier.

In the literature, we can find a number of articles studying dynamic pricing, but most research focus on pricing in make-to-stock (MTS) firms, e.g., Transchel and Minner (2009), Ray et al. (2005), etc. In contrast, we use the Bellman equation (Bellman 1957) to compute the price quote for OKP firms, which are often MTO firms. The earliest work we have found on pricing using Bellman equations is Kinacaid and Darling (1963), and more recent studies used similar methods to model supply chain dynamics, e.g., Stadje (1990), Gallego and Ryzin (1994) and Zhao and Lian (2011), etc. Using numerical analysis, we compare price quotes under the two different pricing strategies. The results show that in large-scale problems the SPS prices can be a good approximation for DPS prices. We explain the application of the proposed DPS through an industrial case from a custom window producing firm. In the case study, we propose a method to evaluate the firm’s available capacity and future order arrivals, which we require to compute prices. We chose a custom window manufacturer for case study because it is a typical OKP firm whose products have hierarchical structures and are processed through multiple production lines. The DPS studied in this work can also be applied in other OKP firms where leadtime is a factor of quality but the leadtime is constrained by the firm’s capacity, e.g., molding companies, high-tech component companies,
or service providers such as transportation companies.

We also develop a set of practical parameter estimation methods for our proposed pricing strategy. In the literature, most authors assume that the distribution of the customers’ “impatience factors”, which can be defined as how much it costs a customer for each time unit that the lead time of its order is increased, is exogenous and known by the firm. Based on this distribution of impatience factors, optimal prices are computed following the discipline of “third degree price discrimination” (Perloff 2009). However, in practice, this impatience factor is usually hard to measure or obtain. In the supply chain management literature the distribution of random variables are usually obtained from a learning process. For example, Chen and Plambeck (2008) develop a learning method to obtain the probability of a customer choosing a substitute, and Tomlin (2009) uses a Bayesian learning process to dynamically update the supplier’s yield distribution. In our model, we develop a maximum-likelihood-estimation (MLE) based learning method to estimate the distribution of the customer’s willingness to pay (WTP) as well as the distribution of the impatience factor. There are also several articles that focus on estimating the distribution of the customer’s WTP. Bishop and Heberlein (1979), Hanemann (1984) and Cameron (1988) proposed methods to estimate the mean of the customer’s WTP when the distribution type is known. Kriström (1990) studied the case where the distribution type is not known and proposed a non-parametric estimator of the distribution of the customer’s WTP, which requires larger sample sizes. Due to sample size limitations in OKP supply chains, we extend the previous literature and study the case where the distribution type is known, but the distribution parameters, such as the mean and variance of a normal distribution, must be estimated. This is realistic when the firm has some rough information on its customers’ WTP distribution. As for the customer’s impatience factor, we have not found any work studying the estimation of its distribution.

We organize the chapter as follows. In Section 3.2 we describe the problem and define the notation and assumptions. In Section 3.3 a dynamic pricing method is presented with
a polynomial algorithm to find the optimal solution. In Section 3.4, a SPS is presented. In Section 3.5, we compare the firm and its customers’ welfare under the two different pricing strategies. In Section 3.6, we propose a method to estimate parameters required to compute the dynamic prices. In Section 3.6.1, we present an industrial case to show how to evaluate the firm’s available capacity and future customer arrivals, while in Section 3.6.2, a learning process is designed to estimate the distribution of a customer’s WTP and impatience factor. Section 3.7 contains final remarks for the chapter.

3.2 Notation and Assumptions

We study an OKP firm which accepts two types of orders, G-orders and U-orders. These orders are for the next production period as shown in Figure 3.1. Our problem is to decide the optimal price quote for each type of order. Every newly received G-order is guaranteed delivery by the end of the next period. To guarantee the promised due date of the G-orders, the firm dispatches a higher priority to the G-orders in production. The quantity of unallocated capacity is used to guarantee the delivery of G-orders. We use the term available capacity to represent the quantity of unallocated capacity. The firm does not accept G-orders when it has no available capacity. In addition, the firm stops accepting G-orders at the beginning of the next period, which we name as deadline. After the deadline, the production schedule in that period is frozen. The firm does not allow new orders to be inserted into a frozen schedule because at the beginning of each production period, the firm needs to reallocate the available resources, e.g., the number of workers at each machine, internal and external logistics, etc. Inserting an order usually incurs extra cost. The superiority of freezing production schedules has been proven by Sridharan et al. (1987).

Notation  We denote the two different price quotes the firm offers for G-order and U-order by \( p^G, p^U : p^G, p^U \in \mathbb{R}^+ \), respectively. The available capacity, which represents the number of capacity units (i.e., man-day, man-hour, etc), is denoted by \( m : m \in \mathbb{Z}^+ \). The number of
future arrivals before the deadline is denoted by \( n : n \in \mathbb{Z}^+ \), which is adaptively estimated, noting that each arrival does not necessarily result in an accepted order. \((m,n)\) represents the case in which the firm has \(m\) available capacity and expects \(n\) future arrivals.

We use \( r : r \in \mathbb{R} \) to represent a customer’s WTP for a G-order. A customer’s WTP is determined by two factors, i.e., its valuation on the firm’s product and the substitutes from the firm’s competitors. Supposing that a customer values the firm’s product at \( V : V \in \mathbb{R} \) and the profit it can obtain by choosing the best substitute is \( S : S \in \mathbb{R}^+ \), then we define a customer’s WTP as \( r = V - S \). Because \( V \) and \( S \) are both random variables, then \( r \) is a random variable. We use \( v : v \in \mathbb{R}^+ \) to represent the customer’s impatience factor. We define the impatience factor as the cost incurred to the customer when there is no due-date guarantee. Without loss of generality, we use \( f(r,v) \) to denote the joint probability density function (JPDF). We also use the notation \( f(r,v;\bar{\theta}) \) to represent the JPDF of \( r \) and \( v \) when the form of the distribution functions depends on the vector \( \bar{\theta} \).

The adaptive control process is described as in Figure 3.2. As shown in Figure 3.2, the dynamic pricing module computes the prices \( p^G \) and \( p^U \), and then the firm quotes the prices to arriving customers. The sample collecting module gathers customers’ choices and their arrival rate, and the production monitoring module monitors the workload of the firm’s each production line in real time. The sample collecting module and the production monitoring module periodically pass the information to the parameter estimating module. Based on the
current firm’s production status, customer arrival rate and customer choices, the parameter learning module adjusts the estimation of \( m, n, \vec{\theta}_r \) and \( \vec{\theta}_v \), and then passes the new estimators back to the dynamic pricing module.

![Diagram of adaptive control process]

**Figure 3.2: The adaptive control process**

Assumptions We make the following assumptions to form our model:

**Assumption 3.1.** The firm makes a take-it-or-leave-it offer.

We assume that the firm makes price quote to each customer. If the customer accepts the price quote, then it places the order; otherwise the customer leaves and does not come back. A similar assumption can be found in previous research, e.g., Gallego and Ryzin (1994), where they assumed that customers do not act strategically by adjusting their buying behavior in response to the firm’s pricing strategy. This is also common in practice. As an OKP firm usually keeps little or no parts inventory and only makes orders on demand, but when the firm orders parts, it requires fast delivery. Therefore, if the customer does not accept the current offer, then a substitution has to be found immediately and hence long term strategic behavior does not happen.

**Assumption 3.2.** The processing time of an order is constant.

Here we assume that the workload of a single order is constant and equal to unity, which we take as a capacity unit. This assumption is consistent with the characteristic of OKP
that the batch size of an order is small, or even just a single unit. For the case where the orders are heterogeneous in processing time, we can approximate the optimal prices through our model, and the method is shown in Section 3.6.1. We abstract from the issues of differing set-up costs between orders.

**Assumption 3.3.** *The variable cost of production is zero.*

As mentioned in earlier, we treat the labor cost as a fixed cost, that is, an added order does not incur additional labor cost. We do not consider material cost as the pricing strategy is our focus recognizing that the problem can be easily generalized by subtracting a constant unit production cost from the unit price. A similar assumption can be found in the literature on production planning and scheduling when pricing is considered, for example, Chen and Hall (2010) and Deng and Yano (2006). There is little loss of generality as with a constant processing time, we can take the variable cost of each order to be fixed, and reinterpret prices as net of costs.

### 3.3 Dynamic Pricing Strategy (DPS)

Our setting can be regarded as an extension of the literature in dynamic pricing, which studies cases in which a firm has a stock of goods to dispose of within a specified time, potential customers arrive sequentially and stochastically, and the probability distribution of prices they are willing to pay is known. The firm sets prices so as to maximize expected cash receipts during the sale, recognizing that unsold items are worthless to the firm (Bitran and Caldentey 2003). Our setting is similar because if the available capacity is not fully allocated after the beginning of a production period, then it is worthless. We use the Bellman equation to compute the optimal price quotes when the firm offers two types of orders.

Suppose that a customer arrives and receives price quotations $p^G$ for G-order and $p^U$ for U-order. If the customer chooses a G-order, then its net gain, denoted by $\xi^G$, is $\xi^G = r - p^G$. Otherwise, if the customer chooses a U-order, then its net gain, denoted by $\xi^U$, is
\( \xi^U = r - p^U - v \). The concept of “net gain” associates with the price-setting firm. If a customer has positive net gain by purchasing from the target firm, it means the net profit to be obtained from the target firm is higher than the one to be obtained from the best offer of other firms (outside options). We define the term “absolute gain” as the difference between the customer’s valuation of the product and the product’s price. Then the net gain equals to the difference of the absolute gains the customer makes by choosing the price-setting firm’s product rather than its best outside option. The net gain measures the difference between the net profits that can be obtained from a supply chain with the target firm and a supply chain without the target firm. The customer chooses the option that creates the higher net gain and only purchases when its net gain is non-negative. Otherwise it leaves without purchasing and its net gain (from the firm) is zero. First, we compute the probabilities of the customer choosing each type of order.

The customer chooses the G-order only if \( \xi^G \geq 0 \) and \( \xi^G \geq \xi^U \). Thus, given \( p^G, p^U \) and the distribution of \( r \) and \( v \), the probability of the customer choosing the G-order, denoted by \( \mathbb{P}_G(p^G, p^U) \), can be obtained as

\[
\mathbb{P}_G(p^G, p^U) = \text{Prob}(\xi^G \geq 0 \text{ and } \xi^G \geq \xi^U) \\
= \text{Prob}(r - p^G \geq 0 \text{ and } r - p^G \geq r - v - p^U) \\
= \int_{p^G}^{+\infty} \int_{p^G - p^U}^{+\infty} f(r, v) dvdr \quad (3.1)
\]

The customer chooses the U-order under two circumstances, i.e., \( \xi^G \geq 0 \text{ and } \xi^G < \xi^U \), or \( \xi^G < 0 \text{ and } \xi^U \geq 0 \). Thus, given \( p^G, p^U \) and the distribution of \( r \) and \( v \), the probability of the customer choosing the U-order, denoted by \( \mathbb{P}_U(p^G, p^U) \), can be obtained as

\[
\mathbb{P}_U(p^G, p^U) = \text{Prob}(\xi^G \geq 0 \text{ and } \xi^G < \xi^U) + \text{Prob}(\xi^G < 0 \text{ and } \xi^U \geq 0) \\
= \text{Prob}(r - p^G \geq 0 \text{ and } r - p^G < r - v - p^U) \\
+ \text{Prob}(r - p^G < 0 \text{ and } r - v - p^U \geq 0) \\
= \int_{p^G}^{+\infty} \int_{0}^{p^G - p^U} f(r, v) dvdr + \int_{p^U}^{p^G} \int_{r - p^U}^{-\infty} f(r, v) dvdr \quad (3.2)
\]
The Bellman equation used to compute the optimal price quotes is based on the probabilities obtained from (3.1) and (3.2).

Suppose that when a customer arrives, the firm faces an \((m, n)\) case. For clarity of expression, we let \(n\) include the current customer. We examine the optimal price quote under two different situations: when capacity is greater or equal to the number of future arrivals, and when capacity is less, i.e., \(m \geq n\) and \(m < n\).

When \(m \geq n\), the firm estimates that the number of future arrivals will not exceed the current available capacity. Then the \((m, n)\) case can be treated as a straightforward uncapacitated problem. Because there is no capacity constraint, a single arrival has no impact on the price quotation to the other arrivals. Thus, all the arrivals are quoted the same price. The firm optimizes the price quotation by maximizing the expected profit contributed by each customer. Thus the optimal price quotation for problem \((m, n)\), denoted by \(\{p_{Gmn}^G, p_{Umn}^U\}\), can be obtained by

\[
\{p_{Gmn}^G, p_{Umn}^U\} = \arg \max_{\{p_G^G, p_U^U\}} \left[ p_G^G \mathbb{P}(p_G^G, p_U^U) + p_U^U \mathbb{P}(p_G^G, p_U^U) \right],
\] (3.3)

which can be solved through the first-order conditions. Because there are \(n\) future arrivals, the maximum expected total profit, denoted by \(\pi_n^m\), is

\[
\pi_n^m = n \left[ p_{Gmn}^G \mathbb{P}(p_{Gmn}^G, p_{Umn}^U) + p_{Umn}^U \mathbb{P}(p_{Gmn}^G, p_{Umn}^U) \right].
\] (3.4)

When \(m < n\), the \((m, n)\) case is a constrained problem. Because the price quoted to a later customer depends on the behavior of earlier customers, we use a Bellman equation to solve the optimal price quote for the current customer. The impact of the current customer is summarized as follows:

- If the current customer chooses the G-order, a unit of available capacity is allocated. The firm earns \(p_G^G\) and then faces an \((m - 1, n - 1)\) problem;

- If the current customer chooses the U-order, it does not affect available capacity preserved for G-orders. Thus, the firm earns \(p_U^U\) and then faces an \((m, n - 1)\) problem.
problem;

- If the current customer does not choose either type of order, then the firm earns no profit and faces an \((m, n - 1)\) problem.

Considering the three possibilities, the optimal price quote for the current customer, \(\{p_{mn}^G, p_{mn}^U\}\), can be obtained by solving the following Bellman equation:

\[
\pi_m^n = \max_{\{p^G, p^U\}} [\mathbb{P}_G(p^G, p^U)[p^G + \pi_{n-1}^{m-1}] + \mathbb{P}_U(p^G, p^U)[p^U + \pi_{n-1}^m]
+ [1 - \mathbb{P}_G(p^G, p^U) - \mathbb{P}_U(p^G, p^U)]\pi_{n-1}^m],
\]

where \(\pi_n^m\) can be solved recursively.

When the firm has no available capacity, the customers can only purchase U-orders. Thus \(\pi_0^n\) for any \(n\) can be obtained as

\[
\pi_0^n = \max_{p^U} np^U \text{Prob}(\xi^U \geq 0) = \max_{p^U} np^U \text{Prob}(r - v - p^U \geq 0)
= \max_{p^U} np^U \int_{p^U}^{+\infty} \int_0^{r-p^U} f(r, v) dv dr.
\]

The example in Figure 3.3 illustrates the process of solving for expected profit when available capacity is 3 and the number of future arrivals is 5, \(\pi_3^3\), through recursion. To solve each \(\pi_j^i\) where \(i \leq m\) and \(j \leq n\), \(\pi_{j-1}^i\) and \(\pi_{j-1}^{i-1}\) are first solved sequentially. Note that the values of \(\pi_2^1\) and \(\pi_2^2\) are known when accessed the second time. Thus, a “tabu list” can be employed to substantially reduce the computational time. We preset an \(m \times n\) array as a tabu list to store every solved value of \(\pi_j^i\). At the beginning of each recursion for computing \(\pi_j^i\), we check the tabu list first and see if it is already solved. If \(\pi_j^i\) is solved already, then the current recursion stops and directly returns the solution stored in the tabu list. In Figure 3.3 each node represents a recursion, and there are 13 recursions in the entire computation process.
Let $T(m, n)$ be the number of recursions required in computing $\pi^m_n | n > m$ when the tabu list is incorporated. Then by analyzing the branched figure as in Figure 3.3, it is not difficult to obtain:

$$T(m, n) = 2m(n - m) + 1 \in O(mn).$$

(3.6)

From (3.6), we can conclude that when the tabu list is employed, $\pi^m_n | n > m$ can be computed within polynomial time, which means that solving $p^G_m$ and $p^U_m$ is not an NP-hard problem.

### 3.4 Static Pricing Strategy (SPS)

In this section we present a SPS in which the firm does not adjust the price quoted to the customers, and then we compare the prices obtained by the SPS and our DPS.

Suppose with a SPS, the firm quotes prices $p^G$ and $p^U$ to every arriving customer. The probability of a customer choosing the G-order is $P_G(p^G, p^U)$, and so the number of customers that choose the G-order follows a binomial distribution with $n$ trials, each of which yields success with probability $P_G(p^G, p^U)$. As the number of G-orders is constrained by $m$, then
the expected number of G-orders, denoted by $N_G(p^G, p^U)$, is obtained as

$$N_G(p^G, p^U) = m[1 - B(m; n, \mathbb{P}_G(p^G, p^U))] + \sum_{i=1}^{m} ib(i; n, \mathbb{P}_G(p^G, p^U)),$$  \hfill (3.7)

where $b(k; n, Pr)$ and $B(k; n, Pr)$ are respectively the pmf (probability mass function) and the cdf of $k$ when $k$ is a random integer following a binomial distribution with $n$ trials and success probability $Pr$.

Similar to the G-orders, the number of customers that choose the U-order follows a binomial distribution with $n$ trials, each of which yields success with probability $\mathbb{P}_U(p^G, p^U)$. Note that besides the customers that initially prefer U-orders, some customers choose U-orders because the firm has no available capacity. Thus, the expected number of U-orders, denoted by $N_U(p^G, p^U)$, is obtained as

$$N_U(p^G, p^U) = n\mathbb{P}_U(p^G, p^U) + \sum_{i=m+1}^{n} [i - m] b(i; n, \mathbb{P}_G(p^G, p^U)) \kappa(p^G, p^U),$$  \hfill (3.8)

where $\kappa(p^G, p^U) = \left[ \int_{p^G}^{+\infty} \int_{p^G-p^U}^{r} f(r, v) dv dr \right] / \mathbb{P}_G(p^G, p^U)$. $\kappa$ computes the probability that a customer’s net gain from choosing a U-order is positive, given that it initially prefers a G-order.

Based on (3.7) and (3.8), we can solve the optimal $p^G$ and $p^U$ by

$$\max_{p^G, p^U} p^G N_G(p^G, p^U) + p^U N_U(p^G, p^U).$$  \hfill (3.9)

We present a numerical analysis to compare the prices obtained from the DPS and SPS. In the numerical analysis, we set $r$ and $v$ to be distributed following three different distributions as shown in Figure 3.4. In the case where $r$ and $v$ are independent, as in Figure 3.4(a), the pdf of each random variable is not affected by the value of the other random variable. In Figure 3.4(b) and Figure 3.4(c), $v$ and $r$ are correlated. We use the distribution in Figure 3.4(b) as an example where $r$ and $v$ are positively correlated because the expected value of $v$ is increasing in the value of $r$, and use the distribution in Figure 3.4(c) as an example where $r$ and $v$ are negatively correlated because the expected value of $v$ is decreasing in the
value of $r$. In the independent case, we set $r$ and $v$ to be both distributed from $U(0, \alpha_1)$, while in the positively correlated and negatively correlated cases, we set the $r$ and $v$ to be jointly distributed from the uniform distribution within the area as shown in Figure 3.4(b) and Figure 3.4(c). We set $\alpha_1 = 10$, $\alpha_2 = 0.8$ and $\alpha_3 = 0.8$ for the numerical test. In the three cases, we set $m/n = 2$. We use numerical methods to solve $p^G$ and $p^U$ from (3.9). In order to avoid the complexity incurred from computing the pdf and cdf of a binomial distribution, we approximate the binomial distribution with normal distribution when $n$ is large ($n \geq 30$).

The prices for G-order and U-order under different problem scales in the three cases are displayed in Figure 3.4. We observe that when the problem scale is small ($m$ and $n$ are small), the difference between two pricing strategies is substantial. The results also show that with increased problem scale, the stochastic problem approaches to the deterministic problem. That is, for both DPS and SPS, the price quotes converges to the optimal solution of deterministic problem.

$$\max_{p^G,p^U} p^G \mathbb{P}_G(p^G, p^U) + p^U \mathbb{P}_U(p^G, p^U), \text{ Subject to: } n \mathbb{P}_G(p^G, p^U) \leq m.$$ 

In the numerical analysis, we can also observe that when the problem scale is larger than a certain point, the difference between the prices obtained in DPS and SPS is very small.

Figure 3.4: Distributions of $r$ and $v$
Under the constraint of static pricing, computing the SPS prices can be much more efficient. Thus, based on the results of numerical analysis, we can conclude that SPS prices can be a good approximation for DPS prices when the problem scale is large.

Although in large scale problems the SPS price is a good approximation for DPS prices, the welfare of the firm and of some customers might be significantly different because in SPS the firm does not change price as time passes. Thus, the optimal price quote at any time may deviate from the original setting. In practice, a more common pricing strategy, a periodic pricing strategy (PPS), can be found where the firm changes the price setting periodically when the problem scale is large and it is not feasible to dynamically compute a price quote for every customer arrival. In our version of PPS, we assume that the firm adjusts price quote each time the new customer arrivals reach a fixed number, which we name as the “pricing interval”. In the next section, we also investigate performance of the
supply chain under different settings of pricing interval in PPS.

3.5 Welfare Analysis

In this section, we analyze the firm’s profit and the customers’ welfare when different pricing strategies are employed.

3.5.1 The firm’s profit in DPS and SPS

The firm’s expected profit can be obtained through (3.4), (3.5) and (3.9). In principle, the DPS should always yield weakly greater profits for the firm because the dynamic price could be set to a constant. The gap between the firm’s profits obtained from our DPS and SPS in the three distribution cases are shown in Figure 3.5.1. In Figure 3.5.1, the curves show the percentage by which the expected profit obtained from our DPS is higher than the one obtained from the SPS. We observe that when the expected value of $v$ increases, the superiority of DPS is more substantial.

3.5.2 The customer’s net welfare obtained from the price-setting firm in DPS and SPS

We define the customers’ net welfare obtained from the price-setting firm as the difference of welfare obtained from a supply chain with the price-setting firm and the one from a supply chain without the price-setting firm. The customers’ net welfare in case $(m, n)$ is denoted by $W_{nm}$. In an extreme case, if all no customers buy from the target firm, the customers’ net welfare (from the price-setting firm) is zero.

First we compute any arriving customer’s expected net gain which can be divided into three parts.

1. If $r \geq p^G$ and $v \geq p^G - p^U$, then the customer chooses the G-order. The expected net gain, denoted by $\xi_1(p^G, p^U)$, can be computed as

$$\xi_1(p^G, p^U) = \int_{p^G}^{+\infty} \int_{p^G - p^U}^{+\infty} [r - p^G] f(r, v) dr.$$
2. If $r \geq p^G$ and $v < p^G - p^U$, then the customer chooses the U-order. The expected net gain, denoted by $\xi_2(p^G, p^U)$, can be computed as

$$\xi_2(p^G, p^U) = \int_{p^G}^{+\infty} \int_0^{p^G - p^U} [r - v - p^U] f(r, v) dv dr.$$  

3. If $p^U \leq r < p^G$ and $r - v \geq p^U$, then the customer chooses the U-order. The expected net gain, denoted by $\xi_3(p^G, p^U)$, can be computed as

$$\xi_3(p^G, p^U) = \int_{p^U}^{p^G} \int_0^{r - p^U} [r - v - p^U] f(r, v) dv dr.$$  

When the customer’s $r$ and $v$ are not within the ranges specified above, it does not purchase and its net gain is zero. Based on the expected net gain of each arriving customer,
In (3.10), the first line computes the welfare of the current customer when price quotes are $p_{mn}^G$ and $p_{mn}^U$, respectively, while the second line computes the total welfare of later arriving customers.

When $m \geq n$, all customers are quoted the constant price regardless of the pricing strategy. Thus, the customers’ net welfare can be obtained as

$$W_n^m = \xi_1(p_{mn}^G, p_{mn}^U) + \xi_2(p_{mn}^G, p_{mn}^U) + \xi_3(p_{mn}^G, p_{mn}^U) + \mathbb{P}_G(p_{mn}^G, p_{mn}^U) W_{n-1}^m + [1 - \mathbb{P}_G(p_{mn}^G, p_{mn}^U)]W_{n-1}^m. \tag{3.10}$$

In (3.10), the first line computes the welfare of the current customer when price quotes are $p_{mn}^G$ and $p_{mn}^U$, respectively, while the second line computes the total welfare of later arriving customers.

When $m \geq n$, all customers are quoted the constant price regardless of the pricing strategy. Thus, the customers’ net welfare can be obtained as

$$W_n^m = n[\xi_1(p^G, p^U) + \xi_2(p^G, p^U) + \xi_3(p^G, p^U)], \tag{3.11}$$

where $p^G$ and $p^U$ are computed by (3.3).

When there is no available capacity, $m = 0$, the customers can only choose U-orders, and then the customers’ net welfare is

$$W_n^0 = n \int_{p^U}^{+\infty} \int_0^{r-p^U} [r - v - p^U]f(r, v)dvdr. \tag{3.12}$$

In order to obtain the customers’ net welfare in DPS and SPS, we substitute $p_{mn}^G$ and $p_{mn}^U$ in (3.10), (3.11) or (3.12) with the prices computed under DPS and SPS for case $(m, n)$. Note that in SPS, $p_{mn}^G$ and $p_{mn}^U$ are constant during the recursion.

Figure 3.5.2 shows the percentage by which the customers’ net welfare obtained by DPS is higher than the one obtained by SPS. We observe that in both the independent and correlated cases, the customers’ net welfare obtained by DPS is higher. We can also observe that when the mean of $v$ increases, the customer benefit more from DPS. This is because the SPS increases the chance that the price is underestimated or overestimated when the future supply-demand ratio ($m/n$) changes. If the future $m/n$ decreases, as mentioned in Section 3.4, $p^G$ should increase while $p^U$ should decrease, and hence in SPS G-orders will be underpriced and U-orders will be overpriced. When this happens, customers that arrive
early and purchase G-orders, are better off. However, despite the customers that are better off, the net gain of the customers that purchase U-orders are reduced, and also because the G-orders are underpriced, it increases the risk that the firm does not have enough available capacity for the late arriving customers that would purchase G-orders. On the other hand, if the future \( m/n \) decreases, G-orders will be overpriced and U-orders will be underpriced when SPS is employed. In this case, customers that purchase U-orders will be better off, but the net gain of the customers that purchase G-orders will be reduced.

Consider an extreme case in which there are 1 unit available capacity and 2 arriving customers whose \( r \) and \( v \) are both distributed from \( U(0, 10) \) as shown in Table 3.1. We compute the price quotes for each customer when DPS and SPS are employed. We can observe that although there is a reduction for the welfare of the customer who arrives earlier, the increase of the welfare of the customer who arrives later leads to the increase of the total welfare.
customer welfare. As the firm’s expected profit also increases in DPS, we conclude that the welfare for the entire supply chain is increased.

Because in DPS, both the firm’s and the customer’s net welfare are increased, the net welfare of the global supply chain is increased, and that is the value of DPS over SPS. In the following section, we propose a learning method to estimate the required parameters. These parameters are usually hard to be measured or observed in practice.

### 3.5.3 The customers’ absolute welfare obtained from the price-setting firm in DPS and SPS

We define the customers’ absolute welfare as the welfare obtained by the customers supposing all the customers choose their outside options after rejecting the price-setting firm. Thus, when computing the customers’ absolute welfare, a customer who rejects the price-setting firm may obtain positive profit.

We use a numerical test to show the customers’ absolute welfare in both DPS and SPS.

---

**Table 3.1: An example of (1, 2) case**

<table>
<thead>
<tr>
<th></th>
<th>DPS: {5.93, 4.36}</th>
<th>SPS: {5.49, 4.24}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price quote to the 1st customer:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The welfare of the 1st customer:</td>
<td>0.886</td>
<td>1.054</td>
</tr>
<tr>
<td>The probability of the 1st customer choosing G-order:</td>
<td>34.36%</td>
<td>39.39%</td>
</tr>
<tr>
<td>Price quote to the 2nd customer if the 1st customer chooses G-order:</td>
<td>(∞, 3.33)</td>
<td>(∞, 4.24)</td>
</tr>
<tr>
<td>The welfare of the 2nd customer if the 1st customer chooses G-order:</td>
<td>0.494</td>
<td>0.319</td>
</tr>
<tr>
<td>Price quote to the 2nd customer if the 1st customer does NOT choose G-order:</td>
<td>(5.00, 5.00)</td>
<td>(5.49, 4.24)</td>
</tr>
<tr>
<td>The welfare of the 2nd customer if the 1st customer does NOT choose G-order:</td>
<td>1.25</td>
<td>1.054</td>
</tr>
<tr>
<td>The expected welfare of the 2nd customer:</td>
<td>0.990</td>
<td>0.764</td>
</tr>
<tr>
<td>Total welfare of the two customers:</td>
<td><strong>1.876</strong></td>
<td><strong>1.818</strong></td>
</tr>
<tr>
<td>The expected profit of the firm:</td>
<td><strong>4.26</strong></td>
<td><strong>4.19</strong></td>
</tr>
</tbody>
</table>
In the numerical test, we suppose that by choosing their best outside option, a customer’s absolute gain, $S$, is distributed from $U(0, 5)$. We set a customer’s valuation of the price-setting firm’s product, $R$, to be distributed from $U(5, 10)$. Supposing that a customer’s impatience factor, $v$, is distributed from $U(0, 10)$, then based on the definition of a customer’s WTP, $r = R - S$, in Section 3.2 we have $f(r, v)$ as in Figure 3.8. If a customer chooses G-order, then its absolute gain is $R - p^G$; if it chooses U-order, then his absolute gain is $R - p^U - v$; if it rejects the price-setting firm, then its absolute gain is $S$. It is straightforward that without the price-setting firm, the absolute welfare of $n$ customers would be $nE(S)$, where $E(S)$ is the expected value of $S$.

We keep $n/m = 2$ and compare the customers’ absolute welfare for problems at different scales. Due to the complexity of computing the customers’ absolute welfare, we evaluate the customers’ absolute welfare based on simulation. That is, we simulate the case where a number of customers arrives sequentially. In DPS, each customer are quoted prices based on current $m$ and $n$, while in SPS, each customer are quoted the same prices computed for the first customer. By summation, we can obtain the customers’ total absolute gain. We repeat the simulation experiment for 100 times, and then the mean of the total absolute gain in
The gap between customers’ absolute welfare using DPS and SPS under different problem scales is shown in Figure 3.9. The dashed line in Figure 3.9 is the power trendline generated by MS Excel. It can be observed that in DPS the customers’ absolute welfare is higher the one in SPS, which is consistent with the result of comparison of customers’ net welfare between in DPS and in SPS.

3.5.4 The relationship between pricing interval and customer welfare

In large scale problems a compromise pricing strategy, PPS, is often employed. We examine the relationship between the price setting interval and the customer’s welfare is studied through a simulation experiment.

In the simulation experiment, we set a customer’s WTP and impatience factor to be both identically and independently distributed from $U(0, 10)$. We set $m = 250$ and $n = 500$ to simulate a large scale problem. We show in Figure 3.10 the percentage by which the customers’ net welfare obtained by PPS is higher than the one obtained by SPS under
Figure 3.10: The gap of customers' net welfare under PPS and SPS for different setting of \( l \) different settings of the pricing interval, denoted by \( l \). From Figure 3.10, we can see that the PPS increases the customer’s welfare, and the superiority of \( l \) decreases when \( l \) is increased. We observe in Figure 3.10 that when \( l \) is less than 100, the change of the superiority of PPS is not very significant. This observation indicates that in practice, the firm can increase the pricing interval within certain range without significantly affecting the customer welfare.

3.6 Parametric Estimation for Dynamic Pricing

As stated in the description of the control system in Section 3.2, the parameter learning module needs to adaptively change the estimates of \( m \), \( n \) and other parameters related to the distribution of \( r \) and \( v \).

3.6.1 Estimate available capacity \((m)\) and future arrivals \((n)\)

In this work, we consider a case where the overall production system within the firm is composed by many subsystems, and the overall demand has various requirements for capacity of each subsystem. We use an industrial case from a custom window manufacturer, which
has been referred to in a previous study (Hong et al. 2010), as an example to demonstrate
the process of identifying the estimators of $m$ and $n$.

As introduced by Hong et al., the design of a custom window can be described by an
AND-OR tree as shown in Figure 3.11. For the purpose of clarity, We use $N\#$ to name a
node in the tree. When a node of the tree, say node $N$, is chosen, the customer then need
to choose its child node(s). Let the set of $N$’s direct child nodes be $DC(N)$, and then the
conditional probability of choosing a child node $e : e \in DC(N)$ is denoted by $PROB_e$. If
$DC(N)$ is dominated by an AND relationship, then $PROB_e = 1 \ \forall e \in DC(N)$. If $DC(N)$ is
dominated by an OR relationship, then $\sum_{e \in DC(N)} PROB_e = 1$

We suppose that any node in the AND-OR tree, $N$, must be processed by a production
line, which is denoted by $L_N$. Note that the production line here can be a virtual production
line. For example, $N1321$ and $N1322$ are just nodes representing feature options, no real
production line is assigned to these nodes. The window production is accomplished through
five real production lines, of which three produce frames (i.e., wood, metal and vinyl), one
cuts glass, and one is for final assembling. In Figure 3.11, each node which requires a real
production line is marked with ‘*’.

Here we define two types of available capacity relating to a node ($N$), i.e., the available
capacity of the production line, denoted by $CL_N$, and the overall available capacity of node,
denoted by $C_N$.

Suppose that the maximum workload that can be processed on $L_N$ in a production period
is $MAXW_N$, and the current total workload of the jobs for G-orders to be processed on $L_N$
is $CURW_N$. Then if $N$ is a node with ‘*’, $CL_N$ can be obtained as

$$CL_N = \frac{MAXW_N - CURW_N}{MAXW_N}$$

If $N$ is a node with no ‘*’, since no real production line is required, then $CL_N = 1$.

$C_N$ is mutually determined by $CL_N$ and the available capacity of $N$’s direct child nodes.
$C_N$ can be obtained as

$$C_N = \min\{CL_N, \sum_{e \in DC(N)} PROB_e C_e\} \text{ if } DC(N) \text{ is dominated by OR relationship}, \quad (3.13)$$

or $C_N = \min\{CL_N, \min_{e \in DC(N)} C_e\} \text{ if } DC(N) \text{ is dominated by AND relationship}.

In (3.13), we use $PROB_e$ as the weight of $C_e$ when computing $C_N$ because when the customers are more prone to choose child node $e$, then $e$ is more critical when computing the overall capacity.

Because each customer may choose different customization for their products, the maximum number of windows the firm can produce varies from period to period. Thus, we denote the average of the maximum number of windows produced in each period by $AMAX$. Then the available capacity, $m$, can be obtained as

$$m = C_{N1} \times AMAX$$

The future arrivals $n$ is the future arrivals before the deadline. In most theoretical work, the assumption is made that the customers arrive according to a Poisson process, and so $n$ can be estimated as $\lambda t$, where $\lambda$ is the arrival rate and $t$ is the time left to the deadline. However, in many cases, the arrival rate is not just a function of time. For example, when the firm’s target market is limited, then the number of future arrivals can also be related to the past arrivals. A variety of methods to predict the expected number of arrivals can be found in Armstrong and Green (2005).

Even though we assumed the processing time of an order is constant (Assumption 3.2), in practice a customer’s order may vary in its capacity requirement (or processing time). In such cases, we can use our method to compute the price for an order with average capacity requirement. Supposing that after accepting the order, the available capacity becomes $m'$, then the prices for G-order and U-order can be approximately obtained at $p_{mn}^G(m - m')$ and $p_{mn}^U(m - m')$, respectively.
Figure 3.11: An AND-OR tree for modeling configuration variations of windows
3.6.2 Learning the distribution of \( r \) and \( v \)

Because it is hard to observe the customer’s WTP and impatience factor in practice, in this section we focus on developing a learning process to estimate the distributions of \( r \) and \( v \).

The form of a distribution might be determined by many features, such as the distribution type (i.e., exponential, normal, uniform, etc.). In practice, the firm might have more information about some particular features but less information about others. We study the case in which the firm knows the distribution type of the customer’s WTP and impatience factor, but is uncertain about values of the distribution parameters. Without loss of generality, we suppose that the firm knows that \( r \) and \( v \) are independent, and the distribution types of \( r \) and \( v \) are normal, but the means (\( \mu_r \) and \( \mu_v \)) and the variances (\( \sigma_r^2 \) and \( \sigma_v^2 \)) are unknown. Thus, the goal of the learning process is to estimate \( \mu_r \), \( \mu_v \), \( \sigma_r \) and \( \sigma_v \) so that the distribution functions required for computing the optimal price can be formed.

We let \( f_r(r; \mu_r, \sigma_r) \) and \( F_r(r; \mu_r, \sigma_r) \) respectively be the pdf and cdf of \( r \), and let \( f^v(v; \mu_v, \sigma_v) \) and \( F^v(v; \mu_v, \sigma_v) \) respectively be the pdf and cdf of \( v \). Because extra parameters (\( \mu_r \), \( \sigma_r \), \( \mu_v \) and \( \sigma_v \)) are needed as inputs in the cdf and pdf, we modify the argument in the notation for the probabilities that a customer purchases a G-order or a U-order in (3.1) and (3.2). Supposing that the firm has recorded \( K \) customers’ purchases and customer \( k : k \in \{1, \ldots, K\} \) is quoted prices \( p^G_k \) and \( p^U_k \) for each type of orders, then the probabilities that a customer purchases a G-order or a U-order can be represented as \( \overline{\theta} \), \( \overline{\theta} \), \( \overline{\theta} \), \( \overline{\theta} \), respectively where \( \overline{\theta} = (\mu_r, \sigma_r, \mu_v, \sigma_v) \).

We develop a maximum-likelihood-estimation (MLE) based parametric estimation method to find the estimates of \( \mu_r \), \( \mu_v \), \( \sigma_r \) and \( \sigma_v \). The likelihood, denoted by \( L \), is constructed as

\[
L = \prod_{k=1}^{K} \left[ \left[ 1 - \overline{\theta} \left( p^G_k, p^U_k \right) - \overline{\theta} \left( p^G_k, p^U_k \right) \right] \frac{\gamma_k^G}{\gamma_k} \left[ \overline{\theta} \left( p^G, p^U \right) \right]^\gamma_k \left[ \overline{\theta} \left( p^G, p^U \right) \right]^\gamma_k \right], \tag{3.14}
\]
where

\[ y_k^0 = \begin{cases} 
1 & \text{if customer } k \text{ leaves without purchase;} \\
0 & \text{otherwise.} 
\end{cases} \]

\[ y_k^G = \begin{cases} 
1 & \text{if customer } k \text{ chooses G-order;} \\
0 & \text{otherwise.} 
\end{cases} \]

\[ y_k^U = \begin{cases} 
1 & \text{if customer } k \text{ chooses U-order;} \\
0 & \text{otherwise.} 
\end{cases} \]

To simplify the form of the likelihood, we take natural logs both sides of (4.2). After simple algebra, we can obtain

\[ \ln L = K \sum_{k=1}^{K} \left[ y_k^0 \ln \left( 1 - \mathbb{P}_G(p^G_k, p^U_k) - \mathbb{P}_U(p^G_k, p^U_k) \right) + y_k^G \ln \mathbb{P}_G(p^G, p^U) + y_k^U \ln \mathbb{P}_U(p^G, p^U) \right]. \]

(3.15)

Because of the complexity of solving the first-order condition of (4.3), we use trust-region method (Conn et al. 2000) to search the optimal setting of parameters.

We present a neural network (NN) based regression method as a benchmark for the MLE based parameter estimation method. In NN, we train the network to regress the probability of a customer choosing each type of orders with respect to the price quote for each type of orders. That is, \( p^G_k \) and \( p^U_k \) are the inputs, and \( \mathbb{P}_G(p^G, p^U) \) and \( \mathbb{P}_U(p^G, p^U) \) are the outputs, where \( \omega \) is the vector of weights which will be determined through Levenberg-Marquardt backpropagation training. In the training process, \( y^G \) and \( y^U \) are the target values for \( \mathbb{P}_G \) and \( \mathbb{P}_U \), respectively. We set 1 hidden layer which includes 20 neurons for the NN.

In the simulation, \( r \) and \( v \) are each normally distributed such that \( r \sim N(10, 2^2) \) and \( v \sim N(3, 2^2) \). The prices quoted to each customer are randomly generated such that \( p^G \sim U(0, 15) \) and \( p^U \sim U(0, p^G) \). Given a combination of \( r, v, p^G \) and \( p^U \), we obtain each customer’s choice indicators, \( y^0, y^G \) and \( y^U \).

We test the two regression methods under different volumes of historical sale records. Then we compare the outputs of MLE based method and NN based method with the true
Table 3.2: RMSE of MLE based regression and NN based regression

<table>
<thead>
<tr>
<th>Regressed function</th>
<th>Sample size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50</td>
</tr>
<tr>
<td>MLE</td>
<td></td>
</tr>
<tr>
<td>$\mathbb{P}^\theta_G$</td>
<td>15.33%</td>
</tr>
<tr>
<td>$\mathbb{P}^\theta_U$</td>
<td>16.08%</td>
</tr>
<tr>
<td>NN</td>
<td></td>
</tr>
<tr>
<td>$\mathbb{P}^\omega_G$</td>
<td>27.50%</td>
</tr>
<tr>
<td>$\mathbb{P}^\omega_U$</td>
<td>27.05%</td>
</tr>
</tbody>
</table>

The result computed by (3.1) and (3.2) in Section 3.3. In the comparison, we randomly choose $p^G$ and $p^U$ from $U(0, 15)$ and $U(0, p^G)$, respectively, and then show the coefficient of variation of the root mean square error, $CVRMSE$, computed as

$$CVRMSE = \sqrt{\frac{E[(y - \hat{y})^2]}{\bar{y}}}$$

where $y$ is the output of the regressed probability functions, $\hat{y}$ is the output of the true probability functions ((3.1) and (3.2)), and $\bar{y}$ is the mean of $\hat{y}$. The comparison of MLE based regression and NN based regression is shown in Table 3.2. In Table 3.2, each value of $CVRMSE$ is the average of 20 rounds of simulation. We observe that the MLE based regression method can achieve an output much closer to the true value computed by (3.1) and (3.2), which supports the superiority of the MLE based regression. However, this MLE based regression method is only applicable when the distribution types of $r$ and $v$ are known. In the case where $r$ and $v$ are distributed with unknown distribution, the NN based regression is the only option because $r$ and $v$ are not required to form $\mathbb{P}_G^\omega(p^G, p^U)$ and $\mathbb{P}_U^\omega(p^G, p^U)$.

Figure 3.12 graphically shows the outputs of the true probability functions and the regressed probability functions using different regression methods when the sample size is 500. We expect that the outputs of NN based regression approximates better when the number of neurons and sample sizes are increased, whereas it is usually not realistic to obtain a large enough sample. However, because the NN based regression is a substitute of the MLE based regression when the firm cannot estimate the distribution type of $r$ and $v$, then it is a good practice for the firm to simultaneously run the two regressions so that the firm can
capture the failure of estimating the distribution type of $r$ and $v$.

![Figure 3.12: The outputs of true probability functions and regressed probability functions](image)

3.7 Chapter Conclusions

In this research we modeled a DPS when the firm’s capacity is limited and a due-date guarantee may be required or favored by some customers. We formulated a dynamic Bellman model to compute the optimal price quotes. We analyzed the computational complexity of the proposed dynamic method, proving that the complexity of the dynamic method to compute the optimal price quotes is polynomial.
Usually it is believed that if the firm dynamically changes prices to maximize its profit, the market demand is exploited and so the customers’ welfare is lowered. However, by comparing DPS with SPS, we show that when the firm dynamically change the prices for each type of orders, both the firm and the customers are better off. This finding strongly supports the superiority of DPS. In the literature, we have not found any research studying the how the DPS increases benefits to the entire supply chain.

For practical applications, we also proposed methods to estimate the parameters, which are complementary to the proposed pricing strategy. We introduced the methods to evaluate the firm’s current available capacity, future arrivals, and the distribution of the customer’s WTP and impatience factor. In the estimation of the distribution of WTP and impatience factor, the proposed estimation method can be easily extended so that the distribution with more unknown parameters can be estimated. We also presented a case study where DPS and SPS could be employed.
Chapter 4

A Dynamic Pricing Strategy for a 3PL Provider with Heterogeneous Customers

4.1 Background

In this chapter, we study the dynamic pricing strategy for a third-party-logistics (3PL) provider. In recent years, the 3PL industry has grown swiftly with E-commerce. According to US census bureau, E-commerce based manufacturing shipments have grown from US$996 billion to US$2,283 billion during the seven years from 2004 to 2010. The growth of E-commerce increases the intensity of competition in many industries, and increasingly companies are encouraged to embrace the one-stop logistics services provided by 3PL providers so that they can increase their supply chains’ efficiency and concentrate on their core competencies (Vaidyanathan 2005). For example, many online sellers prefer to stock their products at a 3PL distribution center so that they can reduce processing time for shipment after customers place orders. 3PL providers have evolved to provide a full set of integrated logistics activities such as transportation, warehousing, freight consolidation and distribution, rate negotiation and logistics information systems (Rabinovich et al. 1999, Sink et al. 1997). Moreover, E-commerce has induced 3PL providers to increase their investment in central warehouses. According to North American Industry Classification System (NAICS), the 3PL providers with warehousing and storage services belong to Warehousing and Storage industry (NAICS Code: 493). The data from the Bureau of Labor Statistics (BLS) in US shows that, from January 2001 to October 2012, the number of employees in Warehousing and Storage increased by 31% while in its NAICS super sector, Transportation and Warehousing, the increase was only 4.3%. The data from Bureau of Economic Analysis (BEA)
in US also shows that, from 2001 to 2011, the GDP added from Warehousing and Storage industry increased by 80.1% while from Transportation and Warehousing sector GDP only increased by 48.0%. These changes in the 3PL industry have increased the efficiency of traditional supply chains, and consequently drawn the attention of the academic community to the problems faced by 3PL providers that provide warehousing and transportation services.

We study the pricing problem for 3PL providers that provide the comprehensive services including warehousing and less-than-truckload (LTL) transportation. This work is motivated by a local 3PL provider whose customers include local manufacturers, wholesalers and retailers. Customers negotiate with the 3PL provider on the price and delivery date for shipments, and then stock their freight at the 3PL provider’s warehouse until the freight is delivered. The 3PL provider transports the customers’ shipments by trucks. The 3PL provider’s transportation network includes multiple routes each of which connects a city in a neighboring state/ province. For each route, there are a certain number of trucks running between the destination city and the home city. The 3PL provider operates its transportation routes with its business partners, usually other 3PL providers located in the destination cities. When the 3PL provider operates a route with a partner, it shares the trucks running on that route and the warehouses at both ends of the route. The trucks usually belong to independent transporters, and for each route, the 3PL provider signs long-term contracts with these transporters. If the 3PL provider temporarily increases the trucks for a route, the temporarily added trucks will charge higher transportation costs than the long-term contracted trucks. Thus, for any route, the daily transportation capacity, which is determined by the number of long-term contracted trucks, is fixed, and if the freight scheduled to be transported in a day is more than the 3PL provider’s daily transportation capacity, the 3PL provider incurs a penalty for hiring temporary trucks. With the daily arrival of hundreds of shipping orders from different customers with different quantities and delivery date preferences, it is difficult for the 3PL provider to satisfy all the customers’ first-best delivery date
requirements with the limited transportation capacity.

We develop a dynamic pricing strategy (DPS) that links the price for each shipping order with the 3PL provider’s capacity usage and inventory holding cost. In our DPS, different delivery dates are priced differently. Similar practices can often be found for firms offering standard orders and rush orders with different prices. However, this simple price differentiation in practice usually cannot promise an exact delivery date. Rather, at best each type of order is promised an expected leadtime or a maximum leadtime that is computed based on theory of priority queues. Because no exact delivery date is promised, this pricing strategy can only be applied in a business environment where customers are not completely strict about just-in-time (JIT) delivery. For the 3PL providers, the pricing problem is more complex when considering the 3PL provider’s transportation capacity, inventory cost and the customers’ delivery date requirements that are more strict and more unpredictable. Thus, a well-designed DPS has the potential to increase the profit of the 3PL provider and its customers, which consequently increases the profit of the entire supply chain.

In this chapter, we compute the optimal price for each delivery-date option through stochastic nonlinear programming (SNLP). In our SNLP model, we use a multinomial logit (MNL) function-based method to estimate customer’s choices of delivery dates, and the maximum-likelihood estimation (MLE) is employed for regressing the MNL function. The MNL function has been extensively used in marketing science to predict customers’ choices from multiple substitutable alternatives. The MNL function and MLE-based method is applicable to our problem for two reasons. First, because customers can obtain different values by choosing to deliver their freight on different delivery dates, we can treat the multiple options of delivery dates as alternatives differentiated in ‘quality’. Second, because 3PL providers usually receive hundreds of orders from local customers each day, large enough samples can be collected for regression.

In a MNL function, the term “taste coefficients” is often used to describe the weights
a consumer puts on different factors of an alternative, and a consumer’s taste coefficients
determine its utility from choosing different alternatives (Train 2003). We use taste coeffi-
cients to describe the weights that determine the choices of the 3PL provider’s customers
although the customers are usually firms. One difficulty in using a standard MNL func-
tion is the assumption that customers are homogeneous in their taste coefficients, which is
not true in our problem where customers often have different delivery date preferences. A
mixed logit function is often employed to solve a discrete choice problem when customers’
taste coefficients are heterogeneous. In a mixed logit function, the probability of choosing
an alternative is computed by integrating the standard logit probability over the density of
the taste coefficients. Thus, the employment of a mixed logit function requires the distri-
butions of all taste coefficients to be known. However, because in our problem, the factors
that affect customer choices are outside the information available to the 3PL providers, the
distributions of customers’ taste coefficients are unknown. Hence, we develop a new variant
of a MNL function for our problem where customers have heterogeneous taste coefficients
with unknown distributions.

Our pricing model can also be employed in other manufacturing or service-providing
settings which have similar features as the 3PL provider studied in this chapter. For example,
a make-to-order company produces goods or parts for its customers that are heterogeneous
in their delivery date preferences, and developing a DPS incorporating its inventory and
production status can increase its profit and optimize the allocation of its resources (i.e.,
inventory and production capacities).

Because the DPS enables the 3PL provider to coordinate its transportation schedule with
its customers’ production schedule, it actually gives a marketing-based solution for supply
chain scheduling. However, little research in supply chain scheduling studies cooperation
between manufacturers and 3PL providers that provide comprehensive warehousing and
LTL transportation services. Moreover, in review articles by Maloni and Carter (2006),
Selviaridis and Spring (2007) and Marasco (2008), no work covers 3PL provider pricing. Lukassen and Wallenburg (2010) reviewed research related to pricing strategies for 3PL services but only in the context of the pricing of logistic services where the 3PL and its client are involved in a long-term contract. Carter et al. (1995) studied how the pricing in LTL transportation industry affects the manufacturers’ lot-sizing decision but not scheduling. We found no literature that studies dynamic delivery-date-concerned pricing for a 3PL provider with warehousing and LTL transportation services as we study in present work.

By adopting our DPS to maximize profit while incorporating its holding cost and transportation capacity, the 3PL provider induces some flexible customers to choose other delivery dates if there is not enough capacity on their first-best delivery dates. Thus, the DPS is actually a priority dispatching strategy when the 3PL provider’s transportation capacity is limited. Priority pricing relates to the literature on pricing/queuing models as reviewed in Section 2.3. As stated in Section 2.3 in the research on priority pricing in priority queues, the firm does not guarantee the delivery dates, and customers make decisions only based on the expected leadtime. Because there is no delivery date guarantee, the firm does not need to control the number of customer choices for each delivery date. Thus, the pricing methods proposed by the work on priority pricing are only applicable in the case where there is no need to induce customer choices or to control the arrival rate. Hence the priority pricing mechanism cannot be applied for the 3PL providers whose customers specify their delivery date requirements and their freight has to be delivered on specified delivery dates.

To support the techniques we employ, a number of articles use logit functions in pricing problems such as price optimization for multiple substitutable products or services. Aydin and Ryan (2000), Hopp and Xu (2005), Maddah and Bish (2007) studied the assortment and pricing problem where the customer choices of substitutable products are modeled by logit functions. Aydin and Porteus (2008) and Dong et al. (2009) studied the pricing and inventory control of products where the customer choices of products are modeled by logit functions.
Huang (2002) studied the pricing of a simple two-mode transportation system where the customer choices of transportation modes are modeled by a MNL function. Ferrer et al. (2010) studied the pricing of product or service bundles where the customer choices of bundles is modeled by a MNL function. Li and Huh (2011) studied a pricing problem by modeling the profit as function of prices and the customer choices of products are modeled by a nested logit function. Rodríguez and Aydın (2011) studied the pricing of configurable products where the customers’ preference on each configuration is modeled by a logit function. So far as we can find, the literature studying pricing problems with logit models does not consider customer heterogeneity in tastes and thus only standard multinomial logit function or nested logit function are employed. A review of research using mixed logit function to model customer choices when the customers are heterogeneous in their taste coefficients can be found from the survey conducted by Hensher and Greene (2003). However, we cannot find literature that uses mixed logit function for price optimization.

The remainder of our chapter is organized as follows. In Section 4.2 we describe the problem and provide notation for our pricing model. In Section 4.3, we study the pricing model for the case where the quantity of freight received daily is deterministic. This case is studied to clarify the analyses of the proposed adjusted MNL model. Section 4.3 is divided into three subsections. First, we develop the adjusted MNL in Section 4.3.1; Second, we develop a SNLP model (SNLP-D) with the adjusted MNL to solve the optimal freight rates for the 3PL provider in Section 4.3.2; finally, we analyze the performance of the developed SNLP model with the adjusted MNL through simulation experiments in Section 4.3.3. In Section 4.4, we present the pricing model for the case where the quantity of shipments received daily is random. In Section 4.5, we compare the DPS with a static pricing strategy (SPS) in terms of the 3PL’s profit, customer and social welfare, and the shadow prices of holding cost and transportation capacity. Section 4.6 contains our concluding remarks.
4.2 Problem Description

We suppose that the 3PL provider’s business is managed in days. In each day, the 3PL provider receives shipping orders from different arriving customers, each of which has a freight to ship. We assume that each freight received by the 3PL provider is featured by a quantity, e.g., weights, sizes, etc., which is an independent and identically distributed (i.i.d.) random variable denoted by \( q : q \in \mathbb{R}^+ \).

The 3PL provider offers \( T \) delivery date options. If a customer chooses delivery date \( t : t \in \{1, \ldots, T\} \), then its freight should be delivered at the \( t \)th day after it arrives at the 3PL provider. The 3PL provider quotes different freight rates for different delivery date options, and the freight rate for a delivery date is defined as the price for delivering a unit quantity of freight at that delivery date including the storage and transportation. The charge for a customer order can be obtained by multiplying the freight rate by the quantity of the freight. We define a freight quote as a list of freight rates for all the delivery date options, and a freight quote is denoted by \( \vec{p} : \vec{p} \in \mathbb{R}^T \) where the \( t \)th element, \( p_t \), is the freight rate for delivery date \( t \). In our DPS, the 3PL provider updates the freight quote daily, and at the beginning of each day, the freight quote is computed based on the 3PL provider’s inventory and transportation capacity usage status. Once a freight quote is decided, it is not changed during the day. The 3PL provider does not quote dynamically by freight order because it is common that the customers inquire about the freight rate first and place orders later after comparing prices among different 3PL providers. It is not realistic for the 3PL provider to change freight quote within a day.

We assume that the 3PL provider incurs a holding cost, denoted by \( h : h \in \mathbb{R}^+ \), for each freight unit stocked in its warehouse for each day. The 3PL provider can dynamically adjust the setting of \( h \) based on its inventory status. The 3PL provider manages its business for each route independently. It stocks the freight for the same route in the same area, and the freight quotes for different routes are different. Thus, we can simplify the problem to a
single-route problem.

In the single route problem, we define the 3PL’s daily transportation capacity, denoted by $c : c \in \mathbb{R}^+$, as the maximum quantity of freight that can be transported each day. The 3PL provider’s daily transportation capacity is fixed because the number of trucks going to the destination city every day is fixed. If freight scheduled to be transported in a day is more than the 3PL provider’s daily transportation capacity, the 3PL provider incurs a penalty of $\omega : \omega \in \mathbb{R}^+$ for each freight unit over the daily transportation capacity. The penalty accounts for the 3PL provider’s loss for hiring temporary trucks or violating the promised delivery dates. As there might be freight received in earlier days already scheduled to be shipped or preserved capacity for future arriving orders on a certain date, the available transportation capacity for each delivery date option may vary. We assume that at the beginning of each day, the 3PL provider counts the available transportation capacity for each delivery date option. We denote the available transportation capacities for delivery date $t$ by $c_t$, and $\bar{c} : \bar{c} \in \mathbb{R}^T$ represents the $c_t$’s for all $t \in \{1, \ldots, T\}$.

Following the trend in the literature, we use the term ‘taste coefficients’ to refer to the weights a customer gives to the factors which determine its utility of choosing an alternative. That is, the taste coefficients determine a customer’s preference on each alternative. We assume that the 3PL provider’s customers have two taste coefficients for each delivery date option: valuation and price sensitivity. We define a customer’s valuation of a delivery date option $t$, denoted by $v_t$, as its valuation for transporting a unit of its freight on delivery date $t$, and its valuation, denoted by $\bar{v} : \bar{v} \in \mathbb{R}^T$, represents the $v_t$’s for all $t \in \{1, \ldots, T\}$. Similarly, we define a customer’s price sensitivity for delivery date $t$, denoted by $\alpha_t$, as its marginal disutility caused by a unit increase of freight rate for delivery date $t$, and its price sensitivity, denoted by $\bar{\alpha} : \bar{\alpha} \in \mathbb{R}^T$, represents the $\alpha_t$’s for all $t \in \{1, \ldots, T\}$. We assume that the values of $\bar{v}$ and $\bar{\alpha}$ for each customer are determined by details outside the information available to the 3PL provider. Hence, we assume that for each customer, each element in $\bar{v}$
and $\bar{\alpha}$ is an i.i.d. random variable with an unknown distribution.

We define a customer’s utility (or profit) from choosing leadtime $t$, denoted by $\xi(t)$, as

$$\xi(t) = v_t - \alpha_t p_t \epsilon,$$

where $\epsilon : \epsilon \in \mathbb{R}$ is a random component of a customer’s profit, possibly as a result of unobserved variables or decision makers’ errors and biases (Su 2008). According to Guadagni and Little (1983), $\epsilon$ is distributed with a standard double exponential (Gumbel extreme value) distribution. A more general form of a customer’s net gain would include a further option-associated parameter and a scale parameter. However, any option-associated parameter, even one dependent on $t$, can be absorbed into $v_t$ without loss of generality and the scaling of net gain is arbitrary because the scale is eliminated when computing the probability of choosing a delivery date using a MNL function. A customer chooses the delivery date option which maximizes its net gain, or rejects without purchase if $\xi(t) < 0$ for all $t \in \{1, \ldots, T\}$. If a customer rejects, then its net gain is 0. Because a customer’s net gain for choosing delivery date $t$ is random and depends on the freight quote $\vec{p}$, we denote the probability of a customer choosing delivery date $t$ by $P_t(\vec{p})$.

We denote the total number of customers arriving each day by $N$. For clarity, we start the analyses of the pricing model by assuming $N$ and each customer’s freight quantity, $q$, to be deterministic. Later we extend the pricing model to incorporate random $N$ and $q$. When $N$ is random, we assume that $N$ is i.i.d. in each day.

4.3 Pricing with Deterministic $N$ and $q$

We construct a stochastic nonlinear programming (SNLP) model to compute the optimal freight quote which maximizes the 3PL provider’s expected profit when $N$ and $q$ are deterministic. In order to build the SNLP, we have to determine the probability of a customer choosing delivery date $t$ given a freight quote $\vec{p}$, $P_t(\vec{p})$. 
4.3.1 Computing $P_t(\bar{p})$ using MLE+MNL based Method

We use a logit function to model the probability of a customer choosing each delivery date. Because the standard MNL model requires customers to be homogeneous in their taste coefficients, it is not applicable in our problem where customers’ taste coefficients, $\boldsymbol{v}$ and $\boldsymbol{\alpha}$, are heterogeneous. In the family of logit functions, the mixed logit function is often employed in the case where customers are heterogeneous in taste coefficients. However, for our problem, the mixed logit function is not applicable because the customers’ taste coefficients follow unknown distributions. In addition, when employing a mixed logit function, the probability of choosing a delivery date option should be approximated through simulation by repetitively drawing the taste coefficients from their distributions [Hensher and Greene 2003]. Because we need to use a heuristic method to solve our SNLP and the distribution of customer choices has to be computed under each freight quote obtained from the iteration, the procedure of approximating a mixed logit function makes it impossible to solve the optimal freight quote within a reasonable computing time. Based on this analysis, we develop an adjusted MNL model for our problem where customers are heterogeneous in their taste coefficients.

In order to develop the adjusted MNL model for our problem, we first use a standard MNL function to model customer choices by treating heterogeneous customers as homogeneous ones whose valuation and price sensitivity are deterministic, denoted by $\bar{v}^S$ and $\bar{\alpha}^S$, respectively. Then, based on $\bar{v}^S$ and $\bar{\alpha}^S$, we use the standard MNL to obtain an approximate probability of a customer choosing delivery date $t$ under freight quote $\bar{p}$, denoted by $P_t^S(\bar{p})$, as

$$P_t^S(\bar{p}) = \frac{e^{v_t^S - \alpha_t^S p_t}}{1 + \sum_{i=1}^P e^{v_i^S - \alpha_i^S p_i}} \quad t = 1, 2, \ldots, T. \quad (4.1)$$

$\bar{v}^S$ and $\bar{\alpha}^S$ can be estimated using MLE. When using MLE, we suppose that $P_t^S(\bar{p})$ is equal to the true probability of a customer choosing delivery date $t$ under freight quote $\bar{p}$. Then if $v_t^S$ and $\alpha_t^S$ are estimated based on the customer choices within $K$ recent days, where $K$ is a
positive integer, the likelihood, denoted by $\mathcal{L}$, is constructed as

$$
\mathcal{L} = \prod_{k=1}^{K} \left[ \prod_{t=1}^{T} \mathbb{P}^S_t(\bar{p}_k) \right]^{Q_{0k}} \prod_{t=1}^{T} \mathbb{P}^S_t(\bar{p}_k)^{Q_{tk}},
$$

(4.2)

where $\bar{p}_k$ is the freight quote in the $k$th day in $K$ days, $Q_{tk}$ is the quantity of freight placed in the $k$th day with delivery date choice $t$, and $Q_{0k}$ is quantity of freight arriving in the $k$th day rejecting all delivery date options. To simplify the form of likelihood, we take natural logs on both sides of (4.2). After simple algebra, we can obtain

$$
\ln \mathcal{L} = \sum_{k=1}^{K} \left[ Q_{0k} \ln \left[ 1 - \sum_{t=1}^{T} \mathbb{P}^S_t(\bar{p}_k) \right] + \sum_{t=1}^{T} Q_{tk} \ln \mathbb{P}^S_t(\bar{p}_k) \right].
$$

(4.3)

In (4.3), substituting $\mathbb{P}^S_t(\bar{p}_k)$ with the forms of $v^S_t$ and $\alpha^S_t$ as defined in (4.1), we can have $\ln \mathcal{L}$ as a function of $v^S_t$ and $\alpha^S_t$ for all $t \in \{1, \ldots, T\}$. Then we find the optimal setting of $v^S_t$ and $\alpha^S_t$ which maximizes $\ln \mathcal{L}$. Because of the complexity of solving the first-order condition of (4.3), the optimal setting of $v^S_t$ and $\alpha^S_t$ for each $t \in \{1, \ldots, T\}$ is obtained using numerical methods.

Because customers are actually heterogeneous in their taste coefficients, we can expect that $\mathbb{P}^S_t(\bar{p})$ computed in (4.1) may not equal to the true probability of a customer choosing delivery date $t$ given freight quote $\bar{p}$ and the estimations of $v^S_t$ and $\alpha^S_t$ for all $t \in \{1, \ldots, T\}$. Thus, after running the MLE with likelihood in (4.3), we use the adjusted MNL function to compute the probability of a customer choosing delivery date $t$ in our SNLP. Let $\mathbb{P}^A_t(\bar{p}_k)$ be the probability obtained using the adjusted MNL function, and we set $\mathbb{P}^A_t(\bar{p}_k)$ to be

$$
\mathbb{P}^A_t(\bar{p}) = \frac{e^{v^S_t - \alpha^S_t p_t + r_t(p_t)}}{1 + \sum_{i=1}^{T} e^{v^S_i - \alpha^S_i p_i + r_i(p_i)}} \quad t = 1, 2, \ldots, T.
$$

(4.4)

In (4.4), we add an adjusting term, $r_t(p_t)$, to the customers’ valuation for each delivery date option to mitigate the errors of prediction caused by customers’ heterogeneity. Note that in our adjusted MNL function, we set the adjusting term for delivery date option $t$ to be a function of $p_t$ by ignoring the impacts from the freight rates for other delivery date options. From the simulation experiments in Section 4.3.3, we can see that this treatment does not affect the accuracy of our adjusted MNL function in predicting customer choices.
The probability of a customer rejecting all delivery date options can be obtained as

\[ 1 - \sum_{i=1}^{T} \mathbb{P}_i^A(\bar{p}) = \frac{1}{1 + \sum_{i=1}^{T} e^{v_i^S - \alpha_i^S p_i + r_i(p_i)}}. \]  

(4.5)

From (4.4) and (4.5), we have

\[ e^{v_i^S - \alpha_i^S p_t + r_t(p_t)} = \frac{\mathbb{P}_i^A(\bar{p})}{1 - \sum_{i=1}^{T} \mathbb{P}_i^A(\bar{p})}. \]

Then we can obtain \( r_t(p_t) \) as

\[ r_t(p_t) = \ln \left[ \frac{\mathbb{P}_t^A(\bar{p})}{1 - \sum_{i=1}^{T} \mathbb{P}_i^A(\bar{p})} \right] - v_t^S + \alpha_t^S p_t. \]  

(4.6)

In (4.6), by supposing that \( \mathbb{P}_t^A(\bar{p}) \) computes the true probability of a customer choosing delivery date \( t \), we substitute \( \mathbb{P}_t^A(\bar{p}) \) and \( 1 - \sum_{i=1}^{T} \mathbb{P}_i^A(\bar{p}) \) with \( Q_{tk} \) and \( Q_{0k} \), respectively, for all \( k \in \{1, \ldots, K\} \). Then, we can obtain the realized values of \( r_t(p_t) \) for all days in the sales record as

\[ r_{tk}(p_{tk}) = \ln \left[ \frac{Q_{tk}}{Q_{0k}} \right] - v_t^S + \alpha_t^S p_{tk} \quad k = 1, 2, \ldots, K, \]

where \( p_{tk} \) is the freight rate for delivery date option \( t \) in the \( k \)th day, and \( r_{tk}(p_{tk}) \) is the realized value of \( r_t(p_t) \). With \( p_{tk} \) and \( r_{tk}(p_{tk}) \) for all \( k \in \{1, \ldots, K\} \) known, we can regress \( r_t(p_t) \) when the form of \( r_t(p_t) \) is given with a series of to-be-determined coefficients.

In order to regress \( r_t(p_t) \) for all \( t \in \{1, 2, \ldots, T\} \), we first need to settle the form of \( r_t(p_t) \). We set \( r_t(p_t) \) to a polynomial because according to the Weierstrass approximation theorem, within a closed domain, any continuous function can be approximated by a polynomial as closely as desired. When \( r_t(p_t) \) is set to a polynomial, the optimal order setting should depend on the fitness of the regressed function. From simulation experiments, we observe that \( r_t(p_t) \) usually can fit closely to the realized values when set to be a cubic function as

\[ r_t(p_t) = \beta_{t0} + \beta_{t1} p_t + \beta_{t2} p_t^2 + \beta_{t3} p_t^3. \]  

(4.7)

where \( \beta_{t0}, \beta_{t1}, \beta_{t2} \) and \( \beta_{t3} \) are the coefficients of \( r_t(p_t) \) to be determined through regression. Note that according to Weierstrass approximation theorem, the regressed polynomial can
only approximate \( r_t(p_t) \) when \( p_t \) is within the interval where \( p_{tk} \) is distributed. Thus, when using the adjusted MNL function to predict customer choices, \( p_t \) should also be constrained in the same interval. We use a simplified simulation as an example to show the regression result of \( r_t(p_t) \) when the form of \( r_t(p_t) \) goes from linear to cubic.

Example

In our example, the 3PL provider only offers a sole delivery date option \((T = 1)\). Thus, we eliminate the subscript ‘\( t \)’ for clarity. We generate the sales record of 30 days, each of which includes 100 customers each with freight quantity drawn from \( U(0.5, 1.5) \). For each of the 30 days, the freight rate is drawn from \( U(0, 8) \). We test the cases where a customer’s valuation, \( V \), and price sensitivity, \( \alpha \), follow different distributions:

**Case 1:** The customers are homogeneous, i.e., \( V = 5 \) and \( \alpha = 1 \).

**Case 2:** The customers are heterogeneous in \( V \), i.e., \( V \sim U(0, 10) \) and \( \alpha = 1 \).

**Case 3:** The customers are heterogeneous in \( \alpha \), i.e., \( V = 5, \alpha \sim U(0, 2) \).

**Case 4:** The customers are heterogeneous in \( V \) and \( \alpha \), i.e., \( V \sim U(0, 10), \alpha \sim U(0, 2) \).

In Figure 4.1, we display the realized values of \( r(p) \) and the regressed curve in each case. We observe that with homogeneous customers, \( r(p) \approx 0 \) for all \( p \), which indicates that the standard MNL can accurately predict the probability of customer choices when the customers are homogeneous. In Cases 2-4, it is obvious that \( r(p) \) is not always 0 at different \( p \). The regression curves for linear regression, quadratic regression and cubic regression are displayed in each chart. The cubic regression obtains a close fit to the realized values of \( r(p) \) in all the cases.

With the adjusted MNL model, we increase the prediction accuracy of customer choices of different delivery dates under different freight quotes \( \tilde{p} \), which is critical for computing the optimal freight quote using our SNLP pricing model.
4.3.2 The SNLP Model with Deterministic $N$ and $q$

We name the SNLP for computing the freight quote under deterministic $N$ and $q$ by SNLP-D. We use a random variable $N_t^\bar{p}$ to denote the realized number of customers choosing delivery date option $t$ under $\bar{p}$. The constraint $qN_t^\bar{p} \leq c_t$ has to be made for all $t \in \{1, \ldots, T\}$ to ensure that the 3PL provider has enough available capacity to guarantee the customers’ freight can be delivered on the chosen delivery dates. From our assumption on transportation capacity, if $qN_t^\bar{p} > c_t$ for some $t$, then a penalty, $\omega$, is incurred for each unit of capacity shortage. Denoting the probability density function (pdf) of $N_t^\bar{p}$ by $f_t^N(N_t^\bar{p}, \bar{p})$, the total penalty incurred to the 3PL provider, denoted by $\Omega^D(\bar{p})$, can be computed as

$$\Omega^D(\bar{p}) = \omega \sum_{t=1}^{T} \left[ \int_{c_t/q}^{+\infty} f_t^N(N_t^\bar{p}, \bar{p}) [qN_t^\bar{p} - c_t] dN_t^\bar{p} \right].$$

Because the probability of a customer choosing delivery date $t$ is $\mathbb{P}_t(\bar{p})$, then with $\Omega^D(\bar{p})$, SNLP-D can be formed as

$$\textbf{SNLP-D:} \quad \max_{\bar{p}} \quad qN \sum_{t=1}^{T} [p_t - ht] \mathbb{P}_t(\bar{p}) - \Omega^D(\bar{p}), \quad \forall \quad p_t \geq 0 \quad t = 1, 2, \ldots, T.$$
In SNLP-D, \( ht \) computes the holding cost incurred by a freight unit if delivery date of the freight is \( t \). The objective function is to maximize the 3PL provider’s expected profit subject to a nonnegative freight rate for each delivery date option. Because the 3PL provider can dynamically adjust the settings of the holding cost, \( h \), and the available transportation capacity for all delivery date options, \( \vec{c} \), it can compute the dynamic freight quote based on its current warehouse and transportation status.

In order to solve SNLP-D, it is necessary to know the form of \( f_t^N(N_t^\vec{p}, \vec{p}) \) in (4.8) for all \( t \in \{1, \ldots, T\} \), which is determined by the distribution of \( N_t^\vec{p} \) given \( \vec{p} \). Let \( \hat{t} \) be a customer’s random choice of leadtime. When the probability of a customer choosing delivery date \( t \), \( \mathbb{P}_t(\vec{p}) \), is estimated for all \( t \in \{1, \ldots, T\} \), we can obtain that given the value of \( N \), \( N_t^\vec{p} \) asymptotically follows \( \mathcal{B}(N, \mathbb{P}_t(\vec{p})) \), which is a binomial distribution with \( N \) independent yes/no experiments, each of which yields success with probability \( \mathbb{P}_t(\vec{p}) \). When \( N \) is large enough (\( > 20 \)), the binomial distribution can be well approximated by the normal distribution \( \text{Box et al.} \, 1978 \). Thus, we can approximate the distribution of \( N_t^\vec{p} \) to the normal distribution as

\[
N_t^\vec{p} \sim \mathcal{N}(N\mathbb{P}_t(\vec{p}), N\mathbb{P}_t(\vec{p})[1 - \mathbb{P}_t(\vec{p})]), \quad t = 1, 2, \ldots, T. \tag{4.10}
\]

From (4.10), \( f_t^N(N_t^\vec{p}, \vec{p}) \) for all \( t \in \{1, \ldots, T\} \) can be formed, and hence SNLP-D can be solved.

4.3.3 Simulation with Deterministic \( N \) and \( q \)

In this section, we design a simulation experiment based on a 3PL provider that provides comprehensive services including warehousing and LTL transportation. To be consistent with the assumptions in Section 4.3, we set the total quantity of shipments in each day to be deterministic. For clarity, we use SNLP-D(S) and SNLP-D(A) to refer to the SNLPs with the standard MNL function and the adjusted MNL function, respectively. Through the simulation experiment, we compare the 3PL provider’s expected profit under the freight quotes computed by SNLP-D(S) and SNLP-D(A).
In our simulation we only simulate the 3PL provider’s business for a single route. Because the total number of customers arriving each day, \( N \), and the quantity of each shipment, \( q \), are assumed to be deterministic in this section, we set \( N = 200 \) and \( q = 1 \) in our simulation.

The 3PL provider offers five delivery date options, i.e., \( T = 5 \). Each generated customer has a first-best delivery date, denoted by \( t^* \). In order to simulate the case where customers’ first-best delivery dates follow a near normal distribution, we set the probabilities of \( t^* = 1 \), \( t^* = 2 \), \ldots, \( t^* = 5 \) to be 0.05, 0.35, 0.3, 0.25, 0.05, respectively. Note that we ran our simulations by setting the customers’ first-best delivery dates to follow different distributions, even including the case where \( t^* = 1 \) for all customers. However, because the results for different settings are qualitatively the same, we omit the results for other settings for brevity.

In our simulation, for each delivery date \( t \), we set a customer’s price sensitivity, \( \alpha_t \), to be distributed from \( U(0.5, 1.5) \) and its valuation, \( v_t \), is determined by

\[
v_t = v^* - \tau|t^* - t|,
\]

where \( v^* : v^* \in \mathbb{R}^+ \) is the customer’s valuation if the shipment is delivered at its first-best delivery date, and \( \tau : \tau \in \mathbb{R}^+ \) is the constant marginal cost incurred by a unit deviation of \( t \) from the customer’s first best delivery date. We name \( \tau \) as the customer’s delivery-date sensitivity. In the simulation, we set \( v^* \) to be distributed from \( U(4, 6) \). For the setting of \( \tau \), we consider 4 different cases, in each of which the mean and the variance of \( \tau \) are high or low compared to \( v^* \).

**Case HH:** The mean and the variance of \( \tau \) are both high, i.e., \( \tau \sim U(0, 4) \).

**Case HL:** The mean of \( \tau \) is high, but the variance of \( \tau \) is low, i.e., \( \tau \sim U(1.8, 2.2) \).

**Case LH:** The mean of \( \tau \) is low, but the variance of \( \tau \) is high, i.e., \( \tau \sim U(0, 2) \).

**Case LL:** The mean and the variance of \( \tau \) are both Low, i.e., \( \tau \sim U(0.8, 1.2) \).

The performance of our pricing method is not affected when the distribution of \( v^* \) and \( \tau \) is scaled to a different magnitude because as we stated in Section 4.2, the scale effect
is eliminated in the MNL function. Thus, we use this four-case simulation to illustrate the performance of our pricing model when the 3PL provider faces different situations in practice. As for the other parameters for the simulation, we set the 3PL’s holding cost for each unit freight for each day, $h$, to be 0.5, and set the penalty for a unit capacity shortage, $\omega$, to be 5.

Using the method introduced in Section 4.3.1, we first run MLE based on the standard MNL function to estimate customers’ valuation and price sensitivity for all delivery date options, $\vec{v}^S$ and $\vec{\alpha}^S$. We generate the sales records for 30 days ($K = 30$), and in each day, the price for each delivery date option is a random real number drawn from $U(0, 10)$. The simulation is repeated 100 times. In Table 4.1, the averages of the estimated $v^S_t$ and $\alpha^S_t$ for all $t \in \{1, \ldots, T\}$ are displayed with the standard deviations in parentheses. We observe from Table 4.1 that based on our setting, the standard deviations are small compared to the average of the estimations for each parameter. This shows that the 3PL provider can obtain stable estimates for each parameter based on a 30-day sales record. In the remainder of our simulation, we use the averages of $v^S_t$ and $\alpha^S_t$ for all $t \in \{1, \ldots, T\}$ to compute the optimal freight quote in each case.

<table>
<thead>
<tr>
<th>Case</th>
<th>Taste coefficient</th>
<th>$t = 1$</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>HH:</td>
<td>$v^S_t$</td>
<td>1.24 (0.07)</td>
<td>2.07 (0.05)</td>
<td>2.24 (0.04)</td>
<td>1.84 (0.06)</td>
<td>1.06 (0.08)</td>
</tr>
<tr>
<td></td>
<td>$\alpha^S_t$</td>
<td>0.55 (0.02)</td>
<td>0.53 (0.02)</td>
<td>0.57 (0.01)</td>
<td>0.53 (0.02)</td>
<td>0.54 (0.02)</td>
</tr>
<tr>
<td>HL:</td>
<td>$v^S_t$</td>
<td>1.08 (0.10)</td>
<td>2.09 (0.08)</td>
<td>2.34 (0.05)</td>
<td>1.81 (0.08)</td>
<td>0.80 (0.12)</td>
</tr>
<tr>
<td></td>
<td>$\alpha^S_t$</td>
<td>0.54 (0.03)</td>
<td>0.53 (0.02)</td>
<td>0.59 (0.02)</td>
<td>0.53 (0.03)</td>
<td>0.52 (0.04)</td>
</tr>
<tr>
<td>LH:</td>
<td>$v^S_t$</td>
<td>2.40 (0.07)</td>
<td>2.99 (0.05)</td>
<td>3.16 (0.05)</td>
<td>2.85 (0.06)</td>
<td>2.25 (0.08)</td>
</tr>
<tr>
<td></td>
<td>$\alpha^S_t$</td>
<td>0.64 (0.02)</td>
<td>0.64 (0.02)</td>
<td>0.67 (0.01)</td>
<td>0.65 (0.02)</td>
<td>0.64 (0.02)</td>
</tr>
<tr>
<td>LL:</td>
<td>$v^S_t$</td>
<td>2.45 (0.06)</td>
<td>3.12 (0.06)</td>
<td>3.30 (0.04)</td>
<td>2.96 (0.05)</td>
<td>2.30 (0.08)</td>
</tr>
<tr>
<td></td>
<td>$\alpha^S_t$</td>
<td>0.66 (0.02)</td>
<td>0.66 (0.02)</td>
<td>0.69 (0.01)</td>
<td>0.66 (0.01)</td>
<td>0.66 (0.03)</td>
</tr>
</tbody>
</table>

After estimating $\vec{v}^S$ and $\vec{\alpha}^S$, we regress $r_t(p_t)$ following the steps introduced in Section 4.3.1. From the sales record, we obtain $p_t$ and the realized values of $r_t(p_t)$ for all $t \in \{1, \ldots, T\}$.
from (4.6) for all days, and then we regress \( r_t(p_t) \) for all \( t \in \{1, \ldots, T\} \) to a cubic function as in (4.7). The regression results are displayed in Table 4.2. As introduced in Section 4.3.1, the regressed polynomial can only approximate \( r_t(p_t) \) when \( p_t \) is within the interval \([0, 10]\) because \( p_t \) was drawn from \( U(0, 10) \) for each \( t \in \{1, \ldots, T\} \) in our simulation.

Table 4.2: The regressed \( r_t(p_t) \) for each delivery date option

<table>
<thead>
<tr>
<th>Case</th>
<th>Coefficient</th>
<th>( t = 1 )</th>
<th>( t = 2 )</th>
<th>( t = 3 )</th>
<th>( t = 4 )</th>
<th>( t = 5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>HH:</td>
<td>( \beta_{t0} )</td>
<td>-0.0464</td>
<td>-0.14794</td>
<td>-0.1116</td>
<td>-0.18236</td>
<td>0.016286</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t1} )</td>
<td>0.01322</td>
<td>0.025461</td>
<td>0.008036</td>
<td>0.025943</td>
<td>-0.0083</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t2} )</td>
<td>-0.00082</td>
<td>-0.00725</td>
<td>-0.00087</td>
<td>-0.00545</td>
<td>0.000102</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t3} )</td>
<td>9.25E-06</td>
<td>0.000529</td>
<td>-7.45E-06</td>
<td>0.000358</td>
<td>2.90E-06</td>
</tr>
<tr>
<td>HL:</td>
<td>( \beta_{t0} )</td>
<td>-0.24412</td>
<td>-0.23465</td>
<td>-0.24304</td>
<td>-0.24504</td>
<td>-0.40412</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t1} )</td>
<td>-0.06511</td>
<td>0.017836</td>
<td>0.041437</td>
<td>-0.02875</td>
<td>-0.01261</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t2} )</td>
<td>0.012195</td>
<td>-0.00513</td>
<td>-0.0076</td>
<td>0.004793</td>
<td>0.003159</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t3} )</td>
<td>-0.00066</td>
<td>0.00037</td>
<td>0.000373</td>
<td>-0.00019</td>
<td>-0.00018</td>
</tr>
<tr>
<td>LH:</td>
<td>( \beta_{t0} )</td>
<td>0.598496</td>
<td>0.112013</td>
<td>0.239677</td>
<td>0.213847</td>
<td>0.564281</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t1} )</td>
<td>-0.00701</td>
<td>0.035354</td>
<td>-0.002</td>
<td>0.021598</td>
<td>0.048857</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t2} )</td>
<td>0.00391</td>
<td>-0.00944</td>
<td>0.000481</td>
<td>-0.00521</td>
<td>-0.01446</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t3} )</td>
<td>-0.00035</td>
<td>0.000639</td>
<td>-2.44E-05</td>
<td>0.000315</td>
<td>0.001087</td>
</tr>
<tr>
<td>LL:</td>
<td>( \beta_{t0} )</td>
<td>0.49876</td>
<td>0.08811</td>
<td>0.284832</td>
<td>0.187254</td>
<td>0.531275</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t1} )</td>
<td>-0.00138</td>
<td>0.042352</td>
<td>-0.0449</td>
<td>0.008621</td>
<td>-0.00555</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t2} )</td>
<td>0.001185</td>
<td>-0.00989</td>
<td>0.010241</td>
<td>-0.00161</td>
<td>-0.00462</td>
</tr>
<tr>
<td></td>
<td>( \beta_{t3} )</td>
<td>-8.44E-05</td>
<td>0.000629</td>
<td>-6.48E-04</td>
<td>9.34E-05</td>
<td>0.000526</td>
</tr>
</tbody>
</table>

With the obtained adjusting term \( r_t(p_t) \) for all \( t \in \{1, \ldots, T\} \) in each case, we can compute the adjusted probabilities of customer choices with the adjusted MNL function from (4.4). In order to test the superiority of adjusted MNL model, we compute the optimal freight quotes computed by SNLP-D(S) and SNLP-D(A), and then compare the profits the 3PL provider can realize using the two freight quotes. Because the inaccuracy of SNLP-D’s objective function in estimating the 3PL provider’ expected profit is caused by customer heterogeneity rather than the setting of available capacity, we only show the results when the available capacity, \( \vec{c} \), is randomly set to \( \vec{c} = (30, 36, 40, 44, 50) \). When repeating the simulation with different settings of available capacities, we obtain the same evidence that shows the superiority of adjusted MNL model. The optimal freight quotes obtained by SNLP-D(S) and SNLP-D(A) are displayed in Table 4.3.
We use $\vec{p}^S$ and $\vec{p}^A$ to denote the optimal freight quotes obtained by SNLP-D(S) and SNLP-D(A), respectively, and $N_{i}^{p^S}$ and $N_{i}^{p^A}$ are the realized number of customers choosing delivery date $t$ under $\vec{p}^S$ and $\vec{p}^A$, respectively. In order to test if the standard MNL function and the adjusted MNL function can accurately predict the probabilities of customer choices when the customers are heterogeneous, we first simulate the true expected number of customers choosing delivery date $t$, denoted by $E(N_{i}^{p^S})$ and $E(N_{i}^{p^A})$, and then we compare $N_{i}^{p^S}(\vec{p}^S)$ and $N_{i}^{p^A}(\vec{p}^A)$ with $E(N_{i}^{p^S})$ and $E(N_{i}^{p^A})$ for all $t \in \{1, \ldots, T\}$. If the standard MNL function and the adjusted MNL function can both accurately predict the probabilities of customer choices, then we should have $N_{i}^{p^S}(\vec{p}^S)$ and $N_{i}^{p^A}(\vec{p}^A)$ statistically equal to $E(N_{i}^{p^S})$ and $E(N_{i}^{p^A})$, respectively. We calculate $E(N_{i}^{p^S})$ and $E(N_{i}^{p^A})$ by averaging $N_{i}^{p^S}$ and $N_{i}^{p^A}$ obtained from 100 runs of the simulation program in which random customers are generated for a typical day and face freight quotes of $\vec{p}^S$ and $\vec{p}^A$, respectively. The results of comparing the accuracies of standard MNL function and the adjusted MNL function in predicting the probabilities of customer choices are displayed in Table 4.4. From Table 4.4, we observe that based on t-tests with 95% confidence level, most $E(N_{i}^{p^S})$ values are significantly different from the paired values computed by $N_{i}^{p^S}(\vec{p}^S)$, whereas all the $E(N_{i}^{p^A})$ values are not significantly different from the paired values computed by $N_{i}^{p^A}(\vec{p}^A)$. From this result, we conclude that when customers are heterogeneous our adjusted MNL function yields better predictions of customer choice distributions.
Table 4.4: Comparing the accuracies of standard MNL function and the adjusted MNL function in predicting the probabilities of customer choices†

<table>
<thead>
<tr>
<th>Model</th>
<th>Case</th>
<th>t = 1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNLP-D(S)</td>
<td>HH</td>
<td>*(21.60, 21.66)</td>
<td>(30.36, 27.02)</td>
<td>*(21.20, 20.84)</td>
<td>(7.13, 7.73)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LH</td>
<td>(25.13, 23.34)</td>
<td>(31.09, 28.23)</td>
<td>*(25.36, 25.44)</td>
<td>(10.74, 11.68)</td>
<td></td>
</tr>
<tr>
<td>SNLP-D(A)</td>
<td>HH</td>
<td>*(21.83,21.52)</td>
<td>*(29.61,30.07)</td>
<td>*(20.51,20.47)</td>
<td>*(7.50,7.61)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HL</td>
<td>*(18.49,18.24)</td>
<td>*(29.85,29.77)</td>
<td>*(20.66,20.95)</td>
<td>*(5.80,5.78)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LH</td>
<td>*(24.87,24.56)</td>
<td>*(30.61,30.72)</td>
<td>*(25.06,24.77)</td>
<td>*(11.44,11.71)</td>
<td></td>
</tr>
</tbody>
</table>

† In each parentheses, the first value is $N \mathbb{P}(\tilde{p}^S)$ (or $N \mathbb{P}(\tilde{p}^A)$), and the second value is the simulated $E(N_t^{\tilde{p}^S})$ (or $E(N_t^{\tilde{p}^A})$).

* Indicates that $N \mathbb{P}(\tilde{p}^S)$ (or $N \mathbb{P}(\tilde{p}^A)$) is not significantly different from the paired simulated value of $E(N_t^{\tilde{p}^S})$ (or $E(N_t^{\tilde{p}^A})$) based on t-test with 95% confidence level.

It is straightforward that because the adjusted MNL function increases the prediction accuracy of the distribution of customer choices, the freight quote obtained by SNLP-D(A) results in higher expected profit for the 3PL provider. Table 4.5 compares the 3PL provider’s expected profits under the freight quotes computed from SNLP-D(S) and SNLP-D(A). Denoting the 3PL’s true expected profits under $\tilde{p}^S$ and $\tilde{p}^A$ by $E(\pi(\tilde{p}^S))$ and $E(\pi(\tilde{p}^A))$, respectively, we obtain $E(\pi(\tilde{p}^S))$ and $E(\pi(\tilde{p}^A))$ by averaging the profits obtained from 100 runs of the simulation program in which random customers are generated for a typical day and face freight quotes of $\tilde{p}^S$ and $\tilde{p}^A$, respectively. In Table 4.5, we observe that by increasing the prediction accuracy of the distribution of customer choices, SNLP-D(A) can more accurately predict the 3PL provider’s expected profit and consequently computes a better freight quote. In contrast, because SNLP-D(S) less accurately predicts the 3PL provider’s expected profit, it computes a worse freight quote although it obtains a higher objective value in each case.

4.4 Pricing with Random $N$ and $q$

A more realistic case is when the total quantity of freight received each day, $N$, and the quantity of each shipment order, $q$, are random and follow particular distributions. Here
Table 4.5: Comparing the 3PL provider’s expected profits under the freight quotes computed from SNLP-D(S) and SNLP-D(A)†

<table>
<thead>
<tr>
<th>Case</th>
<th>SNLP-D(S)</th>
<th>SNLP-D(A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HH:</td>
<td>(490.64, 462.00)</td>
<td>*(471.74, 472.86)</td>
</tr>
<tr>
<td>HL:</td>
<td>(484.58, 429.66)</td>
<td>*(442.34, 441.85)</td>
</tr>
<tr>
<td>LH:</td>
<td>(568.76, 539.13)</td>
<td>*(555.90, 555.92)</td>
</tr>
<tr>
<td>LL:</td>
<td>(567.69, 531.84)</td>
<td>*(547.87, 543.76)</td>
</tr>
</tbody>
</table>

† In each parenthesis, the first value is the expected profit computed from the objective function SNLP-D(S) (or SNLP-D(A)), and the second value is the simulated value.

* Indicates that the expected profit computed from the objective function SNLP-D(S) (or SNLP-D(A)) is not significantly different from the paired simulated value of the objective function based on t-test with 95% confidence level.

we propose a pricing model which solves the near optimal freight quote when \( N \) and \( q \) are random, and develop a method to solve the proposed pricing model.

4.4.1 The SNLP Model with Random \( N \) and \( q \)

We use SNLP-R to name our SNLP model for computing the near optimal freight quote with random \( N \) and \( q \). In this case, we use \( Q^\rho_t \) to denote the total quantity of freight from customers choosing delivery date option \( t \) given freight quote \( \rho_t \). Then similar as in SNLP-D, the total penalty for capacity shortage, denoted by \( \Omega^R(\rho) \), can be obtained as

\[
\Omega^R(\rho) = \omega \sum_{t=1}^{T} \left[ \int_{c_t}^{+\infty} f_{Q_t}(Q^\rho_t, \rho) [Q^\rho_t - c_t] dQ^\rho_t \right]
\]  

(4.11)

where \( f_{Q_t}(Q^\rho_t, \rho) \) is the pdf of \( Q^\rho_t \) under \( \rho \). Then with \( \Omega^R(\rho) \), SNLP-R can be obtained as

**SNLP-R:** \( \max_{\rho} E(q)E(N) \sum_{t=1}^{T} [p_t - ht]P_t(\rho) - \Omega^R(\rho), \quad \rho \geq 0 \quad t = 1, 2, \ldots, T. \)

In SNLP-R, \( E(q) \) and \( E(N) \) are the expected values of \( q \) and \( N \), respectively. In order to solve SNLP-R, we need to determine the forms of \( f_{Q_t}(Q^\rho_t, \rho) \) for all \( t \in \{1, \ldots, T\} \) in (4.11).

Because the value of \( Q^\rho_t \) is mutually determined by the number of customers, the quantity of each customer’s order and the choices of delivery dates, when \( N \) and \( q \) are random, deriving the analytical form of \( f_{Q_t}(Q^\rho_t, \rho) \) is complex. In the literature, histogram or Kernel density estimation can be a general approach to determine \( f_{Q_t}(Q^\rho_t, \rho) \) when \( N \) and \( q \) are both random.
(Green et al. 1988). In this general approach, given any setting of $\vec{p}$, the realized values of $Q_t^{\vec{p}}$ are obtained from large number of repeated simulations and the curve of $f_t(Q_t^{\vec{p}}, \vec{p})$ in $Q_t^{\vec{p}}$ is obtained based on the distribution of the realized values of $Q_t^{\vec{p}}$. This general approach can obtain the approximate curve of $f_t(Q_t^{\vec{p}}, \vec{p})$ in $Q_t^{\vec{p}}$ given any setting of $\vec{p}$ when $N$ and $q$ follow any distributions. However, because the curve of $f_t(Q_t^{\vec{p}}, \vec{p})$ changes with $\vec{p}$, we have to re-draw the curve of $f_t(Q_t^{\vec{p}}, \vec{p})$ every time $\vec{p}$ is changed. Thus, using this general approach may result in unreasonably long computing time when we search the optimal $\vec{p}$ for SNLP-R using heuristics where $\vec{p}$ is iteratively changed and the curve of $f_t(Q_t^{\vec{p}}, \vec{p})$ has to be drawn for each instance of $\vec{p}$. In order to solve SNLP-R within reasonable computing time, we develop a method to find an asymptotic form of $f_t(Q_t^{\vec{p}}, \vec{p})$. Because our method requires a normally distributed $N$, here we only examine the realistic case where $N$ is normally distributed, i.e., $N \sim N(\mu_N, \sigma_N^2)$ where $\mu_N$ and $\sigma_N$ are the mean and standard deviation of $N$, respectively.

Because for any $t$ we can use a Bernoulli distributed variable (0 or 1) to represent if a customer chooses delivery date $t$, then it is straightforward that when $N$ is random, the number of customers choosing delivery date $t$ under $\vec{p}$, $N_t^{\vec{p}}$, is the sum of a random number of i.i.d. Bernoulli distributed random variables. According to Robbins (1948), when $N$ is normally distributed, $N_t^{\vec{p}}$ asymptotically follows the normal distribution as

$$N_t^{\vec{p}} \sim N\left(\mu_{N_t^{\vec{p}}}, \sigma_{N_t^{\vec{p}}}^2\right),$$

where

$$\mu_{N_t^{\vec{p}}} = \mu_N \mathbb{P}_t(\vec{p})$$

and

$$\sigma_{N_t^{\vec{p}}}^2 = \mu_N \mathbb{P}_t(\vec{p}) \left[1 - \mathbb{P}_t(\vec{p})\right] + \sigma_N^2 \left[\mathbb{P}_t(\vec{p})\right]^2.$$ 

Similarly, the total quantity of freight from the customers choosing delivery date $t$, $Q_t^{\vec{p}}$, can be treated as the sum of a random number ($N_t^{\vec{p}}$) of i.i.d. random variables ($q$). Let $\mu_q$ and $\sigma_q$ be the mean and standard deviation of $q$. Again, we can have $Q_t^{\vec{p}}$ asymptotically following
the normal distribution as

\[ Q_t^\bar{p} \sim \mathcal{N}\left(\mu_{Q_t^\bar{p}}, \sigma_{Q_t^\bar{p}}^2\right), \quad (4.12) \]

where

\[ \mu_{Q_t^\bar{p}} = \mu_q \mu_{N_t^\bar{p}} = \mu_q \mu_{N_t^\bar{p}} \mathbb{P}_t(\bar{p}) \]

and

\[ \sigma_{Q_t^\bar{p}}^2 = \mu_{N_t^\bar{p}} \sigma_q^2 + \mu_q^2 \sigma_{N_t^\bar{p}}^2 + \mu_{N_t^\bar{p}} \mathbb{P}_t(\bar{p}) \sigma_q^2 + \mu_q^2 [\mu_{N_t^\bar{p}} \mathbb{P}_t(\bar{p}) [1 - \mathbb{P}_t(\bar{p})] + \sigma_{N_t^\bar{p}}^2 \mathbb{P}_t(\bar{p})]^2]. \]

From (4.12), we can obtain the asymptotic forms of \( f_t^Q(Q_t^\bar{p}, \bar{p}) \) for all \( t \in \{1, \ldots, T\} \), and then SNLP-R is solvable.

4.4.2 Simulation with Random \( N \) and \( q \)

In our simulation experiment to test the pricing model when \( N \) is random, we set \( N \) to be distributed from \( \mathcal{N}(200, 50^2) \) and set \( q \) to be distributed from \( \mathcal{U}(0.5, 1.5) \). The other parameters are the same as the ones used in the simulation experiment for the case with deterministic \( N \) and \( q \).

We use SNLP-R(S) and SNLP-R(A) to refer to the SNLP-R with a standard MNL function and an adjusted MNL function, respectively. The optimal freight quotes obtained by SNLP-R(S) and SNLP-R(A) are displayed in Table 4.6.

Table 4.6: The freight quotes for different delivery date options when \( N \) and \( q \) are random

<table>
<thead>
<tr>
<th>Model</th>
<th>Case</th>
<th>( t = 1 )</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNLP-R(S)</td>
<td>HH:</td>
<td>4.97</td>
<td>6.18</td>
<td>6.01</td>
<td>6.06</td>
<td>6.51</td>
</tr>
<tr>
<td></td>
<td>HL:</td>
<td>4.86</td>
<td>6.17</td>
<td>5.98</td>
<td>6.03</td>
<td>6.53</td>
</tr>
<tr>
<td></td>
<td>LH:</td>
<td>5.65</td>
<td>6.25</td>
<td>6.16</td>
<td>6.07</td>
<td>6.44</td>
</tr>
<tr>
<td></td>
<td>LL:</td>
<td>5.57</td>
<td>6.21</td>
<td>6.17</td>
<td>6.05</td>
<td>6.37</td>
</tr>
<tr>
<td>SNLP-R(A)</td>
<td>HH:</td>
<td>4.92</td>
<td>5.94</td>
<td>5.92</td>
<td>6.00</td>
<td>6.42</td>
</tr>
<tr>
<td></td>
<td>HL:</td>
<td>4.52</td>
<td>5.8</td>
<td>5.79</td>
<td>5.81</td>
<td>6.35</td>
</tr>
<tr>
<td></td>
<td>LH:</td>
<td>5.58</td>
<td>6.15</td>
<td>6.14</td>
<td>6.02</td>
<td>6.44</td>
</tr>
<tr>
<td></td>
<td>LL:</td>
<td>5.40</td>
<td>6.09</td>
<td>6.11</td>
<td>5.92</td>
<td>6.30</td>
</tr>
</tbody>
</table>

Similar to Section 4.3.3 in Table 4.7, we compare the 3PL provider’s expected profits computed from SNLP-R(S) and SNLP-R(A) with the true ones, \( E(\pi(\bar{p}^S)) \) and \( E(\pi(\bar{p}^A)) \),
which are obtained by averaging the realized profits obtained from 100 runs of the simulation program in which random customers are generated for a typical day and face price quotes of $\vec{p}^S$ and $\vec{p}^A$, respectively. We observe that the result is the same as the case when $N$ is deterministic. That is, by increasing the prediction accuracy of the distribution of customer choices, SNLP-R(A) computes a better freight quote that leads to a higher expected profit for the 3PL provider.

Table 4.7: Comparing the 3PL provider’s expected profits under the freight quotes computed from SNLP-R(S) and SNLP-R(A)†

<table>
<thead>
<tr>
<th>Case</th>
<th>SNLP-R(S)</th>
<th>SNLP-R(A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HH:</td>
<td>(482.50, 453.72)</td>
<td>*(463.10, 461.98)</td>
</tr>
<tr>
<td>HL:</td>
<td>(476.16, 435.11)</td>
<td>*(432.68, 438.70)</td>
</tr>
<tr>
<td>LH:</td>
<td>(556.03, 516.83)</td>
<td>*(546.06, 557.76)</td>
</tr>
<tr>
<td>LL:</td>
<td>(554.30, 524.51)</td>
<td>*(535.03, 538.12)</td>
</tr>
</tbody>
</table>

† In each parenthesis, the first value is the expected profit computed from the objective function SNLP-R(S) (or SNLP-R(A)), and the second value is the simulated $E(\pi(\hat{p}^S))$ (or $E(\pi(\hat{p}^A))$).  
* Indicates that the expected profit computed from the objective function SNLP-R(S) (or SNLP-R(A)) is not significantly different from the paired simulated value of $E(\pi(\hat{p}^S))$ (or $E(\pi(\hat{p}^A))$) based on t-test with 95% confidence level.

4.5 Compare the Dynamic Pricing Strategy with the Static Pricing Strategy

We demonstrate the superiority of our DPS using the proposed pricing model by comparing it with a SPS where the 3PL provider does not dynamically change the freight rate for each delivery date option. In the SPS, the 3PL provider set a fixed freight rate, denoted by $p^F$, for each freight unit. If a customer with $q$ units of freight chooses the delivery date $t$, then it is charged $q[p^F + th]$. We find the optimal setting of $p^F$ in the SPS through simulation. That is, we repeat the simulation program with different settings of $p$, and use the $p$ which results in the 3PL provider’s maximum expected profit (simulated by averaging the profits obtained in 500 runs of simulation) as its optimal setting.

In our simulation experiment, we simulate 100 days of the 3PL provider’s business. We
set the 3PL provider’s daily transportation capacity, denoted by \( c \), to \( c = 120 \). As in the previous simulation experiments, we set the 3PL provider to provide five delivery date options \( (T = 5) \). The orders received in each day affect the available capacity in the coming days. Let Day \( k \) be the current day. If a customer arriving in Day \( k \) chooses delivery date option \( t \), then the available transportation capacity in Day \( k + t \) will be reduced by the quantity of that customer’s freight. When the DPS is employed, a certain transportation capacity should be preserved in Day \( k + t \) for orders arriving after Day \( k \) and before Day \( k + t \). In present work we simply preserve the transportation capacity of \( c/T \) in Day \( k + t \) for the orders arriving in each day after Day \( k \) and before Day \( k + t \), and so \( c[t - 1]/T \) of transportation capacity should be preserved in Day \( k + t \) for the orders arriving in all days after Day \( k \) and before Day \( k + t \). The parameters of the 3PL’s unit holding cost, unit penalty for capacity shortage and the distributions of customers’ parameters, are set the same as in Section 4.4.2.

4.5.1 Compare the 3PL Provider’s Expected Profit, Customer Welfare and Social Welfare

The results of the simulation experiment are displayed in Table 4.8. From the simulation, we first obtained the optimal freight rate settings from the SPS. Then we simulated the 100 days of the 3PL provider’s business and compare the utilization of its transportation capacity and the profits under the two pricing strategies. We use \( \pi_{100} \) to denote the profit obtained by the 3PL provider over 100 days. We see that using the DPS, the utilization of the 3PL provider’s transportation capacity is substantially increased, and consequently its profit is increased. The superiority of the DPS increases when there are more customers with high delivery-date sensitivity. The 3PL provider’s expected profit is increased most using the DPS in the HL case where the customers mainly have high delivery-date sensitivity.

We compare customer and social welfare under the SPS and the DPS in Table 4.9. We sum the net gain of all the customers arriving over 100 days. The social welfare is obtained by adding up the simulated customer welfare and the 3PL provider’s profit in Table 4.8. We observe that with the DPS, the customer and social welfare are substantially increased.
Table 4.8: Comparing the 3PL’s expected profit obtained from the two pricing strategies

<table>
<thead>
<tr>
<th>Case</th>
<th>Static pricing strategy</th>
<th>Dynamic pricing strategy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Capacity utilization</td>
<td>π₁₀₀</td>
</tr>
<tr>
<td>HH</td>
<td>79.3% 43066</td>
<td>91.3% 47776(+10.9%)</td>
</tr>
<tr>
<td>HL</td>
<td>66.3% 37282</td>
<td>87.8% 44158(+18.4%)</td>
</tr>
<tr>
<td>LH</td>
<td>85.7% 53037</td>
<td>93.8% 55273(+4.2%)</td>
</tr>
<tr>
<td>LL</td>
<td>86.8% 52161</td>
<td>94.9% 54023(+3.6%)</td>
</tr>
</tbody>
</table>

Hence, with the DPS, the 3PL provider, customers and society are better off.

Table 4.9: Comparing customer welfare and social welfare under two pricing strategies

<table>
<thead>
<tr>
<th>Case</th>
<th>Static pricing strategy</th>
<th>Dynamic pricing strategy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Customer welfare</td>
<td>Social welfare</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HH</td>
<td>24433</td>
<td>67500</td>
</tr>
<tr>
<td>HL</td>
<td>21192</td>
<td>58474</td>
</tr>
<tr>
<td>LH</td>
<td>27608</td>
<td>80646</td>
</tr>
<tr>
<td>LL</td>
<td>26997</td>
<td>78959</td>
</tr>
</tbody>
</table>

4.5.2 Compare the Shadow Prices of Reducing Holding Cost and Increasing Transportation Capacity

We compare the shadow prices of reducing holding cost and increasing transportation capacity in the two pricing strategies to analyze how the different pricing strategies influence the 3PL provider’s decisions about long-term investment.

We define the holding cost shadow price as the increase of daily profit that is caused by a unit reduction of the unit holding cost, $h$. In our simulation, we gradually change the setting of $h$ from 2 to 0.1 with the step, denoted by $\Delta h$, of $\Delta h = 0.1$. We simulate the 3PL provider’s profits under different settings of $h$ and then compute the holding cost shadow price when the current holding cost is $h$, denoted by $\gamma^H(h)$, from the simulated profits when the holding cost are $h$ and $h - \Delta h$. Using $\pi^{100}(h)$ to denote the 3PL provider’s total profit over 100 days obtained from the simulation when the holding cost is $h$, we obtain $\gamma^H(h)$ as

$$\gamma^H(h) = \frac{\pi^{100}(h - \Delta h) - \pi^{100}(h)}{\Delta h} \times \frac{1}{100}.$$
We keep the 3PL provider’s daily transportation capacity constant at \( c = 120 \), and the simulation results are displayed in Figure 4.2. From Figure 4.2 we observe that when holding costs fall, the 3PL provider’s profit increases with both the DPS and the SPS, but the holding cost shadow price is higher with the DPS, indicating that with the DPS the 3PL provider has a higher incentive to reduce its holding cost. This is because that when the DPS is employed, the freight rate for each delivery date option is constrained by both the holding cost and the current available transportation capacity. When the unit holding cost is reduced, the holding cost constraint is weakened and so in the DPS the freight quote makes better use of the current available transportation capacity. Thus, when holding costs are reduced, the 3PL provider can obtain a greater increase in profit using the DPS.
Figure 4.2: The profit ($\pi_{100}$) and the shadow price of reducing holding cost ($\gamma^H$) when the current unit holding cost ($h$) is at different values in DPS and SPS.
We define the transportation capacity shadow price as the increase in daily profit that is caused by a unit increase in the 3PL provider’s daily transportation capacity. In our simulation, we gradually change the setting of daily transportation capacity, \( c \), from 65 to 160 with the step, denoted by \( \Delta c \), of \( \Delta c = 5 \). We simulate the 3PL provider’s profits under different settings of \( c \) and then compute the transportation capacity shadow price when the current daily transportation capacity is \( c \), denoted by \( \gamma^C(c) \), from the simulated profits when the transportation capacity are \( c \) and \( c + \Delta c \). Using \( \pi^{100}(c) \) to denote the 3PL provider’s total profit over 100 days obtained from the simulation when the daily transportation capacity is \( c \), we obtain \( \gamma^C(c) \) as

\[
\gamma^C(c) = \frac{\pi^{100}(c + \Delta c) - \pi^{100}(c)}{\Delta c} \times \frac{1}{100}.
\]

We keep the 3PL provider’s unit holding cost constant at \( h = 0.5 \), and the simulation results are displayed in Figure 4.3. From Figure 4.3, we observe that when daily production capacity increases, the 3PL provider’s profit increases with both the DPS and the SPS, but the transportation capacity shadow price is lower with the DPS, which indicates that with the DPS, the 3PL provider has a lesser incentive to increase its daily transportation capacity. This is because that with the DPS, the utilization of transportation capacity is higher which means that the 3PL provider quotes lower prices to most customers in the DPS so that more orders are placed. When the transportation capacity is increased, the 3PL provider further reduces the freight rate to attract more orders in both pricing strategies, but with the DPS, because the freight rates are already lower, the price elasticity of demand is lower, which means fewer orders are attracted with the DPS with the same reductions of freight rates. Thus, when the daily transportation capacity is increased, the 3PL provider obtains a lower increase in its profit using the DPS.
Figure 4.3: The profit ($\pi_{100}^1$) and the shadow price of increasing transportation capacity ($\gamma^C$) when the current daily transportation capacity ($c$) is at different values in DPS and SPS.
From the analyses of shadow prices in the two pricing strategies, we conclude that the 3PL provider has different investment incentives for holding costs and transportation capacity.

4.6 Chapter Conclusions

In this chapter we develop and study a DPS for a 3PL provider that provides comprehensive logistic services including warehousing and LTL transportation. We use SNLP models to solve the near optimal freight quote which includes freight rates for multiple delivery date options. These SNLP models capture the stochastic nature of our problem while considering the 3PL provider’s holding cost and transportation capacity. Of the two models, the SNLP-D model solves the freight quote for the case where the total quantity of the freight received each day is deterministic, whereas the SNLP-R model is for the more realistic case where the total quantity of the freight received each day is random. We presented the SNLP-D model in order to explain the superiority of our adjusted MNL function more clearly.

We use the MNL function to model customer choices of delivery date options. We cannot use a standard MNL function to model customer choices due to customer heterogeneity. We also cannot use a mixed logit function because we assume that the 3PL provider does not know the distributions of customers’ taste coefficients. Hence, we developed an adjusted MNL function to model customer choices. Through a series of simulation experiments, we show that the adjusted MNL function can accurately predict the distribution of customer choices when customers are heterogeneous. Consequently, we show that the 3PL provider can obtain a higher expected profit when computing freight quotes using our SNLP with the adjusted MNL function. We also design simulation experiments to compare the proposed DPS with a SPS. We show that with the proposed DPS, both the 3PL provider and its customers are better off and the 3PL provider has different investment incentives for the longer-term in the two pricing strategies.

Our work makes three novel contributions. First, by adopting our DPS to optimize its
price setting accounting for its current holding cost and available transportation capacity, the 3PL provider optimizes its resource allocation to customers that are heterogeneous in their delivery date preferences. As described in Section 4.1, the proposed pricing model can also be applied in other manufacturing or service-providing settings that have similar features as the 3PL provider we study. Second, we show that the adjusted MNL function we develop is capable of modelling customer choices when customers are heterogeneous, which indicates that our adjusted MNL function is good replacement for the mixed MNL function in the cases where the mixed MNL function is not applicable.

Finally, our comparisons of the proposed pricing strategies provide management insight for practitioners regarding decisions about employing and implementing a DPS. By comparing the shadow prices of reduced holding costs and increased transportation capacity, we show how practitioners should adjust their long-term investments in inventory and transportation capacity and to maximize profits with the proposed DPS. Through simulation, we demonstrated that with the proposed DPS, the 3PL provider cannot only increase its profit but in addition increase its customers’ profit as well. As the customers of the 3PL provider are usually manufacturers, distributors and retailers, we conclude that the profit of entire supply chain is increased when the proposed DPS is employed by the 3PL provider – essentially improving coordination across the supply chain through dynamic pricing.
Chapter 5

A Dynamic Pricing Strategy Incorporating Overtime

5.1 Background

In this chapter, we study the dynamic pricing strategy that accounts for the overtime cost incurred within OKP firms. Different from the previous chapters, in this chapter, we study two different types of dynamic pricing strategies: a discount-based dynamic pricing strategy and a premium-based dynamic pricing strategy, and we focus on how different dynamic pricing strategies perform as a decentralized coordination strategy to coordinate a multi-echelon supply chain.

Figure 5.1 shows an example of a multi-echelon OKP supply chain, each node of which represents either a manufacturer (M) or a customer (C). We can also treat some service providers, such as third-party-logistics (3PL) providers, as OKP firms because these service providers have to allocate their existing resources to deliver services to customers or further outsource to other service providers. Thus, OKP supply chains are frequently found in practice where a MTO firm orders customized parts from suppliers or outsources some operations to subcontractors.

This work is motivated by a local customized-window manufacturing firm, a typical OKP manufacturer, whose customers include long-term contracted customers and spot-market customers. Long-term-contracted customers make frequent orders and contribute the majority of profit to the firm. Because of their buying power, contracted customers enjoy a lower price. Compared to long-term-contracted customers, spot-market customers cannot guarantee steady demand and only make discrete orders. However, because of the large number of spot-market customers, the firm does not want to lose this market. The firm tries to offer just-in-time (JIT) delivery service to all its customers, and employs a static
pricing strategy in which the price of an order is independent from the firm’s production status and the customer’s delivery date choice. Hence, as demand fluctuates, the firm often incurs considerable overtime labor cost trying to guarantee the delivery times specified by its customers while some orders are still delayed. The same problem also exists in similar firms. For example, MetalFab Inc., that fabricates metal parts for General Electric, also accepts orders from other customers to fill excess capacity and increase revenue, but the leadtimes are often not guaranteed because the firm adopts a static pricing strategy which is independent from its production status [Hall et al., 2009]. Another example can be found in some 3PL service providers. They have long-term-contracted customers and spot-market customers, and freight is determined by size and destination, independent of the schedule status.

Because the pricing strategies for long-term-contracted customers are usually specified by the contracts, we concentrate on pricing strategies for spot-market customers. We propose two dynamic pricing strategies for spot-market orders: a discount-based dynamic pricing strategy (DDP) and a premium-based dynamic pricing strategy (PDP), where discount and premium are dynamically used in each strategy to affect customer of delivery date. To analyze the supply chain’s performance under different pricing strategies, we compare the
price-setting firm’s (or seller’s) profits and its customers’ profits under different pricing strategies. Our results provide a theoretical basis for MTO firms when they choose their optimal pricing strategies.

The remainder of this chapter is organized as follows. In Section 5.2, we describe the problem and provide our notation and assumptions; in Section 5.3, we model two dynamic pricing strategies together with the static pricing strategy and show the optimal prices for the firm under each pricing strategy; in Section 5.4, we propose a series of theorems which compares the static pricing strategy and the two dynamic pricing strategies. In Section 5.5, we present a simulation case study designed to follow the practical case that motivated our work. In our simulation, we test our analytical results in the context of a single-echelon supply chain and a two-echelon supply chain. We draw management implications of this research at the end of Section 5.5. Finally, a summary of conclusions and future research is presented in Section 5.6.

In the literature we can find that Hall et al. (2009) studied a case which is close to the one studied in this work. They studied a supplier that produces products for a core customer, and the supplier also accepts discrete orders from other customers to fill in the excess capacity. They developed a pricing/queuing model to examine different pricing and admission policies for discrete-order customers, and showed that a constant pricing scheme and a simple admission policy is close to optimal. However, the dynamic pricing literature assumed that the supplier does not provide JIT delivery services because the pricing/queuing models are only capable of dealing with a simple queue-like production system where all customers prefer an earlier delivery date. Because we study a firm with orders requiring JIT delivery, we developed different pricing strategies. We compare two dynamic pricing strategies, the DDP strategy and the PDP strategy, which are both common in practice. The DDP strategy, where the firm may give discounts to customers to affect their purchasing behaviors, has been extensively studied for an MTS supply chain. In contrast, we study the
influence of the two dynamic pricing strategies on the coordination of MTO supply chains.

5.2 Notation and assumptions

As described in Section 5.1, a decoupled MTO supply chain is a single-echelon which includes a sole price-setting MTO supplier and multiple customers. Because the orders from long-term-contracted customers are processed with higher priority and the pricing strategies for long-term-contracted customers are specified in the contracts rather than governed by the pricing strategies we model, we treat the long-term-contracted orders scheduled in the supplier’s production system as exogenous and the resources for processing long-term-contracted orders are pre-allocated. Thus, in order to study spot-market pricing strategies, we simplify the problem by assuming that the supplier only has spot-market customers after pre-allocating all the required resources for long-term-contracted customers. Hereafter, we mean ‘spot-market customers’ when we say ‘customers’ unless otherwise stated.

Customers and orders

Customers arrive with specifications for customization and delivery-date requirements for their orders. For simplicity, we use Order $j$ to denote the order from Customer $j$, and we simplify our notation by assigning one order per customer – this entails no loss of generality as for spot-market customers multiple orders are usually independent. A customer order is featured by a workload (in man-hours) that is determined by its quantity and customization requirements. For example, in the custom-window firm, orders include different quantities of widows with different customization requirements, and the workload of the order is the man-hours required for producing all the windows in the order. We use $l_j : l_j \in \mathbb{R}^+$ to denote the workload of Order $j$.

A customer reveals its delivery-date requirement by declaring its first-best delivery date for its order, and we use $d^*_j$ to denote the first-best delivery date of Order $j$. We define the
valuation rate of Order \( j \), denoted by \( v_j : v_j \in \mathbb{R}^+ \), as the maximum price Customer \( j \) would pay for a unit workload if Order \( j \) is delivered at \( d_j^* \). Based on this definition, the valuation of Order \( j \) can be obtained at \( v_j l_j \).

If an order cannot be delivered at the customer’s first-best delivery date, then the customer incurs a loss that is effectively the order’s delivery date sensitivity. Based on customers’ delivery-date sensitivity, the orders are categorized into three types: delivery-date-critical orders, delivery-date-flexible orders and delivery-date-unspecified orders. When an order is not delivered on the first-best delivery date, a delivery-date-critical customer suffers infinite cost, a delivery-date-flexible customer suffers a limited switching cost and a delivery-date-unspecified customer suffers zero cost (no delivery date specified). The delivery-date-unspecified orders are scheduled with lower a priority than other orders, which means that current waiting delivery-date-unspecified orders do not impact the scheduling and pricing of the arriving orders of the other two types. As the delivery-date-unspecified orders do not have a delivery date requirements, our dynamic pricing strategies do not apply to delivery-date-unspecified orders and we do not include such orders in our pricing model. However, to be realistic, in the simulation case study in Section 5.5 we also include delivery-date-unspecified orders. We use \( \beta \) to denote the probability of an order being delivery-date-critical, and so the probability of an order being delivery-date-flexible is \( 1 - \beta \).

To form our analytical model, we make the following assumptions about customers and customer orders:

**Assumption 5.1.** For any Order \( j \), its workload, \( l_j \), is an independent and identically distributed (iid) random variable, and we scale it so that an order’s expected workload equals to the unit value, i.e., \( E(l_j) = 1 \).

In Assumption 5.1 we ignore the economy of scale within the firm. That is, we ignore the reduction of processing time as the received orders accumulate. The highly customized orders received by the firm make it difficult for the firm to realize mass production. The
processing time of an order is estimated based on the workers’ standard working rate.

**Assumption 5.2.** For any Order $j$, $v_j$, is an iid random variable and $v_j \sim U[0, V]$.

Because it is defined that a customer’s valuation for Order $j$ is $v_j l_j$, then with Assumption 5.2, we have that the customer tends to have a higher valuation for an order with higher workload.

**Assumption 5.3.** For any delivery-date-flexible customer, if its order is not delivered on its first-best delivery date, then it incurs a switching cost of $c^A$ for each unit workload of its order, and $0 < c^A < V$.

The cost $c^A$ is effectively the delivery-date-flexible customers’ delivery date sensitivity. Note that we simplify the problem by Assumption 5.3 although in the literature it is usually assumed that a customer suffers a random cost if its first-best delivery date is not satisfied. Based on the case of our custom window firm, this assumption is fair for the following reasons. First, the supplier can usually deliver an order on the customer’s second-best delivery date if its first-best delivery date cannot be satisfied. Thus, $c^A$ actually measures a customer’s loss if its order is delivered on its second-best delivery date. A customer may suffer a greater loss than $c^A$ when its order is delivered on its even-less-preferred delivery dates, but we take those probabilities as insignificant. Second, as stated by the manager of marketing department in our custom-window firm, it is often the case that with small discounts customers are seldom induced to choose their second-best delivery date, but once the discount reaches a certain “threshold”, dramatic changes happen and most delivery-date-flexible customers choose their second-best delivery dates. This phenomenon implies that the delivery-date-flexible customers have similar delivery date sensitivities, $c^A$. It also implies that the firm can estimate the value of $c^A$ based on the threshold discount using the approach we propose.
The production system of the firm

The MTO supplier we model processes a large number of customer orders in a typical day, and if the production of an order is scheduled for a day, then it has to be started and finished within that day. The supplier does not maintain inventory and ships all the orders the same day they are processed. We ignore the transportation time for finished orders. Thus, in order to guarantee the JIT delivery service, the supplier schedules all orders to be processed on the delivery dates agreed on by customers when the orders are placed.

The firm’s daily production is capacitated. We define the production capacity of the supplier as the maximum workload (e.g., in man-hours) for each typical day without overtime, and the supplier’s available capacity on a given day can be obtained by subtracting from capacity the total workload of the orders already scheduled for that day. The supplier incurs overtime when it processes more workload than its capacity. If an order is scheduled for a day with enough available capacity, then only standard labor cost is incurred to process that order. Otherwise, an overtime labor cost is incurred on top of the standard one. In our analysis, the standard labor cost does not affect our results, thus we scale standard labor cost to be zero. We use $c^O \in \mathbb{R}^+$ to denote the firm’s overtime labor rate, which is the overtime cost incurred by the firm for each unit of overtime workload.

Suppose that Customer $j$ chooses delivery date $d_j$ for its order, Order $j$. We use $L(d_j)$ to denote the expected workload of orders which arrive after Order $j$ and before $d_j$, and also have the first-best delivery date $d_j$. If the available capacity in $d_j$ is no more than $L(d_j)$, then we say Order $j$ incurs overtime on its first-best delivery date. We use $\alpha_j$ to denote the probability that Order $j$ incurs overtime on its first-best delivery date.

We make the following assumptions on the firm’s production system to form our model.

**Assumption 5.4.** If an order requires any overtime to be processed, it is treated as fully processed with overtime.

We simplify our model assuming that if there is an order that is partially processed with
overtime, then we treat it as fully processed with overtime – we justify this by recognizing that with hundreds of orders in a day, the impact of an order partially processed with overtime versus fully processed with overtime is insignificant. Moreover, the supplier may choose this as a simplifying policy with regards to its workload.

**Assumption 5.5.** The overtime rate, $c^O$, is greater than the delivery-date-flexible customers’ delivery date sensitivity and less than the maximum possible customer valuation rate, $c^A < c^O < V$.

If $c^A \geq c^O$, then the delivery-date-flexible customers will always insist on their first-best delivery dates by paying the overtime cost, which makes delivery-date-flexible customers the same as delivery-date-critical customers. If $c^O \geq V$, then the supplier never makes profit on an order when the order incurs overtime.

**Assumption 5.6.** For any Order $j$, $\alpha_j$ is an iid random variable, and the expected value of $\alpha_j$ is $\alpha$ which is known to the supplier.

From Assumption 5.6 changing pricing strategy will not change the expected probability of an order incurring overtime on its first-best delivery date. This is true if in each day the available production capacity for processing spot-market orders is mainly determined by the schedule of long-term-contracted orders, which is treated as exogenous in this work.

### 5.3 Models of the Three Pricing Strategies

We model three pricing strategies: a static pricing strategy (SP), the discount-based dynamic pricing strategy (DDP) and the premium-based dynamic pricing strategy (PDP). When the firm quotes prices to customers, it also tells them the workloads and the prices for each unit workload of their orders, and then customers can calculate the total price of their order.
The Static Pricing Strategy (SP)

In SP, the supplier sets the price at the beginning of a decision-making cycle, e.g., the beginning of a season. Then the supplier sticks to the price until the next decision-making cycle. We use $p^S$ to denote the price for a unit workload quoted to customers in SP. Given Order $j$, the expected profit the supplier can make from Order $j$ in SP with $p^S$, denoted by $\pi^S_j(p^S)$, can be obtained as

$$\pi^S_j(p^S) = l_j p \left[1 - \frac{p^S}{V}\right] - \alpha_j l_j c^O \left[1 - \frac{p^S}{V}\right]. \tag{5.1}$$

In (5.1), the first term computes the supplier’s expected revenue obtained from Order $j$, and the second term computes the expected overtime cost incurred by Order $j$. From (5.1), we can obtain the supplier’s expected profit from any order in SP with $p^S$, denoted by $\pi^S(p^S)$, as

$$\pi^S(p^S) = E(\pi^S_j(p^S)) = E(l_j) p^S \left[1 - \frac{p^S}{V}\right] - E(\alpha_j) E(l_j) c^O \left[1 - \frac{p^S}{V}\right] = p^S \left[1 - \frac{p^S}{V}\right] - \alpha c^O \left[1 - \frac{p^S}{V}\right]. \tag{5.2}$$

Maximizing (5.2) with respect to $p^S$ by solving the first-order condition, we can obtain the optimal price setting in SP, denoted by $p^{S*}$, as

$$p^{S*} = \frac{V + \alpha c^O}{2}. \tag{5.3}$$

Equation (5.2)’s second-order condition with respect to $p^S$ ensures that $p^{S*}$ is a maximum:

$$\frac{\partial^2 \pi^S}{\partial (p^S)^2} = -\frac{2}{V} < 0.$$

The Discount-based Dynamic Pricing Strategy (DDP)

In DDP, when a customer’s first-best delivery date incurs overtime, the supplier offers a discount from the standard price if the customer chooses an alternative delivery date which does not incur overtime.
We use $p^D$ and $x^D$ to denote the standard price and the discount (per unit workload) quoted to customers in DDP. Given Order $j$, the expected profit the supplier can make from Order $j$ in DDP with $p^D$ and $x^D$, denoted by $\pi^D_j(p^D, x^D)$, can be obtained as

$$
\pi^D_j(p^D, x^D) = \frac{\alpha_j \beta}{j} \left[ p^D - c^O \right] \left[ 1 - \frac{p^D}{V} \right] + \alpha_j \left[ 1 - \beta \right] l_j \left[ p^D - x^D \right] \left[ 1 - \frac{p^D - x^D + c^A}{V} \right] 
$$

\begin{align}
\quad (i) \\
\quad (ii) \\
\quad (iii)
\end{align}

In (5.4), the three terms, $(i) - (iii)$, compute the supplier’s expected profit obtained from Order $j$ under three possibilities:

(i) Customer $j$ is delivery-date-critical and its first-best delivery date incurs overtime.

(ii) Customer $j$ is delivery-date-flexible and chooses an alternative delivery date when its first-best delivery date incurs overtime.

(iii) Customer $j$’s first-best delivery date does not incur overtime.

For (5.4) we impose the following condition:

$$
x^D \geq c^A. \quad \text{(IC)} \quad (5.5)
$$

The condition (5.5) is essentially an incentive compatibility (IC) condition: in order for delivery-date-flexible customers to have an incentive to choose alternative delivery dates, the discount cannot be less than the delivery-date-flexible customers’ delivery date sensitivity.

From (5.4), we can obtain the supplier’s expected profit from an order in DDP with $p^D$
and \( x^D \), denoted by \( \pi^D(p^D, x^D) \), as

\[
\pi^D(p^D, x^D) = E\left( \pi_j^D(p^D, x^D) \right)
= E(\alpha_j)\beta E(l_j)[p^D - c^O][1 - \frac{p^D}{V}] + E(\alpha_j)[1 - \beta]E(l_j)[p^D - x^D]\left[1 - \frac{p^D - x^D + c^A}{V}\right]
+ [1 - E(\alpha_j)]E(l_j)p^D\left[1 - \frac{p^D}{V}\right]
= \alpha \beta [p^D - c^O][1 - \frac{p^D}{V}] + \alpha [1 - \beta][p^D - x^D]\left[1 - \frac{p^D - x^D + c^A}{V}\right]
+ [1 - \alpha]p^D\left[1 - \frac{p^D}{V}\right].
\] (5.6)

Solving the first-order condition of (5.6) with respect to \( p^D \) and \( x^D \), we can obtain the solutions denoted by \( \hat{p}^D \) and \( \hat{x}^D \), respectively, as

\[
\hat{p}^D = \frac{[1 - \alpha + \alpha \beta]V + \alpha \beta c^O}{2[1 - \alpha + \alpha \beta]} \quad \text{and} \quad (5.7a)
\]

\[
\hat{x}^D = \frac{[1 - \alpha + \alpha \beta]c^A + \alpha \beta c^O}{2[1 - \alpha + \alpha \beta]} \quad \text{and} \quad (5.7b)
\]

By the second-order partial derivative test on (5.6) with respect to \( p^D \) and \( x^D \), we have that the Hessian matrix for (5.6) is negative definite. Thus, we conclude that \( \pi^D(p^D, x^D) \) is maximized at \( \hat{p}^D \) and \( \hat{x}^D \).

We use \( p^{D^*} \) and \( x^{D^*} \) to denote the optimal settings of \( p^D \) and \( x^D \) in DDP, respectively. If \( \hat{x}^D > c^A \), the IC constraint, (5.5), does not bind and so we have \( p^{D^*} = \hat{p}^D \) and \( x^{D^*} = \hat{x}^D \), respectively. Otherwise, the IC constraint binds, and then \( p^{D^*} \) and \( x^{D^*} \) can be solved by Lagrange multiplier method. Hence, in DDP, we have \( p^{D^*} \) and \( x^{D^*} \) as

\[
p^{D^*} = \begin{cases} V + \alpha \beta c^O + \alpha [1 - \beta]c^A \quad & \text{if } \hat{x}^D \leq c^A \\ \hat{p}^D \quad & \text{otherwise} \end{cases} \quad (5.8a)
\]

and

\[
x^{D^*} = \begin{cases} c^A \quad & \text{if } \hat{x}^D \leq c^A \\ \hat{x}^D \quad & \text{otherwise} \end{cases}, \quad (5.8b)
\]

85
The Premium-based Dynamic Pricing Strategy (PDP)

In PDP, when a customer chooses a delivery date that incurs overtime, the supplier adds a premium to the standard price for its order. We use \( p^P \) to denote the standard price and \( x^P \) to denote the premium for a unit workload quoted to customers in PDP. Given Order \( j \), the expected profit the supplier can make from Order \( j \) in PDP with \( p^P \) and \( x^P \), denoted by \( \pi^P_j(p^P, x^P) \), can be obtained as

\[
\pi^P_j(p^P, x^P) = \alpha_j \beta l_j [p^P - c^O + x^P] \left[ 1 - \frac{p^P + x^P}{V} \right] + \alpha_j [1 - \beta] l_j p^P \left[ 1 - \frac{p^P + c^A}{V} \right]
\]

\[
+ [1 - \alpha_j] l_j p^P \left[ 1 - \frac{p^P}{V} \right].
\]

(5.9)

In (5.9), the three terms, (i) – (iii), compute the expected profit the firm can obtain from Order \( j \) under the same three situations as (i) – (iii) in (5.4). Again, we impose the IC condition:

\[
x^P \geq c^A, \quad \text{(IC)}
\]

(5.10)

which is for delivery-date-flexible customers to have an incentive to choose alternative delivery dates, the premium cannot be less than the delivery-date-flexible customers’ delivery date sensitivity.

From (5.9), we can obtain the firm’s expected profit from an order in PDP with \( p^P \) and \( x^P \), denoted by \( \pi^P(p^P, x^P) \), as

\[
\pi^P(p^P, x^P) = E\left( \pi^P_j(p^P, x^P) \right)
\]

\[
= E(\alpha_j) \beta E(l_j) [p^P - c^O + x^P] \left[ 1 - \frac{p^P + x^P}{V} \right] + E(\alpha_j)[1 - \beta] E(l_j) p^P \left[ 1 - \frac{p^P + c^A}{V} \right]
\]

\[
+ [1 - E(\alpha_j)] E(l_j) p^P \left[ 1 - \frac{p^P}{V} \right]
\]

\[
= \alpha \beta [p^P - c^O + x^P] \left[ 1 - \frac{p^P + x^P}{V} \right] + \alpha [1 - \beta] p^P \left[ 1 - \frac{p^P + c^A}{V} \right]
\]

\[
+ [1 - \alpha] p^P \left[ 1 - \frac{p^P}{V} \right].
\]

(5.11)
Solving the first-order condition, we can obtain the solutions of $p^P$ and $x^P$, denoted by $\hat{p}^P$ and $\hat{x}^P$, respectively, as

\[
\hat{p}^P = \frac{[1 - \alpha \beta]V - \alpha[1 - \beta]c^A}{2[1 - \alpha \beta]} \quad \text{and} \quad (5.12a)
\]

\[
\hat{x}^P = \frac{[1 - \alpha \beta]c^O + \alpha[1 - \beta]c^A}{2[1 - \alpha \beta]}.
\quad (5.12b)
\]

By the second-order partial derivative test on (5.11) with respect to $p^P$ and $x^P$, we have that the Hessian matrix for (5.11) is negative definite. Thus, we conclude that $\pi^P(p^P, x^P)$ is maximized at $\hat{p}^P$ and $\hat{x}^P$.

We use $p^{P*}$ and $x^{P*}$ to denote the optimal settings of $p^P$ and $x^P$ in PDP, respectively. If $\hat{x}^P \geq c^A$, then the IC constraint, (5.10), does not bind and we have $p^{P*} = \hat{p}^P$ and $x^{P*} = \hat{x}^P$, respectively. Otherwise, the IC constraint binds, and then $p^{P*}$ and $x^{P*}$ can be solved by Lagrange multiplier method. Hence, in PDP, $p^{P*}$ and $x^{P*}$ can be obtained as

\[
p^{P*} = \begin{cases} 
V + \alpha \beta c^O - \alpha[1 + \beta]c^A / 2 & \text{if } \hat{x}^P \leq c^A \\
\hat{p}^P & \text{otherwise}
\end{cases}
\quad (5.13a)
\]

\[
x^{P*} = \begin{cases} 
c^A & \text{if } \hat{x}^P \leq c^A \\
\hat{x}^P & \text{otherwise}
\end{cases}.
\quad (5.13b)
\]

5.4 Main Results

In this section, we first propose a theorem to compare prices across the three pricing strategies. Then we propose two theorems that compare the supplier's profit and customers' profit under the three pricing strategies. Finally we provide a corollary to compare the profit of the supply chain (or system profit) under the three pricing strategies.

**Theorem 5.1.** In DDP and PDP, the price quotes satisfy:

a) In DDP, all customers are quoted lower prices than in SP, i.e., $p^{P*} < p^{S*}$. 

87
b) In PDP, if a customer chooses a delivery date that incurs overtime, then it is quoted a price higher than in SP; otherwise, it is quoted a price lower than in SP. That is, $p^P*$ and $x^P*$ satisfy $p^P* < p^S* < p^P* + x^P*$.

Proof: See Appendix A.1.

Theorem 5.1 compares the optimal price quotes between the dynamic pricing strategies and the static pricing strategy. Theorem 5.1 implies that all customers prefer DDP while only some the customers prefer PDP. This result has implications for implementing the dynamic pricing strategies discussed in Section 5.5.4.

Theorem 5.2. Under the optimal price settings in the three pricing strategies, the supplier obtains higher profit in dynamic pricing strategies than in the static pricing strategy. Of the two dynamic pricing strategies, the supplier obtains higher profit in PDP than in DDP. That is, $\pi^P(p^P*, x^P*) > \pi^D(p^D*, x^D*) > \pi^S(p^S*)$.

Proof: See Appendix A.2.

Theorem 5.2 shows the supplier’s preferences in the three pricing strategies. From Theorem 5.2 we can conclude that when extending the decoupled supply chain to a multi-echelon supply one, each supplier within the supply chain would choose its pricing strategies following the same preferences. In order to analyze if the supplier’ choices of pricing strategies leads to the optimal profit of the entire supply chain, in a decoupled supply chain, we compare total profit of the customers when the supplier employs different pricing strategies. If the best pricing strategy for the price-setting supplier also leads to the highest customer profit, then we can conclude that both the upstream and the downstream firms prefer that pricing strategy.

We evaluate customers’ profit by computing their expected net gain. In SP, the customers only choose their first-best delivery dates or leave without placing orders. A customer accepts the supplier’s price quote when it has nonnegative net gain from the purchase. Thus, we can
obtain a customer’s expected net gain under $p^S$, denoted by $w^S(p^S)$, as

$$w^S(p^S) = E(l_j) \int_{p^S}^{V} \frac{1}{V} [v_j - p^S] dv_j = \frac{[V - p^S]^2}{2V}. \tag{5.14}$$

In DDP, if Customer $j$ chooses its first-best delivery date, then its net gain is $l_j[v_j - p^D]$; otherwise, its net gain is $l_j[v_j - c^A - p^D + x^D]$. A customer only accepts the supplier’s price quote when the net gain is nonnegative. Hence, we can obtain a customer’s expected net gain in DDP with $p^D$ and $x^D$, denoted by $w^D(p^D, x^D)$, as

$$w^D(p^D, x^D) = \alpha \beta \int_{p^D}^{V} \frac{1}{V} [v_j - p^D] dv_j + \alpha [1 - \beta] \int_{c^A + p^D - x^D}^{V} \frac{1}{V} [v_j - c^A - p^D + x^D] dv_j = \frac{\alpha \beta [V - p^D]^2}{2V} + \alpha [1 - \beta] \frac{[V - c^A - p^D + x^D]^2}{2V} \tag{5.15}$$

In PDP, if Customer $j$ insists its first-best delivery date which incurs overtime, its net gain is $l_j[v_j - p^P - x^P]$; if it chooses an alternative delivery date, its net gain is $l_j[v_j - c^A - p^P]$; otherwise, if it chooses its first-best delivery date without incurring overtime, its net gain is $l_j[v_j - p^P]$. Hence, we can obtain the customer expected net gain in PDP with $p^P$ and $x^P$, denoted by $w^P(p^P, x^P)$, as

$$w^P(p^P, x^P) = \alpha \beta \int_{p^P + x^P}^{V} \frac{1}{V} [v_j - p^P - x^P] dv_j + \alpha [1 - \beta] \int_{c^A + p^P}^{V} \frac{1}{V} [v_j - c^A - p^P] dv_j + [1 - \alpha] E(l_j) \int_{p^P}^{V} \frac{1}{V} [v_j - p^P] dv_j = \frac{\alpha \beta [V - p^P - x^P]^2}{2V} + \alpha [1 - \beta] \frac{[V - c^A - p^P]^2}{2V} + [1 - \alpha] \frac{[V - p^P]^2}{2V}. \tag{5.16}$$

Theorem 5.3. Under the supplier’s optimal price quotes in the three pricing strategies, customers obtain higher profit in dynamic pricing strategies than in the static pricing strategy, and of the two pricing strategies, customers obtain higher profit in PDP than in DDP. That is, $w^P(p^{P*}, x^{P*}) > w^D(p^{D*}, x^{D*}) > w^S(p^{S*})$.

Proof: See Appendix A.3.
From Theorem 5.3, we have the same ranking of the three pricing strategies no matter whether sorted by the supplier’s profit or by customers’ profit. From this feature, we can conclude the supply chain’s preference over the three pricing strategies. We use $W_S(p^*_S), W_D(p^*_D, x^*_D)$ and $W_P(p^*_P, x^*_P)$ to denote the supply chain’s profits under the firm’s optimal price settings in SP, DDP and PDP, respectively, such that $W_S(p^*_S) = \pi_S(p^*_S) + w_S(p^*_S), W_D(p^*_D, x^*_D) = \pi_D(p^*_D, x^*_D) + w_D(p^*_D, x^*_D)$ and $W_P(p^*_P, x^*_P) = \pi_P(p^*_P, x^*_P) + w_P(p^*_P, x^*_P)$. Then from Theorem 5.2 and Theorem 5.3 the following corollary is a straightforward extension.

**Corollary 5.1.** Under the supplier’s optimal price quotes in the three pricing strategies, the supply chain obtains higher profit in dynamic pricing strategies than in the static pricing strategy, and of the two pricing strategies, the supply chain obtains higher profit in PDP than in DDP. That is, $W_P(p^*_P, x^*_P) > W_D(p^*_D, x^*_D) > W_S(p^*_S)$.

From Theorem 2-3 and Corollary 1, we can conclude that within a decoupled supply chain, there is no conflict between the supplier and its customers regarding their preference of pricing strategies. Thus, by proposing a better pricing strategy to the supplier, we also propose a better pricing strategy for the supply chain. Consequently, for a multi-echelon MTO supply chain, the entire supply chain is optimized when the upstream and the downstream firms all choose the pricing strategy that maximizes their own profits.

5.5 Case Study

In this section, we demonstrate our theorems and corollary through a simulation case study designed based on the practical case of the local window manufacturing firm that motivated this research. For privacy, we use ‘Windows’ to refer the firm. We first design a simulation for a single-echelon supply chain, and then we extend the simulation to a two-echelon supply chain.

We describe Windows’s customers, production system and logistic systems as follows.
**Windows’s customers**  Windows’s customers include local builders, renovators, dealers and some individual customers. Because Windows sells to different types of customers, it receives different types of orders that have different delivery date and price requirements. Builders and some large renovators are Windows’s long-term contracted customers and they do not purchase from Windows’s competitors during the contract period. Long-term contracted customers contribute more than 60% to the Window’s total revenue. They usually make orders with large batches and allow longer leadtimes for delivery. For example, a long-term contracted builder often places orders to Windows as soon as it settles its building plan for the following months. Because of these customers’ bargaining power, the firm cannot dynamically change the long-term contracted orders’ prices that are specified in the contract. The long-term contracted orders are scheduled with the highest priority on their required delivery dates. Spot-market customers include delivery-date-critical, delivery-date-flexible and delivery-date-unspecified customers. Delivery-date-critical and delivery-date-flexible customers include small renovators and some individual customers. Delivery-date-unspecified customers include dealers and some individual customers who do not specify a delivery date but prefer a lower price.

**Windows’s production system**  Windows produces three families of windows made from three different major materials: wood, metal, vinyl. These different product families are processed in different production lines. Figure 5.2 shows an example of a production line, namely V10, for vinyl windows. The production lines are separable based on product families. A production line consists of multiple workstations, each of which handles a particular group of operations. When the firm receives an order, software is employed to estimate the workload (in man-hours) of each operation based on the order’s customization requirements. Then the workload for the order is determined by adding up the estimated workloads for all the required operations. In each typical day, a production line runs 8 hours for a standard day shift, which, if required, can be extended at the overtime labor rate. The firm can control
We define that the working hour of each day is equal to the working hours of the “slowest” workstation. That is, denoting the working hour in workstation \( \text{ktw}_t \) on day \( t \), then the working hours on day \( t \), denoted by \( \text{tw}_t \), can be obtained as \[
\max_{t \in T} \text{ktw}_t \]
Assume that the labour cost is \( c \) for each hour, and then the production cost incurred in day \( t \) can be simply obtained by \( \text{tcw}_t \).

Note the working time in each workstation in each day is constrained by a daily production capacity. Without loss of generality, we assume that in each workstation, the daily production capacity is equal to \( w \).

This setting of the manufacturer can be often found in practice, especially in a flow-shop production environment, where the slowest workstation becomes the bottleneck of a production line. For example, in a door and window company, the shop floor layout for vinyl framed window is displayed in Figure 1. Each square in the figure labeled with ‘V’ and a number is a workstation. Since the workload of a job on each workstation is random, the bottleneck workstation changes in different days.

Figure 1. Shop floor layout for the production of vinyl framed window

We assume that a holding cost \( \text{jh}_j \) will be incurred for job \( j \) in each day. In order to reduce the holding cost for the jobs, the manufacturer usually ships the jobs at the end of the day when they are finished. A holding cost only incurs when the manufacturer received orders from the JIT customers that require due date deliveries.

Figure 5.2: An example production line for vinyl window

the number of shifts dispatched to each worker. That is, if the workload (in man-hours) scheduled for a day is low, the firm is capable of reducing the number of workers working on that day to reduce the labor cost. The labor cost incurred to the firm in a working day can be obtained by multiplying the worker’s hour rate by the actual man-hours in that day. The capacity of a production line is the maximum man-hours available from the line for the production in a standard day shift. For example, in Line V10, the firm’s standard capacity is 360 man-hours in each working day. If more workload is scheduled to V10 in a single day, it will incur overtime. Overtime costs are 1.5 times the standard labor rate.

Windows’s logistic system Because Windows’s products are highly customized, it keeps zero inventory of finished products and therefore does not require a warehouse for its finished products. Instead, the firm moves the finished products directly to the loading dock and to the trucks rented from a 3PL provider. The trucks are then sent out to different destinations every day. Consequently, no finished products remain overnight within Windows. As Windows’s customers are usually located within a 12-hour driving distance, the
transportation time is within the customers’ tolerance for next-day-delivery. Effectively this means that we can suppose that orders are delivered immediately after they are finished on the production line.

5.5.1 The settings for the simulation case study

We design our simulation case study based on Windows’s. In our simulation case study, we only consider one product family and the orders are processed in one production line. The production capacity of the production line is 360 man-hours in a working day. The workload of an order is random due to customization. We set the workload of a long-term contracted order (in man-hours) to be uniformly distributed from $[1, 10]$ and set the workload of a spot-market order to be uniformly distributed from $[0.5, 1]$. The total number of orders arriving at the firm is a random integer uniformly distributed from $[100, 200]$. The probability of a newly arriving order being long-term contracted is set to 40%, and for the spot-market orders, we set the probabilities of a newly arriving order being delivery-date-critical, delivery-date-flexible and delivery-date-unspecified are 20%, 30% and 10%, respectively. Because a long-term contracted order usually allows longer leadtimes, we set the delivery date of a long-term contracted order to be a random integer uniformly distributed from $[11, 20]$, and set an spot-market order’s first-best delivery date to be a random integer uniformly distributed from $[1, 10]$. For each day, we generate different types of customers. Let a newly generated customer be Customer $j$. If Customer $j$ is long-term contracted, then we schedule its order on its first-best delivery date; if Customer $j$ is delivery-date-critical or delivery-date-flexible, then we schedule its order only when it accepts the price quoted by the firm; if Customer $j$ is delivery-date-unspecified, then we add its order to the queue of waiting delivery-date-unspecified orders. The firm processes the waiting delivery-date-unspecified orders once there is available capacity on the current day, that is, the total workload of other types of orders scheduled on the day is less than the firm’s daily production capacity. As the prices quoted to the long-term contracted orders and the delivery-date-unspecified orders are not affected
by different pricing strategies, we only consider the delivery-date-critical orders and delivery-
date-flexible orders when comparing different pricing strategies. The simulation procedure
for each day is summarized as in Figure 5.3 where the content of each node is described in
Table 5.1. We repeat this single-day procedure 100 times so that the Windows’s operation
is simulated for 100 days.

In order to be consistent with Windows’s case, we also consider the material cost for
each order in our simulation. We set the labor rate to be 24 $/man-hour at standard hours
and 36 $/man-hour at overtime hours, and set the material cost of an order to be uniformly
distributed from [5, 10].
Let $c_j^L$ and $c_j^M$ be Order $j$’s labor cost and material cost, respectively. Because it is common that a customer’s valuation on an order correlates with the order’s workload and material, in our simulation, we set $v_j$ to be a ratio of Customer $j$’s absolute valuation as a markup of Order $j$’s total cost, $c_j^L + c_j^M$, and we take $v_j$ as uniformly distributed from $[1.2, 1.6]$. Similar to how we handle $v_j$, in each pricing strategy, the price quotes are expressed as ratios of the total cost of an order. For example, in DDP strategy, if the firm set $p^D$ and $x^D$ to be 1.5 and 0.2, then for an order with the total cost of $100, the customer will be quoted $100 \times 1.5 = $150 if it chooses its first-best delivery date, or quoted $100 \times [1.5-0.2] = $130 if it chooses an alternative delivery date that avoids overtime. We also set the switching cost when a customer chooses an alternative delivery date, 10% of the order’s total cost (without overtime) is incurred by the customer as a switching cost, i.e., $c^A = 0.1$.

5.5.2 The simulation for a single-echelon supply chain

In our simulation, we compare the firm’s profits, the customer and supply chain welfare in DDP and PDP with the ones in SP. For the clarity of showing the curves in 2D charts, in
DDP and PDP, we compute the optimal $x^D$ and $x^P$ that maximize the firm’s profit given each settings of $p^D$ and $p^P$, respectively. The curves of optimal $x^D$ and $x^P$ under different settings of $p^D$ and $p^P$ in DDP and PDP, respectively, are displayed in Figure 5.4. In the following results, we only show the firm’s expected profit with each pricing strategy as a function of a single variable, $p^D$ or $p^P$, because $x^D$ or $x^P$ is automatically set to the optimal setting that is paired with $p^D$ or $p^P$ as in Figure 5.4.

We simulate customers’ choices under different price settings in the three pricing strategies, and then obtain Windows’s realized profit in each pricing strategy by subtracting the total simulated overtime cost from the total simulated revenue. We display the curves of the firm’s realized profits under different settings of $p^S$, $p^D$ and $p^P$ in the three pricing strategies in Figure 5.5. In Figure 5.5 we mark the point where the firm’s profit is maximized for each pricing strategy. We observe that at the marked points, $\pi^P(p^{P*}, x^{P*}) > \pi^D(p^{D*}, x^{D*}) > \pi^S(p^{S*})$. The simulation result shows that firm obtains the highest expected profit with PDP, an obtains the lowest expected profit with SP, which is consistent with Theorem 5.2.

We simulate the customers’ profit for each pricing strategy by averaging the net gains of all the simulated customers. The curves of the aggregate customer profit under different
settings of $p^S$, $p^D$ and $p^P$ in the three pricing strategies are displayed in Figure 5.6. The simulation result shows that customers obtain the highest profit with PDP, an obtain the lowest profit with SP, i.e., \( w^P(p^P^*, x^P^*) > w^D(p^D^*, x^D^*) > w^S(p^S^*) \), which is consistent with Theorem 5.3.

Finally, by adding up the firm’s and customers’ profit, we can obtain the curves of the supply chain’s profit under different settings of $p^S$, $p^D$ and $p^P$ in the three pricing strategies as shown in Figure 5.7. We can observe that at the points where the firm’s profits are maximized in the three pricing strategies, the supply chain obtains the highest profit with PDP, an obtains the lowest profit with SP, i.e., \( W^P(p^P^*, x^P^*) > W^D(p^D^*, x^D^*) > W^S(p^S^*) \), which is consistent with Corollary 5.1

5.5.3 The Simulation of a two-echelon Supply Chain

In order to show the superiority of dynamic pricing strategies in a multi-echelon supply chain, we simulate a two-echelon supply chain which includes two manufacturers, M1 and M2, and M1 is a supplier of M2 as shown in Figure 5.8. That is, M1 is upstream of M2. When an end customer arrives at M2 and enquires about the price for a final product, M2 quotes the customer the price and the delivery date. If the customer accepts the quote and
Figure 5.6: The curves of customers’ profit under different settings of \( p^S, p^D \) and \( p^P \) in SP, DDP and PDP, respectively

Figure 5.7: The curves of the supply chain’s profit under different settings of \( p^S, p^D \) and \( p^P \) in SP, DDP and PDP, respectively
places the order, then M2 goes to M1 and enquires about the price for the necessary part. Again, when M2 arrives to M1, M1 quotes to M2 the price and the delivery date. M2 can accept the quote or choose another supplier without placing an order to M1. After the end customer places the order, M2 places an order for the required part to M1. The type of the order from M2 to M1 is the same as the corresponding one from the end customer to M2. That is, for example, if an end customer places a delivery-date-critical order to M2, then M2 also places a delivery-date-critical order to M1.

Without loss of generality, we assume that both M1 and M2 have the same structure as Windows and the parameters for M1 and M2 are set the same as in Section 5.5.1. An order from a customer requires the same labor cost and material cost in M1 and M2. For simplicity, we ignore the varying scale of pricing in the upstream and the downstream firms of the supply chain. That is, for both M2 and end customers, when they purchase from M1 and M2, respectively, we assume that their valuations on an order, $v_j$, both follow the same distribution and the switching cost for choosing an alternative delivery date, $c^A$, both equal to 0.1 as in Section 5.5.1. M2’s switching cost for choosing an alternative delivery date accounts for the renegotiations with end customers due to the change in delivery dates.

We suppose that M1 and M2 both sets price quote to maximize their expected profits, respectively. We first run the simulation for a supply chain which only contains M2 and end customers. We repeat the simulation with different settings of $p^S$, $p^D$, $x^D$, $p^P$ and $x^P$ to find the optimal ones for M2 which maximize its profits in SP, DDP and PDP, respectively. Then based on the orders placed in M2, we run the simulation for a supply chain which only contains M1 and M2. Again, we repeat the simulation with different settings of $p^S$, $p^D$, $x^D$, $p^P$ and $x^P$.
$x^D$, $p^P$ and $x^P$ to find the optimal ones for M1 which maximize its profits in SP, DDP and PDP, respectively. In Figure 5.9, we show the production schedules of M1 and M2 under the optimal price settings in different pricing strategies. In Figure 5.9, the length of each bar indicates the workload of orders scheduled on that day, and we use different shading to show the workload of different types of orders. The black bars are pre-scheduled orders which include the long-term contracted orders and the delivery-date-critical and delivery-date-flexible orders placed before the current day. We randomly generate the pre-scheduled orders from $U(340, 370)$ for this simulation. We can observe that under SP, because the delivery-date-flexible and delivery-date-critical orders are not differentiated, the delivery-date-flexible orders are scheduled to Day 3, 6 and 8 in M1 and to Day 1, 10 in M2 although overtime is incurred. Under DDP and PDP, the delivery-date-flexible orders are induced to the dates without overtime for both M1 and M2. We can observe that the amount of orders received under DDP is less than under PDP although in DDP the price settings are lower. This is because in DDP, the customers are not offered a discount if their first-best delivery dates incur no overtime, and consequently the profits for this group of customers are lowered. It can be noticed that for each type of orders scheduled for each day, the total workload scheduled in M1 is a little lower than in M2. This is because if M2 does not accept M1’s quote, it turns to another supplier.

In Table 5.2, we display the simulation result which support our analytical results. Here we use the sum of M2’s profit from end customers and its net gain from M1 to represent M2’s total gain from the supply chain. Then the social welfare equals to the sum of M1’s profit, M2’s profit, M2’s net gain from M1 and end customer’s net gain from M2. We can observe that M1, M2, M2’s customers and the entire supply chain benefit most with PDP, and benefit least with SP, which is consistent with our theorems and corollary.
Figure 5.9: The production schedules of M1 and M2 under different pricing strategies.
Strategy | M1’s Price quote | M2’s Price quote | M1’s profit | M2’s profit | M2’s net gain from M1 | Customers’ net gain from M2 | Social welfare
--- | --- | --- | --- | --- | --- | --- | ---
SP | $p^S = 1.38$ | $p^S = 1.38$ | 622.71 | 800.69 | 270.04 | 270.04 | 1963.51
DDP | $p^D = 1.37$ | $p^D = 1.37$ | 632.31 | 832.31 | 307.92 | 306.72 | 2079.28
PDP | $p^P = 1.29$ | $p^P = 1.28$ | 668.82 | 870.69 | 318.12 | 485.75 | 2343.38

Table 5.2: The result of the simulation for the two-echelon supply chain

5.5.4 Discussion

Through mathematical analysis and simulation results, we have proved that within a MTO supply chain, the dynamic pricing strategies are superior to the static pricing strategy, and between DDP and PDP, both upstream and downstream firms (or end customers) are better off in PDP. The results are robust regardless of the structure of the supply chain. However, although PDP is superior to DDP, we can also find reasons to employ DDP. For example, for a firm which is current employing SP, we have to consider the customer’s feedback when upgrading the pricing strategy. From the interviews with the manager of the marketing department in Windows, we are informed that if some customers are quoted the prices higher than the current ones, their satisfaction will be significantly reduced, which will jeopardize the firm’s reputation and consequently reduce the firm’s long-term profit. According to Theorem 5.1, if Windows upgrades to PDP, some customers will be quoted with $p^P + x^P$ which are more than what they are quoted in SP. Thus, rather than PDP, Windows has to employ DDP, in which all customers are quoted the prices lower than the ones in SP. When implementing DDP strategy, in order to find $p^{D^*}$, the firm needs to gradually reduce the setting of $p^D$ from the optimal one used in SP until the periodical profit reaches to the maximum value.

For the firms which do not have problem in increasing prices, e.g., some newly founded
firms or monopoly firms, PDP strategy should be the best choice for their pricing strategy.

5.6 Chapter Conclusions

In this chapter, we study the dynamic pricing strategies as a solution for the decentralized coordination of the supply chain in which it is difficult for the upstream and downstream firms to cooperate based on contracts with information and revenue sharing. We present models for a static pricing strategy and two dynamic pricing strategies, i.e., SP, DDP and PDP. Through the analysis of mathematical models, we have shown that within a decoupled supply chain, both the firm and the customers are better off in the dynamic pricing strategies, and hence we draw the conclusion that the supply chain is better off in dynamic pricing strategies. Between the two dynamic pricing strategies presented in the paper, we showed that the PDP is superior to DDP. The results obtained from the analytical model are tested in simulation experiments for both single-echelon and multi-echelon supply chains.

Based on our analytical and simulation results, we also suggested that although PDP is superior to DDP, in some cases, DDP should be chosen when the firm tries to upgrade to a dynamic pricing strategy from a static pricing strategy. Through the example of Windows, we show why DDP should be employed rather than PDP, and how it should upgrade from SP to DDP.

The main contribution of this work is that we proved how the decentralized coordination of a MTO supply chain is realized through dynamic pricing strategy through both analytical model and simulation experiments. More general cases can be found in practice where the finished products can be stocked at the firm’s cost and the production schedule can be optimized by scheduling similar orders to be processed in the same day. For those more general problems, it is usually difficult to analyze the firm’s optimal pricing strategy through analytical methods due to the complexity. Hence, in the future research, the simulation-based methods will be designed to study the more general problems.
Chapter 6

Scheduling with Compressible and Stochastic Release Dates

6.1 Background

In this chapter, we study the scheduling problem that when a manufacturer orders components from its suppliers, the manufacturer faces multiple leadtime options under different prices, and for each leadtime option, instead of a deterministic leadtime, the suppliers of the components only promise the delivery within a time interval. The following examples illustrate this problem setting.

Example 1 A manufacturer orders components from one supplier, who offers orders with different priority options at different prices, e.g., regular order, rush order and super rush order.

Example 2 A manufacturer has multiple substitutable component suppliers in different locations, e.g., local, domestic and offshore. The suppliers offer different prices, while the promised leadtimes are different because of the differences in distance.

Example 3 When a manufacturer makes orders from its supplier(s), there are multiple transportation modes, e.g. by air, by rail and by road, each of which incurs a different transportation cost.

In the three examples above, it is usually hard to guarantee an exact date when the order can be delivered, and instead the manufacturer only expects a time interval. Because the delivery of the component constrains the release of each job within the manufacturer, we
hereafter use the term “compressible and stochastic release dates” to refer to the optional leadtime intervals offered by the component supplier.

In this chapter, we study the manufacturer which can expedite the production by expediting, i.e., overtime, under an overtime cost when the manufacturer’s regular production capacity is not enough. It has been proved that in the make-to-order industry, expediting by overtime production is a common practice to guarantee the due-date delivery (Miller and Fink 1990).

We describe the scheduling problem studied in this chapter as a single-machine scheduling problem with compressible and stochastic release dates, due-date guarantee and costly expediting. The target of the manufacturer is to minimize the total cost which includes the total cost for compressing the release dates and the total cost for expediting.

Our sections are organized as follows. In Section 6.2, we describe the problem with notations. As a foundation for developing the heuristic for the scheduling problem with compressible and stochastic release dates, in Section 6.3, we first study the problem with the setting that the release dates are compressible and deterministic. Then in Section 6.4, we solve the stochastic problem by converting the stochastic problem to the deterministic problem. Finally, a summary of conclusion is presented in Section ??.

6.2 Problem Description and Notations

In this section, we describe our problem settings with notation. For clarity, we summarize the notation used in this chapter in Table 6.1.

Without loss of generality, we suppose that the manufacturer’s production periods are managed in days, and Day $t : t \in \mathbb{N}$ is the $t$th day after the current day. We study a manufacturer which has $J$ jobs to be scheduled, each of which is indexed by $j : j \in \{1, 2, \ldots, J\}$.

We treat the manufacturer as a single-machine production system. A job’s workload is
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t )</td>
<td>Day ( t ) is the ( t )th day after the current day.</td>
</tr>
<tr>
<td>( J )</td>
<td>The total number of the jobs to be scheduled.</td>
</tr>
<tr>
<td>( j )</td>
<td>A job’s index such that ( j \in {1, 2, \ldots, J} ).</td>
</tr>
<tr>
<td>( p_j )</td>
<td>The processing time of Job ( j ).</td>
</tr>
<tr>
<td>( H )</td>
<td>The manufacturer’s daily regular production capacity, i.e., the maximum number of regular hours in each working day.</td>
</tr>
<tr>
<td>( v_j )</td>
<td>The overtime hours used for processing Job ( j ).</td>
</tr>
<tr>
<td>( z )</td>
<td>The cost incurred for a unit overtime hour.</td>
</tr>
<tr>
<td>( r_j )</td>
<td>The release date of Job ( j ).</td>
</tr>
<tr>
<td>( \xi_j )</td>
<td>The compressing fee for Job ( j ), i.e., the cost incurred for compressing the release date of a unit workload (processing time) of Job ( j ).</td>
</tr>
<tr>
<td>( \xi_j(r_j) )</td>
<td>Job ( j )’s compressing fee function when the release date of Job ( j ) is deterministic.</td>
</tr>
<tr>
<td>( r_j^{LB} )</td>
<td>The earliest possible release date of Job ( j ).</td>
</tr>
<tr>
<td>( f_j^R(r</td>
<td>\xi_j) )</td>
</tr>
<tr>
<td>( F_j^R(r</td>
<td>\xi_j) )</td>
</tr>
<tr>
<td>( d_j )</td>
<td>The guaranteed due date (deadline) of Job ( j ).</td>
</tr>
<tr>
<td>( s_j )</td>
<td>Job ( j )’s start time.</td>
</tr>
<tr>
<td>( c_j )</td>
<td>Job ( j )’s completion time.</td>
</tr>
<tr>
<td>( S )</td>
<td>The schedule of all the jobs. In the deterministic case ( S = (\vec{r}, \vec{s}, \vec{v}) ) while in the stochastic case ( S = (\vec{\xi}, \vec{s}, \vec{v}) ). ( \vec{r}, \vec{\xi}, \vec{s} ) and ( \vec{v} ) indicate the release dates, the compressing fees, the staring times and the overtime workloads of all the jobs, respectively.</td>
</tr>
<tr>
<td>( r_j^*(\xi_j) )</td>
<td>The converted release date of Job ( j ) under converting fee ( \xi_j ) when the release date is stochastic. ( * ) can be substituted with ( O, P, N ) or ( L ) to denote the converted release date under different converting policy.</td>
</tr>
<tr>
<td>( L_j(S) )</td>
<td>Job ( j )’s LRIF of when the job Schedule is ( S ).</td>
</tr>
<tr>
<td>( L_j'(S) )</td>
<td>The surrogate measure of Job ( j )’s LRIF.</td>
</tr>
</tbody>
</table>
described by the processing time required by the job. We use $p_j$ to denote Job $j$'s processing time which indicates how many hours are required for the manufacturer to process Job $j$. A job can be processed under the manufacturer’s regular hours or overtime hours in each working day. We assume that if a job is processed in the manufacturer’s regular hours, no cost is incurred; otherwise, if a job is processed in the manufacturer’s overtime hours, an overtime cost $z$ will be incurred for each overtime hour. We assume that within the manufacturer, the maximum number of regular hours in each working day, namely the manufacturer’s daily regular production capacity, is fixed at a positive integer $H$, and the number of overtime working hours in each day is not constrained. This setting is common in practice when a manufacturer employs a number of fulltime workers with fixed base monthly payment for fixed base working hours on each working day, but pays overtime payment if the workers work in overtime hours. We do not assign an upper limit to the overtime hours in each day because it is common that the manufacturer tries to reduce overtime production to avoid overtime cost and so the overtime production capacity usually does not bind. Similar settings can be found in Çelik and Maglaras (2008) where the authors assume the expediting (which captures overtime) capacity to be unlimited. We assume that the jobs are processed with the same rate no matter in regular hours or in overtime hours. That is, the processing time of any job is the same no matter in regular hours or in overtime hours. A job can be processed partially in regular hours and partially in overtime hours, and the overtime hours used for processing Job $j$, namely Job $j$’s overtime workload, is denoted by $v_j$ such that $v_j$ is a non-negative real and $v_j \leq p_j$.

The processing of Job $j$ is constrained by its release date, denoted by $r_j$, and the value of $r_j$ depends on the compressing fee the manufacturer pays to its supplier. We assume that the total compressing cost of a job is proportional to its processing time. That is, supposing that the manufacturer pays compressing fee, denoted by $\xi_j$, for a unit processing time of Job $j$, then the total compressing cost of Job $j$ can be obtained at $\xi_j p_j$. We assume that at the
beginning of Day 0, the manufacturer decides the $r_j$ for all the jobs to be scheduled within the scheduling horizon, and a component is only available to be used after it arrives at the manufacturer. We assume that the components are only delivered at the end of each day, and hence there is $r_j \in \mathbb{Z}^+$, where $\mathbb{Z}^+$ is the set of nonnegative integers. We study both cases that when $\xi_j$ is given, $r_j$ is deterministic and stochastic, respectively.

When the release dates of all the jobs are deterministic, the manufacturer can expect a deterministic $r_j$ for each release-date compressing fee $\xi_j$, and thus we use $\xi_j(r_j)$ to denote the compressing fee function for the release date $r_j$ of Job $j$. Because in practice, it is realistic that a manufacturer increases its procurement cost to reduce a job’s release date, i.e., to expedite the purchased components’ delivery, we assume that $\xi_j(r_j)$ is non-increasing in $r_j$. Without loss of generality, we assume that $\xi_j(r_j) \to 0$ when $r_j \to \infty$ to ensure a nonnegative compressing fee. Because it is often the case that a job’s release date cannot be reduced unlimitedly due to the constraint of the component supplier’s production capability and shipment leadtime, we denote the lower bound of $r_j$ by $r_j^{LB}$ which is exogenous and indicates the earliest possible release date of Job $j$.

When the release dates of all the jobs are stochastic, we use $f^R_j(r|\xi_j)$ and $F^R_j(r|\xi_j)$ to denote the probability mass function (pmf) and the cumulative distribution function (cdf) of Job $j$’s release date under compressing fee $\xi_j$, respectively.

We assume that the jobs are processed nonpreemptively. That is, once the processing of a job starts, it must be completed without inserting other jobs. However, if a job is only partially completed by the end of a day, it can be continued in the next day.

In this work, we ignore the inventory cost for the components and the finished products. The manufacturer provides a due-date-guaranteed service, and the due date of Job $j$ is denoted by $d_j$. Note that because the due dates of all the jobs are guaranteed, the “due dates” referred in this chapter is equivalent as the “deadlines” referred in the scheduling literature. We assume that there is always a release-date setting in which all the jobs can be
processed no later than their due dates. The manufacturer’s objective function for scheduling is

\[
\text{Minimize } \sum_{j=1}^{J} p_j \xi_j + z \sum_{j=1}^{J} v_j,
\]

which is to minimize the sum of total release-date compressing cost and total overtime cost for all the jobs.

### 6.3 Scheduling with Compressible and Deterministic Release Dates

We name the scheduling problem when all the jobs have compressible and deterministic release dates as the CDR problem.

**Lemma 6.1.** The CDR problem is NP-hard.

**Proof:** Following the convention in the literature, to prove that the CDR problem is NP-hard, we only need to show that there is a known NP-complete problem that is reducible to the CDR problem. That is, if there is an NP-complete problem that can be converted to the CDR problem, then the CDR problem can be used to solve that NP-complete problem. Hence, we can conclude that the CDR problem is at least as difficult as that NP-complete problem, which means the CDR problem is NP-hard. Here, we prove that the Partition problem, a typical NP-complete problem [Karp (1972)], is reducible to the CDR problem. Similar proof procedure can also be found in [Stecke and Zhao (2007)].

The Partition problem defined in [Karp (1972)] can be summarized as: given a set of integers \(U\), does there exist a subset \(S \subset U\), such that \(\sum_{e \in S} e = \sum_{e \in U - S} e\)? To prove that the Partition problem is reducible to the CDR problem, we only need to prove that any Partition problem can be converted to a CDR problem.

Suppose that in a special CDR problem, namely an SCDR problem, there are \(J\) jobs to
be scheduled within 3 periods and $\sum_{j=1}^{J} p_j = 3H$. Let Job $J$ have $p_J = H$, $d_J = 2$ and

$$\xi_J(r_J) = \begin{cases} 0 & \text{if } r_J \geq 1, \\ \infty & \text{otherwise.} \end{cases}$$

For any Job $j$ where $j \neq J$, let $d_j = 3$ and $\xi_j(r_j) = 0$ for any $r_j$. Supposing that in the optimal schedule there is $z \sum_{j=1}^{J} v_j = 0$, then it is obvious that in the optimal schedule, Job $J$ starts at $r_J = 1$ and ends at $d_J = 2$. Hence, the problem is equivalent as: given a set of integers $U = \{p_1, p_2, \ldots, p_{J-1}\}$, find a subset $S \subset U$, such that $\sum_{e \in S} e = \sum_{e \in U-S} e = H$, which can represent any Partition problem. Because the SCDR problem is a special form of the CDR problem, then if the CDR problem is solvable, the SCDR problem is solvable, and consequently the Partition problem is solvable. Hence, we can conclude that the CDR problem is at least as difficult as the Partition problem, which means the Partition problem is reducible to the CDR problem, and so the CDR problem is NP-hard. Q.E.D.

Because the CDR problem is NP-hard, a heuristic algorithm is developed to obtain the near optimal schedule which minimizes the total cost. As the foundation for developing the heuristic algorithm for the CDR problem, we first study the scheduling when the job sequence is given. We name the CDR problem when the job sequence is given as CDR-S.

### 6.3.1 Heuristic for CDR-S

Without loss of generality, in CDR-S we say that Job $j$ is at the $j$th position in the job sequence. We use $c_j$ to denote the completion time of Job $j$ where $c_j$ is a nonnegative real number. For example, $c_j = 1.5$ means that Job $j$ is finished in the middle of Day 2. We suppose that there are $J$ jobs to be scheduled between Day $t_1$ and Day $t_2$. Then CDR-S can be mathematically formed as:

$$\text{CDR-S:} \quad \min \sum_{j=1}^{J} \xi_j(r_j)p_j + z \sum_{j=1}^{J} v_j, \quad (6.1)$$
subject to: \[ c_0 = t_1, \] \[ c_J \leq t_2, \] \[ c_{j-1} + \frac{p_j - v_j}{H} \leq c_j, \quad j = 1, 2, \ldots, J, \] \[ r_j + \frac{p_j - v_j}{H} \leq c_j, \quad j = 1, 2, \ldots, J, \] \[ c_j \leq d_j, \quad j = 1, 2, \ldots, J, \] \[ 0 \leq v_j \leq p_j, \quad j = 1, 2, \ldots, J, \] \[ r_j \in \mathbb{Z}^+ \text{ and } r_j \geq r_j^{LB}, \quad j = 1, 2, \ldots, J. \] (6.2) (6.3) (6.4) (6.5) (6.6) (6.7) (6.8)

In the CDR-S model, (6.1) is the firm’s objective function which minimizes the total cost which includes the total compressing cost and the total overtime cost. (6.4) constrains that Job \( j \) has to start after when Job \( j-1 \) job finishes. (6.5) constrains that Job \( j \) has to start after its release date. (6.8) constrains that the release dates are integers and for each Job \( j \), the release date cannot be earlier than a lower bound \( r_j^{LB} \).

Depending on the form of \( \xi_j(r_j) \), the CDR-S problem can be a mixed-integer-linear programming (MILP) or a mixed-integer-nonlinear programming (MINLP) which is difficult to be solved. Therefore, we develop the following heuristic for solving CDR-S to obtain the near optimal schedule within a substantially reduced computation time. We name the heuristic for solving CDR-S problem as HS. In HS, the function \( \xi_j(r_j) \) is modified by setting \( \xi_j(r_j) = +\infty \) when \( r_j < r_j^{LB} \). With this setting, we can assure that there is always a (mathematically) feasible schedule which incurs zero overtime although it might incur infinitely big total cost. For clarity, we use \( s_j : s_j \in \mathbb{R}^+ \) to denote Job \( j \)’s start time. When the schedule of a series of jobs is decided, it indicates that the release date, the overtime workload and the start time of each job are decided. We denote a job schedule by \( S \) such that \( S = (\vec{r}, \vec{s}, \vec{v}) \), where the vectors \( \vec{r}, \vec{s} \) and \( \vec{v} \) are the release dates, the start times and the overtime workloads for all the jobs, respectively. Suppose that the scheduling horizon is from Day \( t_1 \) to Day \( t_2 \). We describe HS by the following steps.

HS:
Step 1. Set $\hat{j} \leftarrow J$, and generate the initial schedule $S$ by scheduling each job to start at its latest start time without overtime, i.e., $s_j \leftarrow \min \{ t_2, d_j \} - p_j/H$, $c_j \leftarrow \min \{ s_{j+1}, d_j \}$ for each $j \in \{1, 2, \ldots, J-1\}$, $s_j \leftarrow c_j - p_j/H$ for each $j \in \{1, 2, \ldots, J-1\}$, $r_j \leftarrow \lfloor s_j \rfloor$ for each $j \in \{1, 2, \ldots, J\}$ and $v_j \leftarrow 0$ for each $j \in \{1, 2, \ldots, J\}$.

Step 2. Schedule Job $\hat{j}$ and settle $r_{\hat{j}}$ through Step 2.1-2.5:

Step 2.1. If $r_{\hat{j}}^{LB} > s_{\hat{j}}$, set $r_{\hat{j}} \leftarrow r_{\hat{j}}^{LB}$.

Step 2.2. Set $\hat{j}' \leftarrow \hat{j}$.

Step 2.3. If $s_{\hat{j}'} < \max \{ r_{\hat{j}'}, c_{\hat{j}'-1} \}$, then set $s_{\hat{j}'} \leftarrow \max \{ r_{\hat{j}'}, c_{\hat{j}'-1} \}$ and $r_{\hat{j}'} \leftarrow \lfloor s_{\hat{j}'} \rfloor$.

Step 2.4. If $s_{\hat{j}'} + p_{\hat{j}'}/H > d_{\hat{j}'}$, then set $c_{\hat{j}'} \leftarrow d_{\hat{j}'}$ and $v_{\hat{j}'} \leftarrow s_{\hat{j}'} + p_{\hat{j}'}/H - d_{\hat{j}'}$; otherwise, set $c_{\hat{j}'} \leftarrow s_{\hat{j}'} + p_{\hat{j}'}/H$.

Step 2.5. If $v_{\hat{j}'}$ increases in Step 2.4, try if the cost incurred by job $\hat{j}'$ can be reduced by setting $r_{\hat{j}'}$ to $r_{\hat{j}'} + 1$. If yes, then sequentially set $r_{\hat{j}} \leftarrow r_{\hat{j}} + 1$, $s_{\hat{j}} \leftarrow r_{\hat{j}}$ and $v_{\hat{j}} \leftarrow s_{\hat{j}} + p_{\hat{j}}/H - d_{\hat{j}}$.

Step 2.6. If $c_{\hat{j}'}$ changes in Step 2.4, set $\hat{j}' \leftarrow \hat{j}' + 1$ and go to 2.3.

Step 2.7. Compute the total cost under the current job schedule. If the total cost is increased comparing to the one under $S$, then reschedule the jobs to $S$, and go to Step 3; otherwise, record the current schedule as $S$, set $r_{\hat{j}} \leftarrow r_{\hat{j}} + 1$, and go back to Step 2.2.

Step 3. If $\hat{j} = 1$, then STOP; otherwise set $\hat{j} \leftarrow \hat{j} - 1$ and go back to Step 2.

In HS, we sequentially adjust the release date, the starting date and the overtime workload of each job from Job $J$ to Job 0. When the start time of Job $\hat{j}$ is changed, the jobs after Job $\hat{j}$ are also adjusted accordingly in Step 2. We use the following example (Figure 6.1) to show the procedure of HS. In the example, we use a five-tuple, $(r_j, s_j, v_j; p_j, d_j)$, to describe a scheduled job as in Figure 6.1(a). Supposing that there is a job sequence which includes three jobs as shown in Figure 6.1(b). For all the three jobs, we assume that the compressing cost function are the same as $\xi_j(r_j)p_j = (r_j - 3)^2$. The unit overtime cost is set to $z = 10$ and the daily regular production capacity is set to $H = 1$. The three jobs are meant to be scheduled within three days. Step 2 is illustrated in Figure 6.1(d)-(f). In Figure 6.1(d), we
have $r_3 = 1$ and $s_3 = 1.6$. Any increase of $r_3$ will result in an increase of total cost. In Figure 6.1(e), when $r_2$ is increased from 0 to 1, $s_2 = 1$ and Job 3 is moved towards the righthand side by 0.1, which results in an increase of 0.1 overtime production. In 6.1(f), when $r_1$ is increased from 0 to 1, the total cost is reduced, then Job 1 is scheduled to $s_1 = 1$, and Job 2 and Job 3 are rescheduled accordingly. Because the due date of Job 2 is $d_2 = 2$, an overtime is incurred at the end of the second day. $r_3$ is increased from 1 to 2. The schedule in Figure 6.1(f) is the final schedule when the job sequence in Figure 6.1(b) is given.

To test the efficiency of HS, we design a numerical experiment to compare HS with the
global solver of Lingo (Version 13.0, 64-bit). The global solver (GS) of Lingo implements a branch and cut algorithm that utilizes linear programming for bounding, and it ensures global optimal solutions for convex and nonconvex MINLPs \cite{LinSchrage2009,BussieckVigerske2010}. We use the GS of Lingo to solve the optimal schedule, and then compare the optimal solution with the one obtained with HS to evaluate the HS’s performance. In order to test the efficiency of HS for large scale problem in Section\ref{sec:6.3.2} we also use Lingo to solve the CDR-S model by treating all the integer decision variables as real variables. Since after the relaxation, the CDR-S model becomes a non-linear-programming (NLP) model, it can be solved quickly by Lingo. Supposing that by solving the relaxed model of CDR-S, the optimal solution of $r_j$ is $r'_j$, we convert $r'_j$ to an integer by setting $r'_j := \lfloor r'_j \rfloor$, and then compute the total cost. For the purpose of clarity, we name this relaxation-based method for solving CDR-S as RS. We also use RS as a benchmark to evaluate the HS’s performance.

In the numerical experiment, we set the compressing fee function, $\xi_j(r_j)$, to be in the form of

$$\xi_j(r_j) = \frac{\alpha_{1j}}{(r_j + 1)^{\alpha_{2j}}}, \quad (6.9)$$

where $\alpha_{1j}$ and $\alpha_{2j}$ are Job $j$’s random parameters both uniformly distributed from interval $(0, 3]$. We use this setting to make $\xi_j(r_j)$ consistent with our assumptions in Section\ref{sec:6.2} i.e., $\xi_j(r_j)$ is non-increasing in $r_j$ and $\xi_j(r_j) \to 0$ when $r_j \to +\infty$. $\alpha_{1j}$ and $\alpha_{2j}$ determine the shape of $\xi_j(r_j)$, and we use the varying realizations of $\alpha_{1j}$ and $\alpha_{2j}$ to represent the heterogeneity of the jobs’ compressing fee functions. Note that here the form of $\xi_j(r_j)$ is only designed for our simulation. In practice, the form of $\xi_j(r_j)$ can be complex, e.g., a piecewise function.

Without loss of generality, we set $r_j$’s lower bound, $r_j^{LB}$, to 0 for each $j \in \{1, 2, \ldots, J\}$. We schedule the randomly generated jobs for 10 days. The processing times and the due dates of the jobs are random integers uniformly distributed from intervals $[1, 20]$ and $[1, 10]$, respectively. The results of the numerical experiment under different problem scales (i.e., number of jobs) are displayed in Table\ref{table:6.2}. In the numerical experiment for each problem...
scale, we set the daily capacity to be 20% more than the average workload (total processing time) on each day. We run the experiment for each problem scale for 50 times with different randomly generated jobs.

In Table 6.2, we show the average time (CPU time) that GS, RS and HS take to compute the schedule. The average percentage gaps indicate the differences between the optimal total costs and the ones obtained from RS and HS. It can be observed that under different problem scales, HS can obtain the near optimal solution with the substantially reduced CPU time.

### 6.3.2 Solving CDR

Because the near optimal total cost for any job sequence can be obtained by solving the CDR-S problem, then in order to solve the CDR problem, a heuristic algorithm, namely H-CDR, is developed based on pairwise-interchange algorithm to settle the job sequence. The pairwise-interchange algorithm is a well applied heuristic which is used to solve job sequencing problems ([Baker and Baker 1974](#), [French 1982](#), [Federico 1995](#), etc.). For clarity, we define that Job $j$ is also the $j$th job in the sequence. When we say “swapping Job $j_1$ and Job $j_2$”, we mean changing the position of Job $j_1$ and Job $j_2$, and after swapping, Job $j_1$ and Job $j_2$, respectively, refer to the Job $j_2$ and the Job $j_1$ before the swapping. We use $\langle j_1, j_2 \rangle$ to denote the job sequence of all the jobs from Job $j_1$ to Job $j_2$. H-CDR can be described by the following steps:

**H-CDR:**

1. Generate the initial sequence for all the $J$ jobs, and solve the initial schedule (by

---

### Table 6.2: Comparing different methods for solving CDR-S

<table>
<thead>
<tr>
<th>$J$</th>
<th>$c$</th>
<th>Optimal</th>
<th>RS</th>
<th>HS</th>
<th>Average gap of total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>12</td>
<td>1.00</td>
<td>0.79</td>
<td></td>
<td>$3.91 \times 10^{-4}$</td>
</tr>
<tr>
<td>20</td>
<td>24</td>
<td>2.06</td>
<td>0.90</td>
<td></td>
<td>$1.17 \times 10^{-3}$</td>
</tr>
<tr>
<td>30</td>
<td>36</td>
<td>8.26</td>
<td>1.00</td>
<td></td>
<td>$4.98 \times 10^{-3}$</td>
</tr>
<tr>
<td>40</td>
<td>48</td>
<td>35.38</td>
<td>1.32</td>
<td></td>
<td>$1.46 \times 10^{-2}$</td>
</tr>
<tr>
<td>50</td>
<td>60</td>
<td>192.67</td>
<td>1.61</td>
<td></td>
<td>$8.98 \times 10^{-2}$</td>
</tr>
</tbody>
</table>
HS or RS). Record the initial schedule as $S$.

Step 2. Set $j_1 \leftarrow J - 1$.

Step 3. Set $j_2 \leftarrow 1$.

Step 4. Swap Job $j_2$ and the first job, denoted by Job $j'_2$, which starts after the day when Job $j_2$ starts, and then run HS or RS to solve the schedule for $\langle j_2, \min \{ j_2 + \delta, J \} \rangle$ where $\delta$ is a preset integer constant. If after the swapping, the total cost is reduced, then update $S$ to the current schedule; otherwise, reschedule the jobs as $S$.

Step 5. Set $j_2 \leftarrow j_2 + 1$. If $j_2 > j_1$, then go to Step 6; otherwise, go back to Step 4.

Step 6. If no swapping happens for the current setting of $j_1$, then STOP; otherwise, set $j_1 \leftarrow j_1 - 1$ and go to Step 7.

Step 7. If $j_1 > 0$, then go back to Step 3; otherwise, STOP.

In H-CDR, we use the earliest-due-date-first rule to generate the initial sequence, and use the longest-processing-time-first (LPT) rule for a tie. We highlighted the LPT for the following reason. In Section 6.2, we assume that the compressing cost for a job is directly proportional to its processing time and a job can only be processed after its release date. Thus, without concerning the impact of the heterogeneity of compressing fee functions (i.e., supposing all jobs have the same compressing fee functions), employing LPT rule can reduce the total compressing cost without increasing overtime cost in some cases. For example, suppose that there are two jobs, Job 1 and Job 2, with the same deadline $d_1 = d_2 = 2$ and the manufacturer needs more than a day to process Job 1. As shown in Figure 6.2, it can be observed that when LPT rule is employed, the release dates for small orders can be deferred so that the total compressing cost is reduced.

We use a positive integer, $\delta$, to control the computation time when the number of jobs is large. Since swapping two jobs which both start and end in the same day will not cause any change of cost, we swap the jobs starting in different days in Step 4. Because the proposed H-CDR is a variant of bubble sorting algorithm, we can conclude that H-CDR has
6. What are exactly the random parameters in Eq. (9)? Please explain.

The random parameters in Eq. (9) do not have practical meanings. We design the form of the compressing fee function,

\[ \text{ Antar } (k, s) \]

as in Eq. (9) so that \( \text{ Antar } (k, s) \) is consistent with our assumptions in Section 3, i.e., \( \text{ Antar } (k, s) \) is non-increasing in \( s \) and \( \text{ Antar } (k, s) \rightarrow 0 \) when \( s \rightarrow \infty \). The random parameters, \( \alpha_k \) and \( \alpha_s \), determine the shape of \( \text{ Antar } (k, s) \), and we use the varying realizations of \( \alpha_k \) and \( \alpha_s \) to represent the heterogeneity of the jobs’ compressing fee function. The form of \( \text{ Antar } (k, s) \) is only designed for our simulation. In practice, the form of \( \text{ Antar } (k, s) \) can be complex, e.g., a piecewise function.

7. Page 11 – Please show me the basis of the ‘biggest-workload-first-rule’ in H-CDR.

Since we revised the term ‘workload’ to ‘processing time’, then the ‘biggest-workload-first-rule’ is revised to ‘longest-processing-time (LPT) rule’ in the new version.

In a general H-CDR, we do not specify a rule in Step 1 to generate an initial sequence so that the manufacturer can define its own rule based one practical situation. However, we highlighted the LPT for the following reason:

In Section 3, we assume that the compressing cost for a job is directly proportional to its processing time and a job can only be processed after its release date. Thus, without concerning the impact of the heterogeneity of compressing fee functions (i.e., supposing all jobs have the same compressing fee functions), employing LPT rule can reduce the total compressing cost without increasing overtime cost in some cases. For example, suppose that there are two jobs, Job 1 and Job 2, with the same deadline \( t_1 = t_2 = 2 \) and the manufacturer needs more than a day to process Job 1. As shown in Figure R1, it can be observed that when LPT rule is employed, the release time for small orders can be deferred so that the total compressing cost is reduced.

Figure R1. An example for illustrating the superiority of LPT rule when multiple jobs have the same deadline

8. Eq. (13) – What’s the physical meaning of \( L_{\alpha} (S) \)? Idleness or unused capacity?

We coined the concept of LRIF, which measures the impact incurred by a job when it is released later than its scheduled starting time. \( L_{\alpha} (S) \) is a surrogate measure of LRIF. \( L_{\alpha} (S) \) does not have an intuitive physical meaning. However, according to our definition, we can observe that when \( L_{\alpha} (S) \) is higher, Job \( j \) might bring larger impact to the system if it is released later than its scheduled starting time. Thus, in a LRIF concerned converting policy, we differentiate the jobs with high \( L_{\alpha} (S) \) and the ones with low \( L_{\alpha} (S) \).

9. H-CSR – Step 1: ‘HS’ is used to schedule the jobs?

Based on the results from the numerical experiments presented above, we study the scheduling problem with compressible and stochastic release dates on the basis of H-CDR(HS).

![Figure 6.2: An example to the superiority of LPT rule when multiple jobs have the same deadline](image)

Table 6.3: Comparing H-CDR(RS) and H-CDR(HS)

<table>
<thead>
<tr>
<th>( \delta )</th>
<th>CPU time</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>H-CDR(RS)</td>
<td>H-CDR(HS)</td>
<td>H-CDR(RS)</td>
</tr>
<tr>
<td>200</td>
<td>10h52m</td>
<td>10m34s</td>
</tr>
<tr>
<td>100</td>
<td>5h56m</td>
<td>5m6s</td>
</tr>
<tr>
<td>50</td>
<td>3h8m</td>
<td>1m47s</td>
</tr>
<tr>
<td>20</td>
<td>2h12m</td>
<td>10.2s</td>
</tr>
</tbody>
</table>

the worst-case and average complexity both at \( O(J^2) \) ([Knuth 1998]). We use H-CDR(RS) and H-CDR(HS) to refer to the H-CDRs where RS and HS are employed, respectively. The performances of H-CDR(RS) and H-CDR(HS) are compared in the following numerical experiment.

In the numerical experiment we set \( J = 200 \) and \( H = 240 \), and the other settings are identical to the ones we used in Section 6.3.1. The schedules obtained by H-CDR(RS) and H-CDR(HS) with different settings of \( \delta \) are compared in Table 6.3. In Table 6.3, it can be observed that along with the increase of \( \delta \), the obtained total cost decreases in a decreasing rate, and for each setting of \( \delta \), H-CDR(HS) can obtain a better result within a reasonable time.

Based on the results from the numerical experiments presented above, we study the scheduling problem with compressible and stochastic release dates on the basis of H-CDR(HS).
6.4 Scheduling with Compressible and Stochastic Release Dates

We name the scheduling problem when all the jobs have compressible and stochastic release dates as the CSR problem.

In this work, we study the scheduling method in which the stochastic release dates are converted to deterministic ones, which are named as the “converted release dates”. This method has been applied in a local window and door company, where the production of colored glass is outsourced to a subcontractor located at a different province. The subcontractor offers two types of orders, i.e., regular order and rush order, which are under different prices with different expected leadtimes. There is also a probability that the glass is broken during the transportation, for which the subcontractor has to redo the job and delay the delivery. When the company schedules its production, it converts the random leadtime for the delivery of glass to a deterministic value. However, it often happens that due to the delay of some glass, the production cannot proceed and the production line is idled.

We name the principle which regulates how to convert the stochastic release dates to the deterministic ones as the “converting policy”. In this work, we first present three simple converting policies which are applied by the mentioned company, i.e., the optimistic (O) policy, the pessimistic (P) policy and the neutral (N) policy. Then we propose a novel converting policy with concerns of the impact of late release.

6.4.1 The optimistic policy, the pessimistic policy and the neutral policy

Suppose that for any Job $j$ in a sequence, there is $f_j^R(r|\xi_j)$ for any given $\xi_j$. Then we denote that in the optimistic policy, the pessimistic policy and the neutral policy, the converted release date of Job $j$ under $\xi_j$ is $r_j^O(\xi_j)$, $r_j^P(\xi_j)$ and $r_j^N(\xi_j)$, respectively. Without loss of generality, we assume that for any $\xi_j$, $r_j$ is distributed from the integer set $[\underline{r}_j(\xi_j), \overline{r}_j(\xi_j)]$. Then the optimistic policy, pessimistic policy and neutral policy can be described as follows.
Optimistic policy In the optimal policy, the converted release date is set to the earliest possible value when $\xi_j$ is given, i.e.,

$$r^O_j(\xi_j) = \xi_j(r_j).$$

Pessimistic policy In the pessimistic policy, the converted release date is set to the latest possible value when $\xi_j$ is given, i.e.,

$$r^P_j(\xi_j) = \tau_j(\xi_j).$$

Neutral policy In the neutral policy, the converted release date is set to the expected value of $r_j$ when $\xi_j$ is given, i.e.,

$$r^N_j(\xi_j) = \left\lceil \sum_{r=r_j(\xi_j)} r f^R_j(r|\xi_j) \right\rceil.$$ (6.10)

Note that in (6.10), we take the ceiling of the real value of expected release date to consist with the assumption that the release dates are random integers.

When we have a converted release date based on any converting policy, denoted by $r^*_j(\xi_j)$, in order to apply the job sequencing algorithm for CDR problem proposed in Section 6.3, $r^L_j$ and $\xi_j(r_j)$ have to be determined. Supposing that $\Xi_j$ is the set of available compressing price options for Job $j$, then we have $r^L_j$ as

$$r^L_j = \min_{\xi \in \Xi_j} r^*_j(\xi).$$

Supposing that there is a set $\Xi^{r_j}$ such that for any $\xi \in \Xi^{r_j}$, the converted release date $r^*_j(\xi)$ equals to $r_j$. Then we obtain $p_j(r_j)$ as:

$$\xi_j(r_j) = \begin{cases} 
\min_{\xi \in \Xi^{r_j}} \xi & \text{if } \Xi^{r_j} \neq \emptyset \\
\xi_j(r_j - 1) & \text{if } \Xi^{r_j} = \emptyset
\end{cases}.$$ (6.11)

When $r^L_j$ and $\xi_j(r_j)$ are determined, the H-CDR can be applied for solving the heuristic schedule of CSR problem. For clarity, we name the CSR scheduling algorithms with the optimistic policy, the pessimistic policy and the neutral policy as H-CSR(O), H-CSR(P) and H-CSR(N), respectively.
6.4.2 Converting policy with concerns of the impact of late release

In CSR, due to the stochasticity of the jobs’ release dates, changing job sequence may happen to avoid system idle when some jobs are released later than their scheduled start time in actual production. However, because the start time of each job is constrained by the actual release date of the job, some idle time might not be avoided and thus an extra overtime might be incurred to ensure that all the jobs can be delivered before the due dates. In this work we coin the concept of a job’s late-release-impact factor (LRIF) which measures the impact incurred by a job when it is released later than its scheduled start time. We denote the schedule in the CSR problem by \( S \) such that \( S = (\vec{\xi}, \vec{s}, \vec{v}) \) where the vectors \( \vec{\xi}, \vec{s} \) and \( \vec{v} \) are the compressing fees, the start times and the overtime workloads for all the jobs, respectively.

Suppose that in a job schedule, \( S \), any Job \( j \)'s scheduled start time is \( s_j \), then if the actual release date of Job \( j \), \( r_j \), is later than \( s_j \), an additional overtime may be incurred. Denoting by \( \Delta v_j(r_j) \) the expected additional overtime incurred by Job \( j \) when the actual release date of Job \( j \) is \( r_j \), then Job \( j \)'s LRIF under schedule \( S \), denoted by \( L_j(S) \), is defined as:

\[
L_j(S) = \sum_{r_j = [s_j] + 1}^{r_j(\xi_j)} \frac{\Delta v_j(r_j) f_j^R(r_j | \xi_j)}{p_j}.
\]  

(6.12)

It is obvious that \( L_j \in [0, 1] \) because \( \Delta v_j(r) \in [0, p_j] \) for any \( r \). From (6.12), we can conclude that if a job has a higher LRIF, it is more likely to incur additional overtime in actual production. Because increasing a job’s scheduled start time can reduce its LRIF, in our LRIF concerned converting policy, when the compressing fee is given, a job with higher LRIF will have a bigger chance to be assigned with a later converted release date. For clarity, we use a simple example which includes only 2 jobs to explain the definition of \( L_j(S) \).

Example: Suppose that in a given schedule \( S \), there are 2 jobs having deadline \( d_1 = d_2 = 2 \) and processing time \( p_1 = p_2 = H \). The compressing fees of the two jobs are given as \( \xi_1 \) and
Table 6.4: Example: four possible incurred overtimes with different realization of \( r_1 \) and \( r_2 \)

<table>
<thead>
<tr>
<th>( r_1 )</th>
<th>( r_2 )</th>
<th>Overtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>( H )</td>
</tr>
</tbody>
</table>

\( \xi_2 \). We assume that under \( \xi_1 \) and \( \xi_2 \), the pmf of the two jobs’ release dates are

\[
f_1^R(r_1|\xi_1) = \begin{cases} 
0.6 \text{ if } r = 0 \\
0.4 \text{ if } r = 1 
\end{cases}
\]

and

\[
f_2^R(r_2|\xi_2) = \begin{cases} 
0.2 \text{ if } r_2 = 0 \\
0.8 \text{ if } r_2 = 1 
\end{cases}
\]

respectively. We display the four possible cases in which \( r_1 \) and \( r_2 \) are realized with different values as in Table 6.4. It is straightforward to find the optimal schedule with the actual release date, and the overtimes under the optimal schedule for all cases are also displayed in Table 6.4. It can be observed that overtime only occurs when \( r_1 = r_2 = 1 \).

First we study the case where \( s_1 = 0 \). From Table 6.4, we can see that if Job 1 starts on its scheduled start time, Job 1 and Job 2 can be processed without overtime regardless Job 2’s release date, which indicates that the expected overtime is 0; otherwise if Job 1 is released later than its scheduled start time, i.e., \( r_1 = 1 \), the expected overtime can be obtained, from Table 6.4, at

\[
f_2^R(0|\xi_2) \times 0 + f_2^R(1|\xi_2) \times H = 0.8H.
\]

Hence, the expected additional overtime incurred when \( r_1 = 1 \), \( \Delta v_1(1) \), can be obtained at

\[
\Delta v_1(1) = 0.8H.
\]

From (6.12), we can finally obtain the LRIF of Job 1 in schedule \( S \) at

\[
L_1(S) = \frac{\Delta v_1(1) f_1^R(1|\xi_1)}{H} = 0.32.
\]
Similarly, we can also compute that when $s_1 = 1$, $L_1(S) = 0$.

In the example, we can compute the LRIF of each jobs, but it can be observed that if there are more jobs, computing LRIF will be difficult because it is difficult to compute $\Delta v_j(r)$. Hence, we design a “surrogate measure” to be the indicator of a job’s LRIF. Surrogate measures have been widely used in the literature when it is not practical to measure a variable, e.g. Goren and Sabuncuoglu (2008).

In a given job schedule, $S$, which has $J$ jobs, we use $L'_j(S)$ to denote the surrogate measure which indicates Job $j$’s LRIF if the schedule is $S$. $L'_j(S)$ is defined as

$$L'_j(S) = 1 - \frac{\sum_{\forall \hat{j} \in \Omega(j)} p_j F^R_j(s_j|\xi_j)}{p_j}, \tag{6.13}$$

where $\Omega(j) = \{ \hat{j} | s_{\hat{j}} \geq s_j \} \text{ and } \hat{j} \neq j \}$. In the actual production, when Job $j$ is released after its scheduled starting date, the jobs in $\Omega(j)$ will be checked for the availability to be processed during the idle incurred by the late release of Job $j$. From (6.13), it can be observed that $\sum_{\forall \hat{j} \in \Omega(j)} p_j F^R_j(s_j|\xi_j)$ actually computes the expected workload of all the jobs which can be processed during the idle incurred by the late release of Job $j$. If $L'_j(S) > 0$, it is more likely that the late release of Job $j$ will cause idle and overtime production. Note that as an indicator of LRIF, $L'_j(S)$ does not have an intuitive physical meaning.

In order to bring LRIF into consideration, we develop a heuristic scheduling algorithm for the CSR problem with the LRIF concerned converting policy, namely H-CSR(L), as follows.

**H-CSR(L):**

Step 1: Schedule the jobs with neutral policy, i.e., H-CSR(N), to obtain $S$.

Step 2: Compute $L'_j(S)$ for each scheduled job. If $L'_j(S) > 0$, then apply pessimistic policy to Job $j$.

Step 3: Schedule the jobs using H-CDR(HS) with the updated converting policy and update $S$. Check the $L'_j(S)$ for each job. If there are some jobs of which the release dates are converted using neutral policy but having $L'_j(S) > 0$, apply pessimistic policy to these jobs and then repeat Step 3; otherwise, STOP and the current $S$ is the final schedule.
Table 6.5: The compressing prices quoted by the component suppliers

<table>
<thead>
<tr>
<th>Option</th>
<th>Price ($p_j$)</th>
<th>Leadtime ($r_j$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super Rush</td>
<td>$3</td>
<td>0-2 days</td>
</tr>
<tr>
<td>Rush</td>
<td>$1</td>
<td>3-5 days</td>
</tr>
<tr>
<td>Regular</td>
<td>$0</td>
<td>6-9 days</td>
</tr>
</tbody>
</table>

6.4.3 Comparing H-CSR(O), H-CSR(P), H-CSR(N) and H-CSR(L)

We design a numerical experiment to test the efficiency of H-CSR(O), H-CSR(P), H-CSR(N) and H-CSR(L). In the numerical experiment, we suppose that for each job, there are three release-date options, which are displayed in Table 6.5. Without loss of generality, we suppose that the manufacturer chooses the release dates of all the jobs at the beginning of the scheduling horizon, i.e., $t = 0$, and then the magnitude of a job’s release date is the same as the magnitude of the leadtime shown in Table 6.5.

We set the processing time for each job to be distributed from 1 to 20. The due date of each order, $d_j$, is uniformly distributed from the integer interval $[3, 10]$. The overtime production cost for processing one unit of processing time is set to $2$. For the purpose of clarity, we use the term “planned total cost” to refer to the total cost incurred if all the jobs are released at the converted release dates, and use the term “actual total cost” to refer to the total cost incurred during the actual production which also includes the extra overtime cost incurred by the late releases of some jobs. We test the performance of different heuristics for different problem scales. We ran 50 times of our numerical experiment for each problem scale, and in each run we generate different series of jobs. The results obtained by different heuristics for different problem scales are displayed in Table 6.6, in which the values are the average of 50 runs. In Table 6.6, we use the abbreviations APTC and AATC to represent the average planned total cost and the average actual total cost, respectively. Because in H-CSR(L), some jobs’ release dates are converted using pessimistic converting policying, we use ANPP to represent the average number of jobs of which the release dates are converted by the pessimistic converting policy.
Table 6.6: Comparing different scheduling method for CSR problem

<table>
<thead>
<tr>
<th>J</th>
<th>c</th>
<th>H-CSR(O) APTC AATC</th>
<th>H-CSR(P) APTC AATC</th>
<th>H-CSR(N) APTC AATC</th>
<th>H-CSR(L) APTC AATC ANPP</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>24</td>
<td>208.2 274.2</td>
<td>293.0 279.2</td>
<td>246 272.3</td>
<td>257.3 261.8 4.2</td>
</tr>
<tr>
<td>50</td>
<td>60</td>
<td>457.3 563.5</td>
<td>618.4 568.4</td>
<td>503 529.0</td>
<td>518.2 515.1 12.6</td>
</tr>
<tr>
<td>100</td>
<td>120</td>
<td>673.3 789.2</td>
<td>854.6 782.8</td>
<td>709 757.3</td>
<td>722.0 738.1 18.2</td>
</tr>
<tr>
<td>200</td>
<td>240</td>
<td>1228.6 1476.7</td>
<td>1571.8 1480.7</td>
<td>1403.4 1456.9</td>
<td>1429.2 1448 26.0</td>
</tr>
</tbody>
</table>

It can be observed that in our numerical experiment, by H-CSR(O), the firm plans much lower total cost, but because the actual release dates are later than the converted ones, a lot of idles are incurred and hence more overtime happens, which leads to a high actual total cost. By H-CSR(P), no extra overtime is incurred, but because the manufacturer overestimates the converted release date for each price, extra unnecessary cost is incurred for purchasing rush orders or super rush orders, which also leads to a high actual total cost. By H-CSR(N), the manufacturer plans a total cost much closer to the actual total cost than by H-CSR(O) and H-CSR(P), and the manufacturer obtains lower actual cost in H-CSR(N). However, the manufacturer still obtains the actual total cost which is higher than the planned total cost because some late released jobs cause extra overtime production. It can be observed that by H-CSR(L), the firm obtains the lowest actual cost and the smallest difference between the planned total cost and the actual cost.

6.5 Chapter Conclusions

In this chapter, we study the CSR problem, i.e., the due-date-guaranteed scheduling problem with compressible and stochastic release dates. We first develop a heuristic, H-CDR(HS) for the CDR problem, i.e. the due-date-guaranteed scheduling problem with compressible and deterministic release dates. Then based on H-CDR(HS), we solve the CSR problem by proposing a heuristic algorithm, where a release date converting policy concerning the impact of the late releases of the jobs is applied. We compare the proposed LRIF-concerned
policy with other release date converting policies which are used in practice. The results of
the numerical experiment show that by applying the LRIF-concerned policy, the firm can
effectively control the compressing cost as well as the overtime production cost incurred by
the late releases of some jobs.
Chapter 7

Conclusions

In this theses work, we focuses on the decentralized coordination method for an OKP supply chain. We study the dynamic pricing strategy (DPS) for different cases and its influences on supply chain coordination. We also develop a series of computing and scheduling methodologies to facilitate the implementation of the proposed DPS. We studied the dynamic pricing strategies for three cases:

In the first case, we study a dynamic pricing strategy for a one-of-a-kind production (OKP) firm with two classes of orders (due-date guaranteed and due-date unguaranteed) at different prices to the sequentially arriving customers, who are also OKP production firms. The prices for two types of orders are quoted to each customer on its arrival. We study two problems in this setting. First, we model a DPS and compare our DPS with a static pricing strategy (SPS). Through a numerical test, we show that both the firm and its customers are better off when our DPS is employed, so that the DPS improves overall performance of the supply chain. Through an industry case, a custom window production firm, we show how to apply the proposed DPS when products are complex. We also develop a method to adaptively estimate the firm’s available capacity, the number of future arrivals and the distributions of the customers’ willingness to pay and impatience factor. The simulation result shows that, when multiple distribution parameters are unknown, the proposed parameter estimating method results in estimates close to the true values.

In the second case, we study the pricing problem for a third-party-logistics (3PL) provider that provides warehousing and less-than-truckload (LTL) transportation services. When customers arrive at the 3PL provider, they specify the delivery dates for their freight, and before the specified delivery dates, their freight is stocked in the 3PL provider’s warehouse. We
propose a dynamic pricing strategy (DPS) and develop a stochastic-nonlinear-programming (SNLP) model which computes the optimal freight rates for different delivery dates incorporating the 3PL provider’s current holding cost and available transportation capacity for each route. As customers are heterogeneous in their valuations and price sensitivities for delivery dates, and the distributions of the customers’ delivery date preferences are unknown to the 3PL provider, we develop an adjusted multinomial logit (MNL) function to predict customer choices so that our SNLP model can obtain near optimal freight rate settings. Through a simulation experiment, we show that the adjusted MNL function can be a good replacement for the mixed MNL function when the mixed MNL function is not applicable. Through simulation we also compare the 3PL provider’s profit, customer and social welfare using a DPS with a static pricing strategy. We show that with our DPS both the 3PL provider and its customers are better off, and the 3PL provider has different investment incentives for reducing holding costs and increasing transportation capacity. The results for this case can be also applied in similar settings that feature holding costs, limited production capacity and delivery-date-sensitive customers.

In the third case, we study dynamic pricing based on a practical make-to-order firm which is currently employing static pricing strategy. The firm has long-term contracted customers and spot-market customers, and the spot-market customers can be further divided into delivery-date-critical, delivery-date-flexible and delivery-date-unspecified customers. We propose two types of dynamic pricing strategies, i.e. the discount-based dynamic pricing strategy (DDP) and the premium-based dynamic pricing strategy (PDP). By studying the simplified supply chain through analytical method, we investigate how the dynamic pricing strategies affect the performance of a complex multi-echelon supply chain. We present simulations for a single-echelon (or two-layer) supply chain and a two-echelon (or three-layers) supply chain to test our analytical results. Based on analytical models and simulation experiments, we prove that within a supply chain, both upstream and downstream firms (or
end customers) are better off by adopting the dynamic pricing strategies, and between the
two dynamic pricing strategies, both upstream and downstream firms (or end customers) are
better off by adopting PDP. However, in some cases PDP is not applicable. We also propose
a method of implementing two types of dynamic pricing strategies.

To facilitate the supply chain coordination, we also develop a scheduling method for a
manufacturer when its suppliers offer different delivery times at different prices. We abstract
the problem as a one-machine scheduling problem which is featured by: (a) the release date
of each job is compressible and stochastic, (b) each job has to be delivered before its due
date (deadline) and (c) the manufacturer can expedite the production through overtime at
an extra cost. The objective function of the scheduling problem is to minimize the total
cost which includes the compressing cost and the overtime production cost. We propose a
heuristic algorithm in which the stochastic problem is converted to the deterministic problem
by a release-time “converting policy”. We coin a concept of a job’s late-release-impact factor
(LRIF) and we propose a LRIF based converting policy. We compare the LRIF based
converting policy with the ones used in practice, and the numerical test shows that the
LRIF based converting policy can obtain the schedule with the lowest actual total cost.
Bibliography


Li, H., Huh, W., 2011. Pricing multiple products with the multinomial logit and nested logit models: Conca


duction Economics 113 (1), 127–147.

Megow, N., Uetz, M., Vredeveld, T., 2006. Models and algorithms for stochastic online

Mendelson, H., Whang, S., 1990. Optimal incentive-compatible priority pricing for the
m/m/1 queue. Operations Research 38 (5), 870–883.

Production Planning & Control 1 (3), 170–178.

Econometric Society, 15–24.

linear buyer’s demand. IIE transactions 26 (2), 34–41.


Pinedo, M., 1983. Stochastic scheduling with release dates and due dates. Operations Re-
search 31 (3), 559–572.

ations Research 52 (2), 213–228.


Qi, X., 2011. Outsourcing and production scheduling for a two-stage flow shop. International


Appendix A

Proofs of Theorems

A.1 Proof of Theorem 5.1

From (5.8) and (5.3), we have that if $\hat{x}^D \leq c^A$,

$$p^S - p^D = \frac{\alpha(1 - \beta)(c^O - c^A)}{2}. $$

Because $c^O > c^A$, we have $p^D < p^S$ when $\hat{x}^D \leq c^A$. If $\hat{x}^D > c^A$, we have

$$p^S - p^D = \frac{\alpha(1 - \alpha)(1 - \beta)c^O}{2[1 - \alpha + \alpha\beta]} > 0. $$

Hence, $p^D < p^S$ always holds.

From (5.13) and (5.3), we have that if $\hat{x}^P \leq c^A$,

$$p^S - p^P = \frac{\alpha(1 - \beta)c^O + \alpha(1 + \beta)c^A}{2} > 0. $$

If $\hat{x}^D > c^A$, we have

$$p^S - p^P = \frac{\alpha(1 - \beta)c^O + \alpha(1 - \beta)c^A}{2[1 - \alpha\beta]} > 0. $$

Hence, $p^P < p^S$ always holds.

From (5.12b), we have that if $\hat{x}^P \leq c^A$

$$\frac{[1 - \alpha\beta]c^O + \alpha[1 - \beta]c^A}{2[1 - \alpha\beta]} \leq c^A$$

$$\Rightarrow [1 - \alpha\beta]c^O \leq [2 - \alpha - \alpha\beta]c^A \quad (A.1)$$

From (5.3), (5.13) and (A.1), we have that if $\hat{x}^P \leq c^A$,

$$p^S - p^P - x^P = \frac{\alpha(1 - \beta)c^O + [\alpha + \alpha\beta - 2]c^A}{2} < 0. $$
If \( \hat{x}^D > c^A \), we have

\[
p^{S*} - p^{P*} - x^{P*} = \frac{-[1 - \alpha]c^O}{2} < 0.
\]

Hence, \( p^{S*} < p^{P*} + x^{P*} \) always holds.

Summarizing the proofs above, we have that Theorem 5.1 is proved.

A.2 Proof of Theorem 5.2

We use two steps to prove Theorem 5.2. In Step 1, we prove \( \pi^D(p^{D*}, x^{D*}) < \pi^P(p^{P*}, x^{P*}) \), and in Step 2, we prove \( \pi^S(p^{S*}) < \pi^D(p^{D*}, x^{D*}) \).

Step 1: Prove \( \pi^D(p^{D*}, x^{D*}) < \pi^P(p^{P*}, x^{P*}) \)

We prove \( \pi^D(p^{D*}, x^{D*}) < \pi^P(p^{P*}, x^{P*}) \) by considering different conditions as shown in Steps 1.1-1.3.

**Step 1.1: Prove** \( \pi^P(p^{P*}, x^{P*}) > \pi^D(p^{D*}, x^{D*}) \) **when** \( \hat{x}^D \leq c^A \)

From (5.8), we have

\[
x^{D*}\big|_{\hat{x}^D \leq c^A} = c^A \quad \text{and} \quad p^{D*}\big|_{\hat{x}^D \leq c^A} = \frac{V + \alpha \beta c^O + \alpha c^A - \alpha \beta c^A}{2}
\]

(A.2)

Substituting \( x^D \) and \( p^D \) in (5.6) with \( x^{D*}\big|_{\hat{x}^D \leq c^A} \) and \( p^{D*}\big|_{\hat{x}^D \leq c^A} \), we have

\[
\pi^D(x^{D*}\big|_{\hat{x}^D \leq c^A}, p^{D*}\big|_{\hat{x}^D \leq c^A}) = \frac{[V - \alpha \beta c^O - \alpha c^A + \alpha \beta c^A]^2}{4V}
\]

(A.3)

Because \( \pi^P(p^{P*}, x^{P*}) \geq \pi^P(p^{P*}\big|_{x^P = c^A}, c^A) \), we only need

\[
\pi^P(p^{P*}\big|_{x^P = c^A}, c^A) > \pi^D(p^{D*}\big|_{\hat{x}^D \leq c^A}, x^{D*}\big|_{\hat{x}^D \leq c^A})
\]

to prove \( \pi^P(p^{P*}, x^{P*}) > \pi^D(p^{D*}\big|_{\hat{x}^D \leq c^A}, x^{D*}\big|_{\hat{x}^D \leq c^A}) \). From (5.13), we have

\[
p^{P*}\big|_{x^P = c^A} = \frac{V - \alpha c^A + \alpha \beta c^O - \alpha \beta c^A}{2}
\]

(A.4)
Substituting $x^P$ and $p^P$ in (5.11) with $c^A$ and $p^P|_{x^P=c^A}$, respectively, we have
\[
\pi^P(p^P|_{x^P=c^A}, c^A) = \frac{\alpha^2 \beta^2 [c^O]^2 - 2 \alpha^2 \beta c^O c^A + \alpha^2 \beta^2 [c^A]^2 - 2 \alpha^2 \beta c^O c^A}{2} + 2 \alpha^2 \beta [c^A]^2 + \alpha^2 [c^A]^2 + 4 \alpha \beta c^O c^A - 2 \alpha \beta c^O V - 4 \alpha \beta [c^A]^2 + 2 \alpha \beta c^A V - 2 \alpha c^A V + V^2]/[4V]
\]
(A.5)

From (A.3) and (A.5), we can obtain
\[
\pi^P(p^P|_{x^P=c^A}, c^A) - \pi^D(p^D|_{x^D \leq c^A}, x^D|_{x^D \leq c^A}) = \frac{\alpha \beta c^A [c^O - c^A][1 - \alpha]}{V} > 0
\]

Hence, we have $\pi^P(p^P, x^P) > \pi^D(p^D, x^D)$ when $\hat{x}^D \leq c^A$. Step 1.1 ends.

**Step 1.2: Prove** $\pi^P(p^P, x^P) > \pi^D(p^D, x^D)$ **when** $\hat{x}^P \geq c^A$ **and** $\hat{x}^D \geq c^A$

From (5.7b), we have that when $\hat{x}^D \geq c^A$
\[
\frac{c^A - \alpha c^A + \alpha \beta c^O + \alpha \beta c^A}{2 \alpha \beta - 2 \alpha c^A + 2 c^A} \geq c^A
\]
\[
\Rightarrow c^A \leq \frac{\alpha \beta c^O}{1 - \alpha + \alpha \beta}
\]
(A.6)

From (5.8), we have
\[
x^D|_{x^D \geq c^A} = \frac{[1 - \alpha D]c^A + \alpha D \beta [c^A + c^O]}{2[1 - \alpha D + \alpha D \beta]}
\]
(A.7a)
\[
p^D|_{x^D \geq c^A} = \frac{[1 - \alpha D]V + \alpha D \beta [V + c^O]}{2[1 - \alpha D + \alpha D \beta]}
\]
(A.7b)

Substituting $x^D$ and $p^D$ in (5.6) with $p^D|_{x^D \geq c^A}$ and $x^D|_{x^D \geq c^A}$, we have
\[
\pi^D(p^D|_{x^D \geq c^A}, x^D|_{x^D \geq c^A})
\]
\[
= \frac{\alpha^2 \beta^2 [c^O]^2 - 2 \alpha^2 \beta c^O V - \alpha^2 \beta^2 [c^A]^2 + 2 \alpha^2 \beta c^A V + 2 \alpha^2 \beta c^O V + 2 \alpha^2 \beta [c^A]^2}{2} - 4 \alpha^2 \beta c^A V - \alpha^2 [c^A]^2 + 2 \alpha^2 c^A V - 2 \alpha \beta c^O V - \alpha \beta [c^A]^2 + 2 \alpha \beta c^A V + \alpha \beta V^2
\]
\[
+ \alpha [c^A]^2 - 2 \alpha c^A V - \alpha V^2 + V^2]/[4V][\alpha \beta - \alpha + 1]
\]
(A.8)

From (5.13), we have
\[
p^P|_{x^P \geq c^A} = \frac{[1 - \alpha \beta]V - \alpha [1 - \beta]c^A}{2[1 - \alpha \beta]}
\]
(A.9a)
Substituting \( x^P \) and \( p^P \) in (5.11) with \( x^P^*_{\hat{x}^P \geq c^A} \) and \( p^P^*_{\hat{x}^P \geq c^A} \), we have

\[
\pi^P(p^P^*_{\hat{x}^P \geq c^A}, x^P^*_{\hat{x}^P \geq c^A}) = \left[ \alpha^2 \beta^2 [c^O]^2 - 2 \alpha^2 \beta^2 c^O V - \alpha^2 \beta^2 [c^A]^2 + 2 \alpha^2 \beta^2 c^A V + 2 \alpha^2 \beta [c^A] V - \alpha^2 [c^A] V - \alpha \beta [c^O]^2 + 2 \alpha \beta c^O V - 2 \alpha \beta c^A V + \alpha \beta V^2 + 2 \alpha c^A V - V^2 \right] / \left[ 4V(\alpha \beta - 1) \right].
\] (A.10)

From (A.8) and (A.10), we can obtain

\[
\pi^P(p^P^*_{\hat{x}^P \geq c^A}, x^P^*_{\hat{x}^P \geq c^A}) - \pi^D(p^D^*_{\hat{x}^D \geq c^A}, x^D^*_{\hat{x}^D \geq c^A}) = \frac{\alpha(1 - \alpha) \kappa_1}{4V(1 - \alpha \beta)(1 - \alpha + \alpha \beta)}
\] (A.11)

where

\[
\kappa_1 = [\beta - \alpha \beta^2][c^O]^2 - (1 - \alpha + \alpha \beta)(1 - \beta)[c^A]^2.
\] (A.12)

From (A.11), we can conclude that we only need \( \kappa_1 > 0 \) to prove \( \pi^P(p^P^*_{\hat{x}^P \geq c^A}, x^P^*_{\hat{x}^P \geq c^A}) - \pi^D(p^D^*_{\hat{x}^D \geq c^A}, x^D^*_{\hat{x}^D \geq c^A}) > 0. \)

Substitute \( c^A \) in (A.12) with the right-hand side of (6). After simple algebra, we have

\[
\kappa_1 \geq \frac{[1 - \alpha] \beta [c^O]^2}{1 - \alpha + \alpha \beta} > 0.
\]

Hence, we have \( \pi^P(p^P^*, x^P^*) > \pi^D(p^D^*, x^D^*) \) when \( \hat{x}^P \geq c^A \) and \( \hat{x}^D \geq c^A \). Step 1.2 ends.

**Step 1.3: Prove** \( \pi^P(p^P^*, x^P^*) > \pi^D(p^D^*, x^D^*) \) when \( \hat{x}^D \geq c^A \) and \( \hat{x}^P \leq c^A \)

From (A.1) and (6), we can obtain the range of \( c^O \), when \( \hat{x}^P \leq c^A \) and \( \hat{x}^D \geq c^A \), as

\[
c^O \in [\kappa_{2,1}, \kappa_{2,2}]
\] (A.13)

where

\[
\kappa_{2,1} = \frac{1 - \alpha + \alpha \beta}{\alpha \beta} c^A \quad \text{and} \quad \kappa_{2,2} = \frac{2 - \alpha - \alpha \beta}{1 - \alpha \beta} c^A.
\]
From (5.13), we have

\[ \pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) = \pi^P(p^{P*}|_{xP = c_A}, c^A), \]

where \( \pi^P(p^{P*}|_{xP = c_A}, c^A) \) can be obtained from (A.5). Then from (A.5) and (A.8), we can obtain

\[
\pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) = \pi^D(p^{D*}|_{\hat{x}D \geq c_A}, x^{D*}|_{\hat{x}D \geq c_A})
\]

\[
= \frac{\alpha[\kappa_{3,1}[c^O]^2 + \kappa_{3,2}c^O + \kappa_{3,3}]}{4V[1 + \alpha\beta - \alpha]} \]  

(A.14)

where

\[
\kappa_{3,1} = -\alpha^2\beta^2[1 - \beta],
\]

\[
\kappa_{3,2} = -2\beta[2 - \alpha - \alpha\beta][1 + \alpha\beta - \alpha]c^A \text{ and }
\]

\[
\kappa_{3,3} = -[1 + \alpha\beta - \alpha][1 - \alpha + 3\beta - 2\alpha\beta - \alpha\beta^2][c^A]^2.
\]

From (A.14), we can find that we only need \( \kappa_{3,1}[c^O]^2 + \kappa_{3,2}c^O + \kappa_{3,3} > 0 \) to prove \( \pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) - \pi^D(p^{D*}|_{\hat{x}D \geq c_A}, x^{D*}|_{\hat{x}D \geq c_A}) > 0 \). It can be observed that \( \kappa_{3,1} < 0 \), which means \( \kappa_{3,1}[c^O]^2 + \kappa_{3,2}c^O + \kappa_{3,3} \) is concave in \( c^O \). Because \( c^O \in [\kappa_{2,1}, \kappa_{2,2}] \) in (A.13), it can be concluded that we only need to show \( \kappa_{3,1}[c^O]^2 + \kappa_{3,2}c^O + \kappa_{3,3} > 0 \) when \( c^O \) is at \( \kappa_{2,1} \) and \( \kappa_{2,2} \), or

\[
\left[ \pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) - \pi^D(p^{D*}|_{\hat{x}D \geq c_A}, x^{D*}|_{\hat{x}D \geq c_A}) \right]_{c^O=\kappa_{2,1}} \geq 0 \quad \text{and} \quad \ (A.15)
\]

\[
\left[ \pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) - \pi^D(p^{D*}|_{\hat{x}D \geq c_A}, x^{D*}|_{\hat{x}D \geq c_A}) \right]_{c^O=\kappa_{2,2}} \geq 0 \quad \text{(A.16)}
\]

When \( c^O = \kappa_{2,1} \), there is \( \hat{x}D = c^A \) and \( \hat{x}P \leq c^A \). Thus, we can obtain

\[
\left[ \pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) - \pi^D(p^{D*}|_{\hat{x}D \geq c_A}, x^{D*}|_{\hat{x}D \geq c_A}) \right]_{c^O=\kappa_{2,1}} = \pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) - \pi^D(p^{D*}|_{xD = c_A}, x^{D*}|_{\hat{x}D = c_A})
\]

From Step 1.1, we have

\[
\pi^P(p^{P*}|_{\hat{x}P \leq c_A}, x^{P*}|_{\hat{x}P \leq c_A}) > \pi^D(p^{D*}|_{\hat{x}D = c_A}, x^{D*}|_{\hat{x}D = c_A}).
\]
Thus, (A.15) is proved.

When \( c^A = \kappa_{2,2} \), there is \( \hat{x}^D \geq c^A \) and \( \hat{x}^P = c^A \). Thus, we can obtain

\[
\left[ \pi^P(p^P|_{\hat{x}^P \leq c^A}, x^P|_{\hat{x}^P \leq c^A}) - \pi^D(p^D|_{\hat{x}^D \geq c^A}, x^D|_{\hat{x}^D \geq c^A}) \right]_{c^O = \kappa_{2,2}}
\]

\[= \pi^P(p^P|_{\hat{x}^P = c^A}, x^P|_{\hat{x}^P = c^A}) - \pi^D(p^D|_{\hat{x}^D \geq c^A}, x^D|_{\hat{x}^D \geq c^A}) \]

From Step 1.2, we have

\[
\pi^P(p^P|_{\hat{x}^P = c^A}, x^P|_{\hat{x}^P = c^A}) > \pi^D(p^D|_{\hat{x}^D \geq c^A}, x^D|_{\hat{x}^D \geq c^A})
\]

Thus (A.16) is proved.

Hence, we have \( \pi^P(p^P, x^P) > \pi^D(p^D, x^D) \) when \( \hat{x}^D \geq c^A \) and \( \hat{x}^P \leq c^A \). Step 1.3 ends.

Summarizing the proofs in Steps 1.1-1.3, we can conclude that \( \pi^P(p^P, x^P) > \pi^D(p^D, x^D) \) always holds. Step 1 ends.

Step 2: Prove \( \pi^D(p^D, x^D) > \pi^S(p^S) \)

Substituting \( p^S \) in (5.2) with \( p^S \) in (5.3), we have

\[
\pi^S(p^S) = \frac{[V - \alpha c^O]^2}{4V} \tag{A.17}
\]

Because \( \pi^D(p^D, x^D) \geq \pi^D(p^D|_{x^D = c^A}, c^A) \), we only need \( \pi^S(p^S) < \pi^D(p^D|_{x^D = c^A}, c^A) \) to prove \( \pi^S(p^S) < \pi^D(p^D, x^D) \).

From (5.8), we have

\[
p^D|_{x^D = c^A} = \frac{V + \alpha \beta c^O + \alpha c^A - \alpha \beta c^A}{2} \tag{A.18}
\]

Substituting \( x^D \) and \( p^D \) in (5.6) with \( c^A \) and \( p^D|_{x^D = c^A} \), respectively, we have

\[
\pi^D(p^D|_{x^D = c^A}, c^A) = \frac{[V - \alpha \beta c^O - \alpha c^A + \alpha \beta c^A]^2}{4V} \tag{A.19}
\]

From (A.17) and (A.19), we can obtain

\[
\pi^D(p^D|_{x^D = c^A}, c^A) - \pi^S(p^S) = \frac{\alpha c^O - c^A}[1 - \beta] \kappa_4 \frac{\alpha c^O}{4V} \tag{A.20}
\]
\[ \kappa_4 = 2V - \alpha c^O - \alpha \beta c^O - [\alpha - \alpha \beta]c^A. \]  
(A.21)

From (A.20), it can be observed that we only need \( \kappa_4 > 0 \) to prove \( \pi^D(p^D_{\hat{x}D \leq c^A}, c^A) - \pi^S(p^S) > 0 \). Because \( \alpha - \alpha \beta > 0 \), we have

\[ [\alpha - \alpha \beta]c^A < [\alpha - \alpha \beta]c^O \]  
(A.22)

From (A.21) and (A.22), we have

\[ \kappa_4 > 2V - \alpha c^O - \alpha \beta c^O - [\alpha - \alpha \beta]c^O = 2V - 2\alpha c^O > 0. \]  
(A.23)

Then we have \( \pi^D(p^D_{\hat{x}D = c^A}, c^A) > \pi^S(p^S) \) which leads to \( \pi^D(p^D_{\hat{x}D = c^A}, x^D) > \pi^S(p^S) \). Step 2 ends.

Based on the proofs in Steps 1-2, we have \( \pi^P(p^P_{\hat{x}P}, x^P) > \pi^D(p^D_{\hat{x}D \leq c^A}, x^D) > \pi^S(p^S) \). Theorem 5.2 is proved.

A.3 Proof of Theorem 5.3

Similar as the proof for Theorem 5.2, we use two steps to prove Theorem 5.3. In Step 1, we prove \( w^D(p^D_{\hat{x}D \leq c^A}, x^D) < w^P(p^P_{\hat{x}P}, x^P) \), and in Step 2, we prove \( w^S(p^S) < w^D(p^D_{\hat{x}D = c^A}, x^D) \).

Step 1: Prove \( w^D(p^D_{\hat{x}D \leq c^A}, x^D) < w^P(p^P_{\hat{x}P}, x^P) \)

We prove \( w^D(p^D_{\hat{x}D \leq c^A}, x^D) < w^P(p^P_{\hat{x}P}, x^P) \) by considering different conditions as shown in Steps 1.1-1.4.

**Step 1.1: Prove** \( w^P(p^P_{\hat{x}P}, x^P) > w^D(p^D_{\hat{x}D \leq c^A}, x^D) \) when \( \hat{x}D \leq c^A \) and \( \hat{x}P \leq c^A \)

Substituting \( x^D \) and \( p^D \) in (5.15) with \( x^D|_{\hat{x}D \leq c^A} \) and \( p^D|_{\hat{x}D \leq c^A} \) in (A.2), we have

\[ w^D(p^D_{\hat{x}D \leq c^A}, x^D|_{\hat{x}D \leq c^A}) = \frac{[V - \alpha \beta c^O - \alpha c^A + \alpha \beta c^A]^2}{8V} \]  
(A.24)
Substituting $x^P$ and $p^P$ in (5.16) with $c^A$ and $p^P x |_{x = c^A}$ in (A.4), respectively, we have

$$w^P(p^P |_{x^P \leq c^A}, x^P |_{x^P \leq c^A})$$

$$= \left( \alpha^2 \beta^2 |c^O|^2 - 2\alpha^2 \beta^2 c^O c^A + \alpha^2 \beta^2 |c^A|^2 + 2\alpha^2 \beta c^O c^A \right.$$ 

$$- 2\alpha^2 \beta |c^A|^2 - 3\alpha^2 |c^A|^2 - 2\alpha \beta c^O V + 2\alpha \beta c^A V + 4\alpha |c^A|^2$$

$$- 2\alpha c^A V + V^2 \right) / (8V) \quad (A.25)$$

From (A.24) and (A.25), we can obtain

$$w^P(p^P |_{x^P \leq c^A}, x^P |_{x^P \leq c^A}) - w^D(p^D |_{x^D \leq c^A}, x^D |_{x^D \leq c^A})$$

$$= \frac{\alpha[1 - \alpha]|c^A|^2}{2V} > 0 \quad (A.26)$$

Hence, we have $w^P(p^P, x^P) > w^D(p^D, x^D)$ when $\hat{x}^D \leq c^A$ and $\hat{x}^P \leq c^A$. Step 1.1 ends.

**Step 1.2:** Prove $w^P(p^P, x^P) > w^D(p^D, x^D)$ **when $\hat{x}^D \leq c^A$ and $\hat{x}^P \geq c^A$**

From (A.26), we can conclude that we only need

$$w^P(p^P |_{x^P \geq c^A}, x^P |_{x^P \geq c^A}) \geq w^P(p^P |_{x^P \leq c^A}, x^P |_{x^P \leq c^A})$$

To prove $w^P(p^P |_{x^P \geq c^A}, x^P |_{x^P \geq c^A}) > w^D(p^D |_{x^D \leq c^A}, x^D |_{x^D \leq c^A})$.

Substituting $x^P$ and $p^P$ in (5.16) with $x^P |_{x^P \geq c^A}$ and $p^P x |_{x^P \geq c^A}$ in (A.9), we have

$$w^P(p^P |_{x^P \geq c^A}, x^P |_{x^P \geq c^A})$$

$$= \left[ \alpha^2 \beta^2 |c^O|^2 - 2\alpha^2 \beta^2 c^O V - \alpha^2 \beta^2 |c^A|^2 + 2\alpha^2 \beta^2 c^A V - 2\alpha^2 \beta |c^A|^2 \right.$$ 

$$- 2\alpha^2 \beta c^A V + 3\alpha^2 |c^A|^2 - \alpha \beta |c^O|^2 + 2\alpha \beta c^O V + 4\alpha \beta |c^A|^2$$

$$- 2\alpha \beta c^A V + \alpha \beta V^2 - 4\alpha |c^A|^2 + 2\alpha c^A V - V^2 \right) / [8V[\alpha \beta - 1]]. \quad (A.27)$$

From (A.25) and (A.27), we have

$$w^P(p^P |_{x^P \geq c^A}, x^P |_{x^P \geq c^A}) - w^P(p^P |_{x^P \leq c^A}, x^P |_{x^P \leq c^A})$$

$$= \frac{\kappa_{5.1} \kappa_{5.2}}{8V[1 - \alpha \beta]} \quad (A.28)$$
where
\[ \kappa_{5.1} = [1 - \alpha \beta]c^0 - [2 - \alpha - \alpha \beta]c^A \text{ and} \]
\[ \kappa_{5.2} = [1 - \alpha \beta]c^0 + [2 - 3\alpha + \alpha \beta]c^A. \]

From (5.12b), we have that when \( \hat{x}^P \geq c^A \)
\[ \frac{[1 - \alpha^P \beta]c^0 + \alpha[1 - \beta]c^A}{2[1 - \alpha \beta]} \geq c^A \]
\[ \Rightarrow c^0 \geq \frac{2 - \alpha - \alpha \beta}{1 - \alpha \beta}c^A \]
(A.29)

Substituting \( c^0 \) in \( \kappa_{5.1} \) and \( \kappa_{5.2} \) with the right-hand side of (A.29), we have
\[ k_{5.1} \geq 0 \text{ and } k_{5.2} \geq 4[1 - \alpha]c^A > 0 \]

Thus, from (A.28), we have \( w^P(p^{P*}|_{\hat{x}^P \geq c^A}, x^{P*}|_{\hat{x}^P \geq c^A}) \geq w^P(p^{P*}|_{\hat{x}^P \leq c^A}, x^{P*}|_{\hat{x}^P \leq c^A}) \), which leads to \( w^P(p^{P*}|_{\hat{x}^P \geq c^A}, x^{P*}|_{\hat{x}^P \geq c^A}) > w^D(p^{D*}|_{\hat{x}^D \leq c^A}, x^{D*}|_{\hat{x}^D \leq c^A}) \).

Hence, we have \( w^P(p^{P*}, x^{P*}) > w^D(p^{D*}, x^{D*}) \) when \( \hat{x}^D \leq c^A \) and \( \hat{x}^P \geq c^A. \) Step 1.2 ends.

**Step 1.3: Prove** \( w^P(p^{P*}, x^{P*}) > w^D(p^{D*}, x^{D*}) \) when \( \hat{x}^D \geq c^A \) and \( \hat{x}^P \geq c^A \)

Substituting \( x^D \) and \( p^D \) in (5.15) with \( x^{D*}|_{\hat{x}^D \geq c^A} \) and \( p^{D*}|_{\hat{x}^D \geq c^A} \) in (A.7), we have
\[ w^D(p^{D*}|_{\hat{x}^D \geq c^A}, x^{D*}|_{\hat{x}^D \geq c^A}) \]
\[ = \left[ \alpha^2 \beta^2[c^O]^2 - 2\alpha^2 \beta^2 c^A V - \alpha^2 \beta^2[c^A]^2 + 2\alpha^2 \beta^2 c^A V + 2\alpha^2 \beta c^0 V + 2\alpha^2 \beta[c^A]^2 \right. \]
\[ -4\alpha^2 \beta^2 c^A V - \alpha^2[c^A]^2 + 2\alpha^2 \beta c^0 V - \alpha \beta[c^A]^2 + 2\alpha \beta^2 c^A V + \alpha^2 V^2 \]
\[ +\alpha[c^A]^2 - 2\alpha c^A V - \alpha V^2 + V^2] / [8V[\alpha^2 \beta^2 + 1]] \]
(A.30)

From (A.27) and (A.30), we can obtain
\[ w^P(p^{P*}|_{\hat{x}^P \geq c^A}, x^{P*}|_{\hat{x}^P \geq c^A}) - w^D(p^{D*}|_{\hat{x}^D \geq c^A}, x^{D*}|_{\hat{x}^D \geq c^A}) \]
\[ = \frac{\alpha[1 - \alpha]k_6}{8V[1 - \alpha \beta][1 - \alpha + \alpha \beta]} \]
(A.31)

where
\[ k_6 = \beta[1 - \alpha \beta][c^O]^2 + 3[1 - \alpha + \alpha \beta][1 - \beta][c^A]^2 > 0. \]
(A.32)
From (A.31), we can conclude that \( w^P(p^{P*}|_{\hat{x}P\leq c^A}, x^{P*}|_{\hat{x}P\leq c^A}) - w^D(p^{D*}|_{\hat{x}D\geq c^A}, x^{D*}|_{\hat{x}D\geq c^A}) > 0. \)

Hence, we have \( w^P(p^{P*}, x^{P*}) > w^D(p^{D*}, x^{D*}) \) when \( \hat{x}D \geq c^A \) and \( \hat{x}P \leq c^A \). Step 1.2 ends.

**Step 1.4: Prove** \( w^P(p^{P*}, x^{P*}) > w^D(p^{D*}, x^{D*}) \) when \( \hat{x}D \geq c^A \) and \( \hat{x}P \leq c^A \)

From (A.25) and (A.30), we can obtain

\[
\frac{w^P(p^{P*}|_{\hat{x}P\leq c^A}, x^{P*}|_{\hat{x}P\leq c^A}) - w^D(p^{D*}|_{\hat{x}D\geq c^A}, x^{D*}|_{\hat{x}D\geq c^A})}{8V[1 + \alpha \beta - \alpha]} = \frac{\alpha[k_{7.1}[c^O]^2 + k_{7.2}c^O + k_{7.3}]}{8V[1 + \alpha \beta - \alpha]} \tag{A.33}
\]

where

\[
k_{7.1} = -\alpha^2 \beta^2[1 - \beta],
k_{7.2} = 2\alpha \beta[1 - \beta][1 + \alpha \beta - \alpha]c^A \quad \text{and} \quad k_{7.3} = [1 + \alpha \beta - \alpha][\beta - 3\alpha - 2\alpha \beta + \alpha \beta^2 + 3][c^A]^2.
\]

From (A.33), we can find that we only need \( k_{7.1}[c^O]^2 + k_{7.2}c^O + k_{7.3} > 0 \) to prove \( [w^P(p^{P*}, x^{P*}) - w^D(p^{D*}, x^{D*})]|_{\hat{x}D\geq c^A, \hat{x}P\leq c^A} > 0 \). It can be observed that \( k_{7.1} < 0 \), which means \( k_{7.1}[c^O]^2 + k_{7.2}c^O + k_{7.3} \) is concave in \( c^O \). Because when \( \hat{x}D \geq c^A \) and \( \hat{x}P \leq c^A \), \( c^O \in [k_{2.1}, k_{2.2}] \) as in (A.13), it can be concluded that we only need to show \( k_{7.1}[c^O]^2 + k_{7.2}c^O + k_{7.3} > 0 \) when \( c^O \) is at \( k_{2.1} \) and \( k_{2.2} \), or

\[
[w^P(p^{P*}|_{\hat{x}P\leq c^A}, x^{P*}|_{\hat{x}P\leq c^A}) - w^D(p^{D*}|_{\hat{x}D\geq c^A}, x^{D*}|_{\hat{x}D\geq c^A})]_{c^O=k_{2.1}} \geq 0 \quad \text{and} \quad (A.34)
\]

\[
[w^P(p^{P*}|_{\hat{x}P\leq c^A}, x^{P*}|_{\hat{x}P\leq c^A}) - w^D(p^{D*}|_{\hat{x}D\geq c^A}, x^{D*}|_{\hat{x}D\geq c^A})]_{c^O=k_{2.2}} \geq 0 \quad (A.35)
\]

When \( c^O = k_{2.1} \), there is \( \hat{x}D = c^A \) and \( \hat{x}P \leq c^A \). Thus, we can obtain

\[
[w^P(p^{P*}|_{\hat{x}P\leq c^A}, x^{P*}|_{\hat{x}P\leq c^A}) - w^D(p^{D*}|_{\hat{x}D\geq c^A}, x^{D*}|_{\hat{x}D\geq c^A})]_{c^O=k_{2.1}} = w^P(p^{P*}|_{\hat{x}P\leq c^A}, x^{P*}|_{\hat{x}P\leq c^A}) - w^D(p^{D*}|_{\hat{x}D=c^A}, x^{D*}|_{\hat{x}D=c^A})
\]

From Step 1.1, we have \( w^P(p^{P*}|_{\hat{x}P\leq c^A}, x^{P*}|_{\hat{x}P\leq c^A}) > w^D(p^{D*}|_{\hat{x}D=c^A}, x^{D*}|_{\hat{x}D=c^A}) \). Thus (A.34) is proved.
When \( c^A = \kappa_{2,2} \), there is \( \hat{x}^D \geq c^A \) and \( \hat{x}^P = c^A \). Thus, we can obtain

\[
\left[ w^P(p^P|\hat{x}^P \leq c^A, x^P|\hat{x}^P \leq c^A) - w^D(p^D|\hat{x}^D \geq c^A, x^D|\hat{x}^D \geq c^A) \right] \bigg|_{O = \kappa_{2,2}}
\]

\[
= w^P(p^P|\hat{x}^P = c^A, x^P|\hat{x}^P = c^A) - w^D(p^D|\hat{x}^D \geq c^A, x^D|\hat{x}^D \geq c^A)
\]

From Step 1.3, we have \( w^P(p^P|\hat{x}^P = c^A, x^P|\hat{x}^P = c^A) > w^D(p^D|\hat{x}^D \geq c^A, x^D|\hat{x}^D \geq c^A) \). Thus (A.35) is proved.

Hence, we have \( w^P(p^P, x^P) > w^D(p^D, x^D) \) when \( \hat{x}^D \geq c^A \) and \( \hat{x}^P \leq c^A \). Step 1.4 ends.

Summarizing the proofs in Steps 1.1-1.4, we can conclude that \( w^P(p^P, x^P) > w^D(p^D, x^D) \) always holds. Step 1 ends.

Step 2: Prove \( w^D(p^D, x^D) > w^S(p^S) \)

From Theorem 5.1, we can conclude that because all customers are quoted lower prices in DDP than in SP, it is obvious that \( w^D(p^D, x^D) > w^S(p^S) \).

Based on the proofs in Steps 1-2, we have \( w^P(p^P, x^P) > w^D(p^D, x^D) > w^S(p^S) \). Hence, Theorem 5.3 is proved.