Three Essays in Operations Management

by

Dror Z. Hermel

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

THE FACULTY OF GRADUATE STUDIES
(Business Administration)

The University Of British Columbia
(Vancouver)

September 2013

© Dror Z. Hermel, 2013
Abstract

This dissertation addresses three topics in the domain of operations management. First we study the problem of profit allocation in a supply chain using a bargaining approach. We present a novel framework for the analysis of this problem. The application of our framework results in a prescription for the required profit allocations. We prove that in a setting where all supply chain agents can communicate, possibly coordinating their actions, the allocation prescribed by our bargaining framework coincides with the Shapley value of a cooperative game associated with the setting. Next, we study revenue management in the presence of strategic consumers, who face some uncertainty regarding the product valuation. We show, contradictory to the main stream of the literature regarding strategic consumers, that under certain circumstances, the retailer may prefer facing strategic consumers rather than myopic ones. Finally, we study the issue of cross-dock operations management at a shift-level. We target the main gap identified in the literature for this issue, and present a holistic framework for the allocation of cross-dock resources to processing of containers and freight. We show, using simulated data that our approach outperforms current practices.
Preface

Chapter 2 is co-authored by Daniel Granot and Mahesh Nagarajan. Chapter 3 is co-authored by Benny Mantin and was edited based on helpful comments from Yossi Aviv. Chapter 4 is co-authored by Nicole Adler, Hamed Hasheminia, and Michael J. Fry. In all chapters I was the main contributor. I was responsible for developing the models, carrying out analysis and reporting the results, as presented in this dissertation. Chapters 3 and 4 benefited from insightful editing comments suggested by my advisors. All chapters will be reformatted and submitted for publication in academic peer reviewed journals.
Table of Contents

Abstract ........................................................................................................ ii
Preface ........................................................................................................... iii
Table of Contents ........................................................................................... iv
List of Tables .................................................................................................. vi
List of Figures ................................................................................................. vii
Acknowledgments .......................................................................................... viii
Dedication ....................................................................................................... ix

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

2 Bargaining For Supply Chain Profit Allocation .......................................... 4
  2.1 Introduction ............................................................................................. 4
  2.2 The Bargaining Model and Analysis ....................................................... 7
    2.2.1 Illustrative Example ......................................................................... 8
    2.2.2 Non-pivotal Agents Cannot Communicate ...................................... 8
    2.2.3 Non-pivotal Agents Can Communicate .......................................... 12
  2.3 Implications and Future Research ............................................................. 20
  2.4 Proof of Comment in Footnote 6 ............................................................... 22
List of Tables

Table 3.1  Valuation Transition Probability . . . . . . . . . . . . . . . . . . 29
Table 3.2  Surplus Comparison: Buying In Period 1 vs. Waiting For Pe-
           riod 2 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30
Table 3.3  The Seller’s Expected Revenue For the Second Period . . . . . 32
Table 3.4  Surplus Comparison: Buying In Period 1 vs. Buying In Period
           2 With Returns Policy - Refund \( r_f \) . . . . . . . . . . . . . . . . . . 46
Table 4.1  Comparison of QAP, cdQAP, and mdQAP . . . . . . . . . . . . . . . 66
List of Figures

Figure 3.1  Revenue For The Different Pricing Policies . . . . . . . . . . 40
Figure 3.2  Profit When Consumers Are All Myopic Or All Strategic . . 43

Figure 4.1  Flow of Information Between MRCDSP Phases . . . . . . . 49
Figure 4.2  Example of Container Clustering . . . . . . . . . . . . . . . . 54
Figure 4.3  Cluster Processing Time as a Function of Allocated Resources 68
Figure 4.4  Efficiency Gains - Cluster Makespans : MRCDSP Stage 3 vs.
            Random-Assignment . . . . . . . . . . . . . . . . . . . . . . . . . . . 69
Figure 4.5  Illustrative Example - MRCDSP Stage 4 for Different Cross-
            Dock Resource Levels . . . . . . . . . . . . . . . . . . . . . . . . . . . 71
Figure 4.6  Percent Improvement of the MRCDSP Stage 4 vs Random As-
            signment Per Mode . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 73
The dissertation presented herein is a by product of an ongoing process of my personal growth and evolvement as an academic researcher and as an individual. Throughout this ongoing endeavour, I have learned much about myself and about my ability to partake in research aimed at creating (possibly) useful and (hopefully) interesting knowledge.

I want to extend my gratitude to those who provided me with the opportunity to partake in this very rewarding and satisfying journey, and at times shared the experience with me, helping me pave my way. First and foremost, I wish to thank my advisors - Daniel Granot, Mahesh Nagarajan - who offered me their time and patience, and shared with me their vast knowledge and experience. I further thank my advisors - Frieda Granot, Daniel Granot, and Mahesh Nagarajan - for providing me with extensive funds during my PhD studies, a priceless contribution which enabled me to devote all of my time to my development as a future academic.

I want to thank my co-authors. Specifically, I want to thank Mike Fry for his timeless efforts helping revise Chapter 4 of this dissertation, and for agreeing to serve as a dissertation committee member. Finally I want to thank a number of friends. Benny Mantin, a co-author, mentor, and friend. Hamed Hasheminia, a friend who has shown me that patience and hard work really do payoff in the long run. And last, but definitely not least, I wish to thank Antoine Saure and Greg Werker, who were my second family during my time at UBC.

My final yet most important gratitude goes to my loving family.
Dedication

To Eitan Gilboa Hermel
Chapter 1

Introduction

This dissertation presents three essays, each contributing to the field of operations management by attempting to mitigate a gap identified in the academic literature. In each essay we introduce relevant existing concepts from economics and mathematics, adapted to our settings, and derive new results, insights, and methodologies, that enable a better understanding of these settings. We can classify the essays comprising this dissertation not only by the problem they target, but also by the level of planning addressed in each. The first essay focuses on strategic issues of supply chain formation, structure, and coordination. The second essay addresses the tactical issue of revenue management when consumers face valuation uncertainty. The third essay focuses on operational issues arising in a cross-dock facility when planning a shift level schedule of resources to tasks.

In the first essay we study a bargaining problem in a supply chain setting, where decentralized decision makers must decide how to cooperate. Specifically, we focus on a single pivotal agent and $N$ non-pivotal agents bargaining over the allocation of profits they generate by cooperation. Our approach to this problem using a bargaining framework should be contrasted with the existing practice in the operations management and economics literature to coordinate decentralized supply chains using a contract type mechanism. The specifics of the contract, however, are often left unaddressed, and it is natural to think of these contract details as agreed upon by the agents through negotiations. We propose an adaptation of the Nash bargaining framework to model negotiations. Namely, we present a novel
methodology for construction of the disagreement outcome using readily available data, endogenous to the problem, rather than using an exogenously determined disagreement outcome, which is often the case in many models.

We analyze the problem under two distinguishable scenarios which differ in the ability of the non-pivotal agents to communicate amongst themselves, possibly coordinating their actions. In the setting where the non-pivotal agents are not allowed to communicate among themselves, application of our framework leads to a prescribed supply chain structure and a profit allocation scheme given by a simple closed form formula. This allocation scheme constitutes a Nash-Nash equilibrium to the underlying problem, an inter-related system of bilateral Nash bargaining problems between the pivotal agent and each of the non-pivotal agents. In the complementary case, where non-pivotal agents can communicate, we prove that our framework leads to a Nash bargaining solution which coincides with the Shapley value of an associated cooperative game.

In the second essay we consider a profit maximizing seller who encounters consumers over two periods. The consumers face valuation uncertainty in the first period, which is resolved in the second period. We show that the seller may benefit from encountering strategic, rather than myopic consumers. While this result is not new, our novel model explains this phenomena using a simple setting of valuation change, rather than by more complex concepts such as patience level or a complex mathematical model (and end of horizon conditions within that model) as was previously done.

After establishing the optimal pricing policy, and the possible benefit from encountering strategic consumers, we extend our model to allow for product returns. We analyze the case where all consumers are strategic, focusing our attention on the benefits the seller may obtain offering a return option for products sold in the first period. We find that the seller could enhance revenues by offering consumers a return option. Moreover, we establish that the seller may benefit from providing a full, rather than a partial refund for the returns.

In the third essay we propose a framework for container scheduling and resource allocation decisions at a cross-dock facility. Our approach minimizes material flow and optimizes the scheduling of inbound and outbound containers subject to resource constraints at the cross-dock. While the issues of container scheduling
and resource allocation problems, at cross-dock facilities, have been studied previously in isolation, our work is the first to consider a complete view of cross-dock operations, providing optimal dock-door-to-container and resource allocations. We present a comprehensive framework that includes container clustering to reduce the problem size, a dock-door assignment algorithm, and container scheduling models that can be solved to optimality for practically-sized problems. We conclude by showing, using a comparative numerical study, that our approach outperforms current practice at a large cross-dock facility. Carrying out our approach on simulated data reduces the average time required for processing a set of containers by 37% and reduces the weighted-distance material traveled within the cross-dock by 45%.

Each of the essays is self contained and contains the relevant assumptions, literature review, and conclusions, which are not repeated in separate sections. The rest of the dissertation is organized so that each chapter includes one essay. Proofs not included in the body of the essays are included in a technical appendix.
Chapter 2

Bargaining For Supply Chain Profit Allocation

2.1 Introduction

Supply chains have been a source for research in many fields for the past few decades. Specifically, the issue of coordination among decentralized supply chain participants has received much attention in the economics, operations management, logistics, and industrial engineering literature. This is largely due to the fact that decentralized decision making often leads to inefficiencies with respect to coordinated decision making. One well studied example is double-marginalization (Spengler (1950)). Many of the studies on coordination take an industrial-organization approach (c.f. Bolton and Dewatripont (2005)), modelling interactions between agents as a non-cooperative game, where the result is a channel coordinating contract, for the setting at stake (e.g., Cachon and Lariviere (2005); Pasternack (1985); Bernstein and Federgruen (2005)). While this non-cooperative approach enables the achievement of significant insights in certain scenarios, it has a major drawback in the form of dependence of the outcome on specific modeling choices. Examples for this are: The sequence of events which is not always based on some sound logic and in some cases may even be random (e.g., Compte and Jehiel (2010); Okada (2011)); the action space available for the agents may be limited to a choice of model parameters, and hence lacks the ability to coordinate certain channel struc-
tures (e.g., Cachon and Lariviere (2005)).

An alternative approach, which eliminates some of the aforementioned limitations is to approach the problem of coordination using cooperative game theory frameworks. This approach assumes that a given cooperation structure generates a certain profit, and focuses on allocating it to the agents\(^1\) (c.f. Peleg and Sudholter (2007)). The underlying premise is that coalitions of agents cooperate in such a manner that they maximize their profits, and all values of an associated characteristic function, mapping coalitions to profit values, need to be accounted for when allocating the profit. The allocation schemes suggested are characterized by a different set of some (possibly) desirable properties. The Shapley value, for example, is the unique allocation that satisfies, simultaneously, the following attributes: Pareto optimality, symmetry, dummy players are allocated zero profit, and additivity. Further, it may be also thought of as an allocation where each agent receives profits that equal the average of all his marginal contributions to all coalitions\(^2\). Other cooperative solution concepts prevalently used in the operations management literature are: the Nucleolus, the Core Set, the von Neumann-Morgenstern Stable Set, and the Nash-Bargaining Solution (NBS).

In a supply chain setting, different agents have some incentive "to cooperate, but also have some conflicting interest with respect to how this cooperation should be carried out" (Muthoo (1999)), bargaining (or negotiating) appears to be a natural approach for the determination of the cooperation structure, and the resulting profit allocation. In this study we propose a bargaining framework for supply chain profit allocation using the foundations set-forth in the seminal papers by Nash (Nash (1950);Nash (1953)) regarding a prescription for an allocation resulting from bargaining over a profit achievable under cooperation, where no-cooperation leads to inefficiency, in the sense that the profit achievable under cooperation is greater than the profit possible under no-cooperation (the latter outcome is often referred to as the disagreement outcome).

Our work joins many recent papers in the operations management literature that incorporate bargaining into their methodology. Many of these papers revisit

\(^1\)We emphasize that this is significantly different from studying a revenue sharing type mechanism for coordination.

\(^2\)For further reading on the Shapley value we refer the readers to Roth (1988)
existing (and well studied) coordination mechanisms (or supply chain structures), and focus on the effect of incorporating bargaining on model parameters (as is the case for example in Wu et al. (2009), and Feng et al. (2012)). We, however, follow Nagarajan and Bassok (2008), Lovejoy (2010), and Aydin and Heese (2012), and provide an holistic approach, using the bargaining framework to carry out the analysis of the system to achieve, or predict, the cooperation structure of the supply chain that is the result the negotiations process, as well as the associated profit allocation.

Nagarajan and Bassok (2008), Lovejoy (2010), and Aydin and Heese (2012), study a two echelon supply chain with a pivotal agent, e.g., a monopolist retailer, and a set of $N$ non-pivotal agents, e.g., suppliers or manufacturers. In Nagarajan and Bassok (2008), the suppliers provide complementary products, in Lovejoy (2010), they deliver fully substitutable products, and in Aydin and Heese (2012), the suppliers provide partially substitutable products, in the context of an assortment problem). Our approach, based on the Nash Bargaining framework, provides a unified approach, and enables the study of all of these systems (as well as a wide range of other systems and settings).

Our framework prescribes a profit allocation scheme based on the expected result of a simultaneous negotiation process carried out by the agents in the system, all possessing common and complete information. The inputs required by our framework are, essentially, the same inputs required for a definition of a cooperative game - the set of agents (and all possible coalitions), and a "characteristic function" (mapping coalitions to values). In the supply chain context, the possible coalitions are, for example, the cooperation structures achievable by a single pivotal agent, e.g., retailer, and $N$ non-pivotal agents, e.g., suppliers or manufacturers. The different profits resulting from these cooperating structures can be used to define the values of the characteristic function. We turn to the Nash Bargaining problem in order to determine the profit allocation scheme to achieve our desired allocation.

3In this study we assume that this value is provided to us as exogenous data. However, we note that this may not necessarily be the case, and the profits may need to be calculated based on some given demand function. The profit is then calculated assuming a unique decision maker for each coalition - equivalent to full cooperation for each coalition.

4We mention that even though we analyze and discuss a supply chain structure consisting of
We will assume, as we justify in section 4.3, that bargaining is carried out by all agents that are members of the coalition generating the maximum achievable profit, and that these agents bargain over this profit. The disagreement outcome, for this bargaining setting, is not necessarily straightforward to identify. Surprisingly, despite the impact of the disagreement outcome on the resulting allocation (Binmore et al. (1989)), it is a common assumption in the bargaining literature that these outcomes are given exogenously. In order to appropriately account for the disagreement outcome, our suggested framework uses the information available to construct these independently. Our approach to construct the disagreement outcome is based on the assumption that in case bargaining with all non-pivotal agents, $N$, breaks down, the pivotal agent attempts to establish cooperation with the sub-coalition $S^1$, $S^1 \subseteq N$, yielding her the highest possible payoff. If in turn, bargaining with $S^1$ breaks down, the pivotal agent attempts to achieve cooperation with a new sub-coalition, $S^2$, $S^2 \subseteq S^1$, etc., until reaching such a sub-coalition for which the consequences of failure to reach an agreement is clear to all parties.

We present our framework under two complementary scenarios. In one, we limit the ability of non-pivotal agents to communicate, as may be the case if we account for anti-trust law. We achieve a closed form formula for the resulting allocation scheme. This allocation possesses certain lucrative properties: By construction, it is an interior point of the core, and adheres to the axioms introduced by Nash (Nash (1950);Nash (1953)) for the NBS; further, it constitutes a Nash-Nash solution concept to the assortment problem presented in Aydin and Heese (2012). In the complementary setting, the non-pivotal agents are allowed to communicate, and the allocation scheme achieved through our bargaining framework coincides with the Shapley value of the corresponding cooperative game.

### 2.2 The Bargaining Model and Analysis

We will first consider a simple example to demonstrate some of the challenges we face in the bargaining problem. Subsequently, we formally introduce and analyze the general supply chain bargaining problem.
2.2.1 Illustrative Example

Consider the following three person bargaining situation: a pivotal agent, denoted by $R$, designating a retailer, and two other agents, denoted 1 and 2, designating, respectively, Manufacturer 1 ($M_1$) and Manufacturer 2 ($M_2$). The agents can cooperate and achieve profits as follows. If all three agents cooperate, they can secure a profit of 9 (denoted by $v(R,1,2)$). If only $R$ and $M_1$ cooperate they achieve a profit of 7 (denoted $v(R,1)$), and if only $R$ and $M_2$ cooperate, they can get a profit of 5 (denoted $v(R,2)$). No other options for cooperation yielding positive profits are available for the agents.$^5$

Given the profits that $R$, $M_1$, and $M_2$ can achieve from cooperation, we would like to address the question of how the three agents will divide between them the profit, 9, which they can achieve if all three decide to cooperate. Presumably, they will resort to some sort of bargaining to determine their share of the profits. Accordingly, we will employ the Nash Bargaining (NB) framework to model the bargaining process among the agents, and thus ensure that the profit allocations we derive possess certain desirable properties.

To apply the NB framework, we need to identify the set, $S$, on which the parties bargain, which is the profit of 9 if all three decide to cooperate, as well as the disagreement outcome if the parties fail to reach an agreement. Usually, in the NB framework the disagreement outcome is exogenously given, reflecting outside opportunities available to the agents. The main challenge that we face is the absence of an externally given disagreement outcome, and we overcome this challenge by explicitly accounting for the players’ outside option if bargaining breaks down.

Apparently, the analysis depends crucially whether the non-pivotal agents are, or are not, able to communicate. Accordingly we will separately consider these two cases.

2.2.2 Non-pivotal Agents Cannot Communicate

Communication and coordination between the non-pivotal agents are not possible if their relations are governed by anti-trust laws such as the Sherman Antitrust Act.

$^5$Note that we implicitly assume monotonicity. This assumption is required only for the setting where the non-pivotal agents can communicate, where it is discussed explicitly. When comparing the two setting this assumption is also imposed on the non-communicating setting.
of 1890, the Clayton Act of 1914, the Celler-Kefauver Act of 1950, etc.

Let us consider first the illustrative example. If bargaining over the profit allocation of \( v(R, 1, 2) = 9 \) collapses, \( R \) can either bargain with \( M_1 \) over \( v(R, 1) = 7 \), or she can bargain with \( M_2 \) over \( v(R, 2) = 5 \), but naturally, she cannot bargain again with both \( M_1 \) and \( M_2 \), since it is assumed that this bargaining has collapsed. The bargaining is carried out by \( R \), who negotiates with \( M_1 \) or \( M_2 \), separately, regarding her share of \( v(R, 1) \) and \( v(R, 2) \), if \( M_1 \) or \( M_2 \), respectively, is chosen by \( R \) as her bargaining partner. Now, \( R \) can leverage the fact that \( M_1 \) and \( M_2 \) cannot communicate and possibly coordinate their actions, to induce \( M_1 \) and \( M_2 \) to offer her increasingly larger shares of \( v(R, 1) \) and \( v(R, 2) \), respectively, just to be chosen by \( R \) as her sole bargaining partner. At the limit, this would result with \( M_2 \) offering \( R \) the entire value of \( v(R, 2) = 5 \) just to be selected by \( R \) as her only bargaining partner. Therefore, the inability of the non-pivotal players to communicate implies that \( M_2 \) has relinquished completely his share of \( v(R, 2) = 5 \) to \( R \), at the stage where \( R \) is carrying out bilateral bargaining with \( M_1 \) and \( M_2 \).

For each non-pivotal agent \( M_i \), we will refer to \( R \)'s outside option with respect to \( M_i \), as the best \( R \) and the rest of the non-pivotal players can do at the absence of \( M_i \). Thus, \( R \)'s outside option with respect to \( M_1 \) is \( v(R, 2) \) and her outside option with respect to \( M_2 \) is \( v(R, 1) \).

Let us turn back now to the bargaining problem between \( R, M_1 \) and \( M_2 \), over \( v(R, 1, 2) \). In this problem, \( R \) can claim that the bargaining with each non-pivotal agent, \( M_i \), should exclusively be on the marginal profit that can be achieved with the help of \( M_i \), above \( R \)'s outside option with respect to \( M_i \). That is, bargaining with \( M_1 \) should be only on \( (v(R, 1, 2) - v(R, 2)) \), bargaining with \( M_2 \) should only be on \( (v(R, 1, 2) - v(R, 1)) \), and, in view of the outcome in bilateral bargaining stage, \( R \) has an exclusive claim over \( v(R, 2) \). Employing Nash’s symmetry principal, beyond the disagreement outcome, we conclude that \( M_1 \)'s allocation should be exactly equal to \( R \)'s allocation from bargaining over \( (v(R, 1, 2) - v(R, 2)) \), and \( M_2 \)'s allocation should be exactly equal to \( R \)'s allocation from bargaining over \( (v(R, 1, 2) - v(R, 1)) \), and \( R \) exclusively claims \( v(R, 2) \), and we derive the following allocation of \( v(R, 1, 2) \):
Let us extend the analysis, and suppose \( \{M_1, M_2, M_3\} \) is a set of three manufacturers. \( R \) can cooperate with any subset of them yielding profits as follows:

\[
v(R, 1, 2, 3) \geq v(R, 1, 2) \geq v(R, 1, 3) \geq v(R, 2, 3) \geq v(R, 1) \geq v(R, 2) \geq v(R, 3).
\]

All other coalitions generate zero profits. We would like to use the NB framework to determine the allocation of \( v(R, 1, 2, 3) \) among all agents. We begin by identifying the outside options that \( R \) has when she bargains with each non-pivotal agent, which are as follows: When bargaining with \( M_1 \), \( R \)'s outside option is \( v(R, 2, 3) \); when bargaining with \( M_2 \), \( R \)'s outside option is \( v(R, 1, 3) \); and, when bargaining with \( M_3 \), \( R \)'s outside option is \( v(R, 1, 2) \). Then, as we have done in the two non-pivotal agents example, taking into account \( R \)'s outside options and employing Nash’s symmetry axiom we conclude as follows: Over the marginal profit \( (v(R, 1, 2, 3) - v(R, 2, 3)) \), \( R \)'s and \( M_1 \)'s shares should be equal; over \( (v(R, 1, 2, 3) - v(R, 1, 3)) \), \( R \)'s and \( M_2 \)'s shares should be equal; over \( (v(R, 1, 2, 3) - v(R, 1, 2)) \), \( R \)'s and \( M_3 \)'s shares should be equal, and \( R \) exclusively claims \( v(R, 2, 3) \).

The resulting outcome is

\[
X_R = v(R, 2, 3) + \frac{(v(R, 1, 3) - v(R, 2, 3))}{2} + \frac{(v(R, 1, 2) - v(R, 1, 3))}{3} + \frac{(v(R, 1, 2, 3) - v(R, 1, 2))}{4} = \frac{2}{3}
\]

\[
X_1 = \frac{(v(R, 1, 3) - v(R, 2, 3))}{2} + \frac{(v(R, 1, 2) - v(R, 1, 3))}{3} + \frac{(v(R, 1, 2, 3) - v(R, 1, 2))}{4} = \frac{2}{3}
\]

\[
X_2 = \frac{(v(R, 1, 2) - v(R, 1, 3))}{3} + \frac{(v(R, 1, 2, 3) - v(R, 1, 2))}{4} = \frac{2}{3}
\]

\[
X_3 = \frac{(v(R, 1, 2, 3) - v(R, 1, 2))}{4}.
\]
The analysis can be easily extended to the general case. Specifically, the various coalitions are ordered in decreasing order of the profits they generate, \( v(R, T^*_1) \geq v(R, T^*_2) \geq v(R, T^*_3) \geq \ldots \), where it is assumed that \( T^*_1 \) is the set, \( N \), of all manufacturers. Let \( T^*_k \) denote the coalition with the highest coalitional value, or, equivalently the smallest index, such that for each \( i \in N \), there exists a \( j, j \leq k \), such that \( M_i \notin T^*_j \). Thus, \( k \) is the smallest index such that \( R \) has an outside option, \( T, T \in \{ T^*_1, \ldots, T^*_k \} \), with respect to any non pivotal agent.

As in the example previously considered, \( R \) bargains with each agent only over the profit margin above the outside option that \( R \) has with respect to this agent. To derive the allocation vector of \( v(R, T^*_1) \) in the general case, let \( Q_j \) denote the set of non-pivotal agents entitled to a share of the margin \( v(R, T^*_j) - v(R, T^*_{j+1}) \), \( j = 1, \ldots, k - 1 \), among all entitled non-pivotal agents, \( Q_j \), and \( R \), then each agent \( i \in Q_j \) is allocated:

\[
\pi_j = \frac{v\left(T^*_j\right) - v\left(T^*_{j+1}\right)}{|Q_j| + 1}, \quad j = 1, \ldots, k - 1.
\]

Accordingly, we derive a closed form expression for the allocation vector of the profit, \( v(R, T^*_1) \), referred to as the bargaining solution and denoted \( X^{BS} \):

\[
X^{BS}_R = v(R, T^*_k) + \sum_j \pi_j, \quad \text{and} \quad X^{BS}_i = \sum_{j \in Q_j} \pi_j, i \in N. \quad (2.4)
\]

The bargaining solution \( X^{BS} \), given by (2.4), coincides with the bargaining solution derived independently by Aydin and Heese (2012) using a completely different approach. In their setting, the pivotal agent carries out bi-lateral bargaining with each of the non-pivotal agents regarding cooperation and profit allocation. The bargaining problems, each carried out between the pivotal agent and a single non-
pivotal agent, are modelled as NB problems. These problems are inter-related by the disagreement outcomes that the pivotal agent utilizes in each of the problems. Specifically, the pivotal agent’s disagreement outcome in the bargaining problem with each agent $i$ is her allocation in the bargaining problem between her and all other agents except agent $i$. In that respect, Aydin and Heese’s approach is similar to our outside option approach. Aydin and Heese’s solution, where an inter-related system of bargaining problems is solved in equilibrium, is referred to in the literature as a Nash-Nash equilibrium.

**Proposition 1** $X^{BS}$, given by (2.4), is a Nash-Nash equilibrium to the non-cooperative game wherein $R$ simultaneously negotiate bilaterally with all the non-pivotal agents about the allocation of $v(R, T^*_1)$.

The proof of the proposition follows immediately from the coincidence of $X^{BS}$ with the profit allocation proposed in Theorem 1 of Aydin and Heese (2012), which can be verified by substituting $\beta_k = 1$ for all $k$, in their formula (9).

### 2.2.3 Non-pivotal Agents Can Communicate

Notwithstanding anti-trust laws, communication and coordination between the non-pivotal agents could be possible, for example, if the products they offer are relatively complementary.

Consider again the illustrative example, where $R$ bargains with $M_1$ and $M_2$ over the allocation of $v(R, 1, 2)$. As in the “no-communication” case, if the bargaining over $v(R, 1, 2)$ collapses, $R$ can still offer either $M_1$ or $M_2$ to bargain with her over the respective profits the parties can achieve. However, by contrast with the “no communication” setting, $M_1$ and $M_2$ are able to communicate in this case and could possibly coordinate their actions in order to extract some of the profit $R$ was able to claim in the "no communication” case. Specifically, $R$ may be unable to induce the non-pivotal players, $M_1$ and $M_2$, to offer her successively increasing shares of $v(R, 1)$ and $v(R, 2)$ in order to be $R$’s chosen bargaining partner.

In fact, in the "communication case", we will assume that in the bilateral bargaining stage between $R$, $M_1$ and $M_2$, $R$ is able to secure only $\frac{v(R, 1)}{2}$ and $\frac{v(R, 2)}{2}$, in the bargaining problems over $v(R, 1)$ and $v(R, 2)$, respectively.
Let us turn now to the bargaining problem over \( v(R, 1, 2) \). Our approach in the "communication case" in this stage is a bit different than in the "no communication case" presented above. Specifically, rather than deriving directly the allocation of \( v(R, 1, 2) \), as it was done in the "no communication case", we will derive instead the disagreement outcome vector for the bargaining problem \( v(R, 1, 2) \).

Now, as explained above, \( R \) is able to secure either \( v(R, 1) \) or \( v(R, 2) \), if she bargains with either \( M_1 \) or \( M_2 \), respectively. Thus, we assume that \( R \) can secure \( v(R, 1) \) if bargaining over \( v(R, 1, 2) \) breaks down. Following Nash’s symmetry principle, \( M_1 \) can claim only a share of \( v(R, 1) \) which is equal to \( R \)'s share of \( v(R, 1) \) beyond the profit, \( \frac{v(R, 2)}{2} \), that \( R \) can secure at the absence of \( M_1 \). That is, \( M_1 \) can secure a claim of \( \frac{v(R, 1) - v(R, 2)}{2} \) of \( v(R, 1) \). Since \( R \)'s secured profit at the absence of \( M_2 \) is \( \frac{v(R, 1)}{2} \), \( M_2 \) cannot claim any share of \( v(R, 1) \).

We conclude that if bargaining over \( v(R, 1, 2) \) breaks down in the "communication case", the parties can claim the following allocations: \( R \) gets a payoff of \( 3 \frac{1}{2} \), \( M_1 \) gets a payoff of 1, and \( M_2 \) gets zero. We will refer to the vector \((3 \frac{1}{2}, 1, 0)\) as the disagreement outcome for the bargaining problem over \( v(R, 1, 2) \), using the disagreement outcome \((3 \frac{1}{2}, 1, 0)\) for the NB problem over \( v(R, 1, 2) \), we obtain the following profit shares for the players:

\[
X_R = \frac{v(R, 1, 2) - (\frac{v(R, 1)}{2} + \frac{v(R, 1) - v(R, 2)}{2}) + \frac{v(R, 1)}{2}}{3} = 5
\]

\[
X_{M_1} = \frac{v(R, 1, 2) - (\frac{v(R, 1)}{2} + \frac{v(R, 1) - v(R, 2)}{2}) + \frac{v(R, 1) - v(R, 2)}{2}}{3} = 2 \frac{1}{2}
\]

\[
X_{M_2} = \frac{v(R, 1, 2) - (\frac{v(R, 1)}{2} + \frac{v(R, 1) - v(R, 2)}{2})}{3} = \frac{1}{2}
\]

Note that, as expected, \( R \)'s share of the profit in the "communication case" is lower than her share in the "no communication case", and, by contrast, \( M_1 \)'s and \( M_2 \)'s shares are larger in the "communication case". Further, note that this profit allocation coincides with the Shapley value of the cooperative game associated with the NB problem.

\textsuperscript{6}In fact, this statement remains true for the setting where \( R \) faces \( N \) non-pivotal agents when we assume the payoffs are such that the characteristic function for the associated cooperative game is monotonic in the coalition size. The proof of this statement is provided below. Note that it is a direct consequence of comparing \( R \)'s payoff for both cases as provided in (2.4) and (2.19).
We later prove the NB solution for responding to $R$'s corresponding outside option. We conclude therefore that the disagreement outcome in case bargaining over $v(R, 1, 2, 3)$ breaks down, each of these agents can only claim their egalitarian share from their marginal profit contribution above $R$'s corresponding outside option. We conclude therefore that the disagreement outcome vector $d(v(R, 1, 2, 3)) = (d_R(v(R, 1, 2, 3)), d_i(v(R, 1, 2, 3)), i = 1, 2, 3)$, corresponding to $R, M_1, M_2, M_3$, in the NB problem over $v(R, 1, 2, 3)$, is as follows:

$$
\begin{align*}
d_R(v(R, 1, 2, 3)) &= \phi_R(1, 2) \\
d_1(v(R, 1, 2, 3)) &= \phi_R(1, 2) - \phi_R(2, 3) \\
d_2(v(R, 1, 2, 3)) &= \phi_R(1, 2) - \phi_R(1, 3) \\
d_3(v(R, 1, 2, 3)) &= \phi_R(1, 2) - \phi_R(1, 2) = 0.
\end{align*}
$$  

(2.6)

We later prove the NB solution for $v(R, 1, 2, 3)$ with the disagreement outcome vector $d(v(R, 1, 2, 3))$ coincides with the Shapley value for the associated cooperative game.

We can now turn to extending our analysis to the case of a set $N$, of non-pivotal
agents, \( N = \{ M_1, M_2, \ldots, M_n \} \). We will assume that the highest profit is achieved by cooperation of the pivotal agent, \( R \), with the entire set of non-pivotal agents, and that \( v(R, T) \) is monotonic in \( T \). That is \( v(R, N) \geq v(R, T) \geq v(R, S) \) for \( S \subseteq T \subseteq N \). We suggest that the monotonicity assumption is reasonable since the non-pivotal agents are allowed to communicate and possibly coordinate their actions (e.g., if \( v(R, T) < v(R, S) \) and \( S \subset T \) then all agents in \( T \) can coordinate their actions and achieve \( v(R, S) \)). Recall that we refer to \((N \cup \{R\}, v(R, \cdot))\), with \( v(R, S), S \subseteq N \), as the cooperative game associated with the NB problem.

All agents bargain on their share of \( v(R, N) \). As before, we adopt the NB framework to model the bargaining process over \( v(R, N) \). The disagreement outcome, denoted \( d(v(R, N)) \), reflects the outside options of the players, stemming from their abilities to form coalitions.

We aim to construct a recursive formula for \( d(v(R, N)) \). Let \( NBS(v(R, T), d(v(R, T))) \) denote the NBS over \( v(R, T) \) with a disagreement outcome \( d(v(R, T)) \). Then, recognizing \( R \)'s ability to select the cooperation structure in case bargaining breaks down, we have that \( R \)'s disagreement outcome, \( d_R(v(R, N)) \), if bargaining over \( v(R, N) \) breaks down is given by:

\[
d_R(v(R, N)) \equiv \max_{S \subseteq N, |S| = |N| - 1} \{ NBS_R(v(R, S), d(v(R, S))) \}, \tag{2.7}
\]

where \( NBS_R \) is \( R \)'s allocation in the corresponding NB problem.

As in the example previously considered, the disagreement outcome for any non-pivotal agent, \( u \), denoted \( d_u(v(R, N)) \), is defined to be equal to \( R \)'s marginal benefit from the inclusion of \( u \) in the cooperation structure, which, as was in the non-communicating case, can also be thought of as agent's \( u \) egalitarian share of the profit generated above \( R \)'s disagreement outcome when not cooperating with \( u \). That is,

\[
d_u(v(R, N)) = d_R(v(R, N)) - NBS_R\{v(R, N/\{u\}), d(v(R, N/\{u\}))\}. \tag{2.8}
\]

A recursive formula for \( R \)'s disagreement outcome corresponding to \( R \) and any
given set of non-pivotal agents $T \subseteq N$, $|T| > 2$, is as follows:

$$d_R(v(R, T)) = \max_{S \subseteq T, |S| = |T| - 1} NBS_R\{v(R, S), d(v(R, S))\}, \quad (2.9)$$

and for any non-pivotal agent $u$:

$$d_u(v(R, T)) = d_R(v(R, T)) - NBS_R\{v(R, T/\{u\}), d(v(R, T/\{u\}))\}. \quad (2.10)$$

As a base case for the recursion we have the case of a NB problem between $R$ and any of the non-pivotal agents. Specifically, when bargaining with any non-pivotal agent $u \in N$,

$$d_R(v(R, \{u\})) = \frac{v(R, \{u\})}{2}. \quad (2.11)$$

This enables us to recursively solve for $d_R(v(R, S))$ for any coalition of non-pivotal agents $S$, and consequently, for the disagreement outcome $d_u(v(R, S))$ for all $u \in S$.

We define a cooperative game $(N \cup \{R\}, v)$ where $v$ is the "characteristic function" as described above to be the cooperative game associated with our Nash Bargaining problem, and are able to now state the following theorem.

**Theorem 1** The NBS over $v(R, N)$ with a disagreement outcome, $d(v(R, N))$, endogenously generated via (2.7)-(2.11), coincides with the Shapley value of the cooperative game $(N \cup \{R\}, v)$ associated with the Nash Bargaining problem.

Before presenting the proof we introduce the following notation. For any set of non-pivotal agents $T$, we will refer to $(T \cup \{R\}, v)$ as the cooperative game associated with the coalition $\{T \cup \{R\}\}$. Further, we denote by $NBS_i\{v(R, T), d(v(R, t))\}$ and $Sh_i(T \cup \{R\}, v)$ the allocation to agent $i \in \{T \cup \{R\}\}$ from the NB problem $\{v(R, T), d(v(R, T))\}$ and from the Shapley value for the cooperative game $(T \cup \{R\}, v)$, respectively.

The proof is by induction on the number of non-pivotal agents. When $R$ bargains with a single agent, the pivotal and non-pivotal agents share the profit equally, both according to the NB solution and the Shapley value. For clarity of exposition, let us also provide the proof for the two non-pivotal agent setting, considered in our illustrative example, where $v(R, 1, 2) > v(R, 1) > v(R, 2)$. As previously shown, the disagreement outcome vector in this case is $d(v(R, 1, 2)) = (\frac{v(R, 1)}{2}, \frac{v(R, 1) - v(R, 2)}{2}, 0)$. 16
Now the Nash Bargaining solution, $NBS\{v(R,1,2), (\frac{v(R,1)}{2}, \frac{v(R,1)-v(R,2)}{2}, 0)\}$, must satisfy symmetry. That is,

$$NBS_R\{v(R,1,2), d(v(R,1,2))\} - d_R(v(R,1,2)) =$$

$$NBS_1\{v(R,1,2), d(v(R,1,2))\} - d_1(v(R,1,2))$$

$$NBS_R\{v(R,1,2), d(v(R,1,2))\} - d_R(v(R,1,2)) =$$

$$NBS_2\{v(R,1,2), d(v(R,1,2))\} - d_2(v(R,1,2))$$

(2.12)

$$NBS_1\{v(R,1,2), d(v(R,1,2))\} - d_1(v(R,1,2)) =$$

$$NBS_2\{v(R,1,2), d(v(R,1,2))\} - d_2(v(R,1,2)).$$

As well as Pareto Optimality:

$$\sum_{i \in \{R,1,2\}} NBS_i\{v(R,1,2), d(v(R,1,2))\} = v(R,1,2).$$

(2.13)

Solving (2.12) and (2.13) yields:

$$NBS_R\{v(R,1,2), d(v(R,1,2))\} = \frac{2v(R,1,2) + v(R,1) + v(R,2)}{6}$$

$$NBS_1\{v(R,1,2), d(v(R,1,2))\} = \frac{2v(R,1,2) + v(R,1) - 2v(R,2)}{6}$$

(2.14)

$$NBS_2\{v(R,1,2), d(v(R,1,2))\} = \frac{2v(R,1,2) + v(R,2) - 2v(R,1)}{6}.$$  

Then, it can be easily verified, using the explicit expression for the Shapley value, that $Sh(\{1,2\} \cup \{R\}, v) = NBS\{v(R,1,2), (\frac{v(R,1)}{2}, \frac{v(R,1)-v(R,2)}{2}, 0)\}$, given by (2.14).

We now proceed to state our induction hypothesis. For any bargaining setting where a pivotal agent, $R$, bargains with a set, $S$, of $k - 1$ communicating non-pivotal agents over $v(R,S)$, the allocation prescribed by our framework, as described by equations (2.7)-(2.11), coincides with $Sh(S \cup \{R\}, v)$. We need to prove that, for any bargaining setting where the pivotal agent, $R$, bargains with a set, $T$, of $k$ communicating non-pivotal agents over $v(R,T)$, the allocation prescribed by our framework coincides with the Shapley value, $Sh(T \cup \{R\}, v)$.  

17
Let $T$ be an arbitrary set of $k$ non-pivotal agents. Then, as given by (2.9):

$$d_R(v(R, T)) = \max_{j \in T} \{NBS_R\{v(R, T / \{j\}), d(v(R, (T / \{j\})))\}\}.$$  

By the induction hypothesis, since $|T/\{j\}| = k - 1$,

$$d_R(v(R, T)) = Sh_R((T / \{j^*\}) \cup \{R\}, v)$$

$$\equiv \max_{j \in T} \{Sh_R((T/Mj) \cup \{R\}, v)\}, \text{for some } j^* \in T.$$  

(2.15)

By (2.10), for any non-pivotal agent, $j \in T$,

$$d_j(v(R, T)) = d_R(v(R, T)) - NBS_R\{v(R, T / \{j\}), d(v(R, T / \{j\})))\}.$$  

By our induction step we have

$$NBS_R\{v(R, T / \{j\}), d(v(R, T / \{j\})))\} = Sh_R((T / \{j\}) \cup \{R\}, v)$$

and from (2.15), $d_R(v(R, T)) = Sh_R((T / \{j^*\}) \cup \{R\}, v)$. We therefore have that for any $j \in T$:

$$d_j(v(R, T)) = d_R(v(R, T)) - Sh_R((T / \{j\}) \cup \{R\}, v)$$

$$\equiv Sh_R((T / \{j^*\}) \cup \{R\}, v) - Sh_R((T / \{j\}) \cup \{R\}, v).$$  

(2.16)

The symmetry of the NBS over $v(R, T)$ implies that for any two agents $i, j \in \{T \cup \{R\}\}, i \neq j$:

$$NBS_i\{v(R, T), d(v(R, T)))\} - d_i(v(R, T)) = NBS_j\{v(R, T), d(v(R, T)))\} - d_j(v(R, T))).$$

In particular, for $R$ and $j \in T$,

$$NBS_j\{v(R, T), d(v(R, T)))\} - d_j(v(R, T)) =$$

$$NBS_R\{v(R, T), d(v(R, T)))\} - d_R(v(R, T)).$$  

(2.17)
From Pareto Optimality of the NBS:

\[
\sum_{j=1}^{|T|} [NBS_j(v(R,T),d(v(R,T)))] + NBS_R(v(R,T),d(v(R,T))) = v(R,T),
\] (2.18)

and by (2.16), 

\[-d_{R}(v(R,T)) + d_{j}(v(R,T)) = -Sh_{R}((T\setminus\{j\}) \cup \{R\},v), \quad \forall j \in T.\]

We therefore have:

\[
\sum_{j=1}^{|T|} [NBS_j(v(R,T),d(v(R,T)))] + NBS_R(v(R,T),d(v(R,T))) =
\]

\[
\sum_{j=1}^{|T|} [NBS_R(v(R,T),d(v(R,T)))] - Sh_{R}((T\setminus\{j\}) \cup \{R\},v) + NBS_R(v(R,T),d(v(R,T))),
\]

which implies that:

\[
(|T| + 1) \cdot NBS_R(v(R,T),d(v(R,T))) - \sum_{j=1}^{|T|} Sh_{R}((T\setminus\{j\}) \cup \{R\},v) = v(R,T),
\]

which we can re-write as:

\[
NBS_R(v(R,T),d(v(R,T))) = \frac{v(R,T) + \sum_{j=1}^{|T|} Sh_{R}((T\setminus\{j\}) \cup \{R\},v)}{|T| + 1}. (2.19)
\]

By Hart and Mas-Colell (1989, Section 2), or Peleg and Sudholter (2007, Section 8.4):\(^7\)

\[
\frac{1}{|T| + 1} v(R,T) + \frac{1}{|T| + 1} \sum_{j=1}^{|T|} Sh_{R}((T\setminus\{j\}) \cup \{R\},v) = Sh_{R}(T \cup \{R\},v). \quad (2.20)
\]

Therefore, by (2.19) and (2.20) we conclude that our allocation for the pivotal agent, R, derived via (2.7)-(2.11), coincides with \(Sh_{R}(T \cup \{R\},v)\).

Next, consider any other agent, \(j \in T\), and recall that the Shapley value satisfies

\(^7\)This formulation is a result of the \textit{potential function} as defined in Hart and Mas-Colell (1989, Section 2). The allocation resulting from this \textit{potential function} guarantees efficiency as it calls for an equal allocation of profits beyond marginal contributions.
the balanced contributions property (Myerson, 1980). That is, for any \( j \in T \):

\[
Sh_R(T \cup \{R\}, v) - Sh_R((T \setminus j) \cup \{R\}, v) = Sh_j(T \cup \{R\}, v) - Sh_j((T \cup \{R\}) \setminus \{R\}, v),
\]

which yields - since \( \forall j \in T \) we know \( Sh_j((T \cup \{R\}) \setminus \{R\}, v) = 0 \) - that:

\[
Sh_R(T \cup \{R\}, v) - Sh_R((T \setminus \{j\}) \cup \{R\}, v) = Sh_j(T \cup \{R\}, v).
\]

Notice that by our construction, the allocations to the agents, resulting from \( NBS\{v(R, T), d(v(R, T))\} \), satisfy:

\[
NBS_j\{v(R, T), d(v(R, T))\} = NBS_R\{v(R, T), d(v(R, T))\} - d_R(v(R, T)) + d_j(v(R, T)),
\]

and recall that \(-d_R(v(R, T)) + d_j(v(R, T)) = -Sh_R((T \setminus \{j\}) \cup \{R\}, v)\). Therefore:

\[
NBS_j\{v(R, T), d(v(R, T))\} = NBS_R\{v(R, T), d(v(R, T))\} - Sh_R((T \setminus \{j\}) \cup \{R\}, v),
\]

but since \( NBS_R\{v(R, T), d(v(R, T))\} = Sh_R(T \cup \{R\}, v) \), we have that:

\[
NBS_j\{v(R, T), d(v(R, T))\} = Sh_R(T \cup \{R\}, v) - Sh_R((T \setminus \{j\}) \cup \{R\}, v),
\]

which implies that: \( NBS_j\{v(R, T), d(v(R, T))\} = Sh_j(T \cup \{R\}, v) \), and the proof is complete.

### 2.3 Implications and Future Research

Motivated by the problem of revenue sharing in a supply chain setting, we propose a novel bargaining framework for the allocation of supply chain profits. We focus on bargaining problems where the disagreement outcome is not determined exogenously, and must be inferred, and constructed, based on data endogenous to the problem. Focusing attention to this issue results in allocations that have desirable properties in excess of those guaranteed by the NBS, which serves as a foundation for our framework.

In the setting where the non-pivotal agents can not communicate, the resulting allocation is guaranteed, by the manner of its construction, to be an internal point
of the core set. This implies a notion of stability for our solution, as well as a notion of fairness, since proper coalitions are allocated more than what they can achieve on their own. In the setting where the non-pivotal agents can communicate, the resulting allocation is guaranteed to also satisfy the balanced contributions property, due to its equality to the Shapley value, which also constitutes a notion of fairness regarding the allocation.

We note that both communication structures exist in practice, and a possible ramification of our analysis is that there are some possible settings, where introducing limitations on the ability of certain agents to communicate, and possibly coordinate their actions, leads to drastic reduction in the equity of allocation of profits in this setting, something which may not necessarily coincide with the original motivation for placing these limitations.

A further implication of the allocations prescribed by our framework, relevant to supply chain coordination, is the ability to gain insight regarding the ability to implement a mechanism to coordinate a channel via a contract. Specifically, let us assume, as is done in contract theory, that we can translate (or map) a given contract into resulting profits, as a function of model parameters. In that case, we can prescribe a profit allocation that would result in cooperation, and consequently choose contract parameters that will result in these allocations. This would provide us with a specific implementable contract to coordinate the channel.

This work fills gaps in the existing supply chain management literature and bargaining literature. The concept of a bargaining problem in which the disagreement outcome needs to be constructed independently, based on data endogenous to the problem, as introduced in our framework, provides the ability to relieve the dependence on exogenous disagreement outcomes in certain settings. Specifically, in the supply chain setting, it also enables a unified approach for the analysis of a wide variety of supply chain structures, thus far studied independently or un-addressed. These contributions still have much room for further development. Specifically, we identify two promising avenues of future research. One is to extend this framework into a setting where there are no pivotal agents - for example, two competing retailers in the supply chain setting. This would allow analysis currently un-addressable by our framework. The second, is the incorporation of uncertainty regarding profits generated by the different coalitions, for example when demand is stochastic.
2.4 Proof of Comment in Footnote 6

We wish to prove that, for a given bargaining problem, where \( N \) non-pivotal agents bargaining with a pivotal agent, denoted \( R \), regarding the allocation of \( v(r,T_1^*) \), the following holds:

\[
X^\text{BS}_R(v(R,T_1^*)) \geq Sh_R(T_1^* \cup R,v).
\]

We assume that the function \( v \) is monotone (as assumed for our communicating setting). Further we assume the notation is such that \( v(r,T_{j+1}^*) = v(r,T_1^*/\{j\}) \), \( \forall j \in 1,2,\ldots,N \).

We prove by induction regarding the number of non-pivotal agents.

For the base case, where \( |N| = 2 \), we can compare equations (2.2) and (2.6) and achieve that:

\[
X^\text{BS}_R(v(R,T_1^*)) \geq Sh_R(T_1^* \cup R,v).
\]

Our induction hypothesis is that this holds for any case where there are \( |N| = k \) non-pivotal agents, i.e.

\[
X^\text{BS}_R(v(R,T_j+1^*)) \geq Sh_R(T_{j+1}^* \cup R,v) \quad \forall j = 1,2,\ldots,N.
\]

We will now turn to prove for the case that \( |N| = k + 1 \). We begin by noting that:

\[
X^\text{BS}_R(v(R,T_1^*)) = \frac{v(R,T_1^*) - v(R,T_2^*)}{|k| + 1} + X^\text{BS}_R(v(R,T_2^*)) \tag{2.21}
\]

therefore:

\[
(|k| + 1) \cdot X^\text{BS}_R(v(R,T_1^*)) = v(R,T_1^*) - v(R,T_2^*) + (|k| + 1) \cdot X^\text{BS}_R(v(R,T_2^*)) \tag{2.22}
\]

which we can re-write as:

\[
(|k| + 1) \cdot X^\text{BS}_R(v(R,T_1^*)) = v(R,T_1^*) + X^\text{BS}_R(v(R,T_2^*)) - v(R,T_2^*) + |k| \cdot X^\text{BS}_R(v(R,T_2^*))
\]

\[
= v(R,T_1^*) + X^\text{BS}_R(v(R,T_2^*)) - v(R,T_2^*) + |k| \cdot \left(\frac{v(R,T_2^*) - v(R,T_2^*)}{|k|}\right) + X^\text{BS}_R(v(R,T_2^*))
\]

\[
= v(R,T_1^*) + X^\text{BS}_R(v(R,T_2^*)) + X^\text{BS}_R(v(R,T_1^*)) - v(R,T_1^*) + (|k| - 1) \cdot X^\text{BS}_R(v(R,T_1^*))
\]

\[
= v(R,T_1^*) + X^\text{BS}_R(v(R,T_2^*)) + X^\text{BS}_R(v(R,T_1^*)) - v(R,T_1^*) + (|k| - 1) \cdot \left(\frac{v(R,T_2^*) - v(R,T_4^*)}{|k| - 1}\right) + X^\text{BS}_R(v(R,T_4^*))
\]
And generally we can write this as:

\[(|k| + 1) \cdot X_R^{BS}(v(R, T_1^*)) = v(R, T_1^*) + \sum_{i=2}^{j} X_R^{BS}(v(R, T_i^*) - v(R, T_j^*) + (|k| + 2 - j) \cdot X_R^{BS}(v(R, T_j^*)).\]

Specifically, for \(j = k + 1\), which is the value we are interested in, the formula is:

\[(|k| + 1) \cdot X_R^{BS}(v(R, T_1^*)) = v(R, T_1^*) + \sum_{i=2}^{k+1} X_R^{BS}(v(R, T_i^*) - v(R, T_k^*) + X_R^{BS}(v(R, T_k^*)).\]

Further, noting that \(X_R^{BS}(v(R, T_k^*)) = v(R, T_k^*)\) we achieve that:

\[(|k| + 1) \cdot X_R^{BS}(v(R, T_1^*)) = v(R, T_1^*) + \sum_{i=2}^{k+1} X_R^{BS}(v(R, T_i^*). (2.23)\]

or

\[X_R^{BS}(v(R, T_1^*)) = \frac{v(R, T_1^*)}{(|k| + 1)} + \frac{1}{(|k| + 1)} \sum_{i=2}^{k+1} X_R^{BS}(v(R, T_i^*). (2.24)\]

By our induction hypothesis, from (2.20) and from (2.24) we have that:

\[X_R^{BS}(v(R, T_1^*)) = \frac{v(R, T_1^*)}{(|k| + 1)} + \frac{1}{(|k| + 1)} \sum_{i=2}^{k+1} X_R^{BS}(v(R, T_i^*) \geq (2.25)\]

\[v(R, T_1^*) + \frac{1}{(|k| + 1)} \sum_{i=2}^{k+1} Sh_R(T_i^* \cup R, v) = Sh_R(T_1^* \cup R, v). (2.26)\]

Which completes our proof that \(X_R^{BS}(v(R, T_1^*)) \geq Sh_R(T_1^* \cup R, v).\)
Chapter 3

Selling to Strategic Consumers: on the Benefits of Valuation Uncertainty

3.1 Introduction

Consumers are oftentimes uncertain about their valuation for future consumption of services and products. This arises in many instances, such as in tourism and travel, where consumers may develop positive expectations or face scheduling conflicts. Changes in valuation could be internally or externally induced. Considering the previous example of tourism and travel, the latter change can be stimulated by external inputs such as major events (sports, exhibitions, concerts) that coincide with the travel plans and may be perceived both positively and negatively by different consumers.1

When making a purchase decision, consumers possess some initial valuation for this good. Consumers may recognize that their valuations could change over

---

1Examples from other industries are numerous. For instance, season passes that are purchased ahead of the season in a variety of recreational sports. Another example is entertainment events such as concerts and sport matches. An interesting example emerges from the world of sports memorabilia stimulated by transfers of players from one team to another (consider, for example, the transfer of the hockey player Wayne Gretzky from Edmonton to LA in the early 1980s, or that of the NBA player James Lebron from Cleveland to Miami).
time. Consumers who account for those possible future valuation fluctuations behave strategically, whereas their counterparts, who are unaware of these fluctuations, or simply ignore them, can be characterized as myopic. In this chapter we seek to identify the seller’s benefits that may arise from encountering strategic, rather than myopic consumers who face valuation uncertainty. Specifically, we are interested in addressing the following question: Can the seller be better off when all consumers behave strategically than when they all behave myopically? Presumably, the seller would take advantage of the consumers’ changing valuations, yet, exploiting rational expectations, strategic consumers would predict this behavior by the retailer and adjust their purchasing decisions in the first period accordingly. Thus, strategic consumers face a tradeoff in the first period: purchase immediately, when the price might be lower, or wait for the second period until the valuation uncertainty is resolved. Hence, it is not clear a-priori whether the seller can exploit the consumers’ awareness of their future uncertainty and be better off.

Modeling sellers’ dynamic pricing decisions in the presence of strategic consumers can be traced back to the seminal works by Coase (1972) and Stokey (1981). These were later followed, among others, by works such as Gul et al. (1986) and Besanko and Winston (1990), and more recently by Su (2007), Aviv and Pazgal (2008), and Dasu and Tong (2010). In this emerging literature, it is generally assumed that the evolution of prices over time is such that the pricing scheme is non-increasing over time. This is an outcome of discounting imposed on consumers’ valuations (or surplus) over time. Strategic consumers anticipate the non-decreasing pricing behavior, and respond accordingly by timing their purchases so that they maximize, separately, their consumer’s surplus. The resulting equilibrium is such that the presence of strategic consumers hurts the seller’s revenue—strategic consumers are willing to wait as they consider buying the good now versus later. Hence, they introduce inter-temporal substitution which exposes the seller to inter-temporal competition with herself, ultimately forcing the seller to reduce its prices, thereby, reducing the profit.

The operations management literature has seen a recent surge in studies addressing revenue management in the presence of strategic consumers. Some of the related issues addressed include dynamic pricing decisions (e.g., the special issue of the European Journal of Operational Research on revenue management
and dynamic pricing (Levin and McGill (2009)), as well as capacity and inventory decisions, such as rationing, display of inventory (is all inventory displayed or just one unit at a time), and replenishment (e.g., Liu and Van-Ryzin (2008); Yin et al. (2009); Cachon and Swinney (2009)). The recent focus of this literature is on mitigating the negative effect on revenue and profit associated with the presence of strategic consumers. For recent reviews see Shen and Su (2007) and Aviv et al. (2009). Though valuation uncertainty has been mentioned as an issue of importance in a number of papers (e.g. Su (2007); Swinney (2011)), it has received little attention in the literature.

Our interest is in a setting where a seller offers a good for sale during an advanced-selling period, and the consumers, who arrive at this time, face valuation uncertainty about their future valuation at time of consumption, which occurs during the selling period. Existing work is generally concerned with the conditions under which sellers should offer advance sales of a good. For example, Shugan and Xie (2001) show that advance selling can be used as a tool to mitigate the seller’s inability to price discriminate consumers, if consumers face uncertainty regarding their future valuations; Nocke et al. (2010) define conditions upon which a discounted advanced selling period price serves as a price discrimination device; Fay and Xie (2010) compare and contrast advance selling and probabilistic selling. They provide conditions on heterogeneity of consumer’s valuation for which it may be beneficial for the seller to deploy each approach in order to mitigate uncertainty in demand; and Chu and Zhang (2011) investigate the relationship between information sharing and a pricing decision when advance-selling is offered. The setting is such that before selling commences, the seller provides some information regarding the properties of the product, and announces a pricing scheme for pre-order units and regular sales period units of the product. Their main conclusion is regarding the optimal information sharing strategy, which depends on the expected profit margin and the standard deviation of consumer valuation.

In this chapter, we study, in a stylized fashion, a model that emphasizes the effects of valuation uncertainty and the consequences it may bear on dynamic pricing decisions in the presence of strategic consumers, who time their purchase to maximize their own utility. We show that, contrary to the mainstream literature on strategic consumers, due to consumers’ valuation uncertainty, the optimal pricing
scheme is such that it is not necessarily decreasing over time. This result has also been derived, for different reasons, by Su (2007) and Levin et al. (2010). Similar to Xie and Shugan (2001), we consider the conditions under which prices may increase over time, when the seller offers the good in an advance selling period. We also find that the presence of strategic consumers with uncertain valuations enables the seller to extract additional surplus from the consumers when compared to the case where consumers behave myopically. This finding is important. As mentioned before, the literature is in agreement on the perils posed to sellers due to the presence of strategic consumers. By contrast, we reveal that the presence of strategic consumers may, in fact, be beneficial for sellers if the consumers are faced with valuation uncertainty.

We proceed to investigate how our result could be further exploited by the sellers when they deploy a return policy. Return policy is a common practice used to mitigate the effects of consumers’ valuation uncertainty, and induce early purchasing. Indeed, we find that offering consumers the option of returning the product can, in our setting, enhance sellers’ revenues. Moreover, the sellers may benefit from offering a full, rather than a partial refund to consumers upon return of the product.

The rest of this chapter is organized as follows: In Section 3.2 we present our model, which is analyzed and discussed in Section 3.3. Consumer return policies are considered in Section 3.4. We present a numerical study in Section 3.5 and concluding remarks in Section 3.6. Throughout this chapter we assume the costs for the seller are normalized to zero, and therefore we use the terms profits and revenues interchangeably.

### 3.2 Model

We study a stylized two-period, dynamic pricing model, in which a profit maximizing seller, who is endowed with $Q$ units of a product, encounters strategic consumers. At the beginning of the first period, to which we refer as the advance-selling period, the seller announces the first period price, $p_1$, and $N_1$ consumers show up in the first period independently of the announced price ($N_1 \sim \text{Poisson}(\Lambda_1)$). The consumers who arrive in this advance-selling period face uncertainty regard-
ing their valuation for the product. They possess an initial valuation, but, being strategic consumers, they are aware this valuation may change as described below in Section 3.2.1. We assume that a proportion $\gamma_1$ of the consumers that show up in the advance-selling period have an initial valuation of $V_L$, whereas the remaining fraction, $1 - \gamma_1$, of the consumers initially value the product at $V_H$.

The consumers that arrive during the advance-selling period decide whether to buy the product at the posted advance-selling price, $p_1$, or postpone their purchasing decision until the second period. We refer to the second period as the selling period. When making this wait-or-buy decision in the advance-selling period, each consumer accounts for three important factors: (i) The fact that in the selling period there may be a different selling price, $p_2$; (ii) the possibility of a stock-out; and (iii) the possibility of a different personal valuation for the product.

At the beginning of the selling period the valuation uncertainty is resolved and the seller announces the second period price, $p_2$. All consumers that arrived in the advance-selling period who have decided to wait, return to the store, and, at the same time, another cohort of $N_2$ consumers arrive, independently of the announced second period price ($N_2 \sim \text{Poisson}(\Lambda_2)$). Of the new consumers that arrive, a fraction of $\gamma_2$ value the product at $V_L$, and the remaining fraction, $1 - \gamma_2$, value the product at $V_H$. Sales are carried out, and the season ends.

### 3.2.1 Consumers’ Valuations and Decisions

The consumers that arrive in the advance-selling period face valuation uncertainty. We model this uncertainty by assuming that the consumers have an initial valuation $V_L$ or $V_H$, and allow this valuation to change in a probabilistic fashion. Specifically, consumers with initial valuation $V_L$ will end up either with valuation $V_L$ with probability $\alpha$ or with valuation $V_H$ with probability $(1 - \alpha)$, whereas consumers with initial valuation $V_H$ will end up either with valuation $V_L$ with probability $\beta$, or with valuation $V_H$ with probability $(1 - \beta)$. This is summarized in Table 3.1. The change in valuation occurs simultaneously, following the end of the advance-selling period, for all consumers that arrive in the advance-selling period.

Since each consumer knows his own initial valuation for the product and the probability of change of this valuation, he can make a purchasing decision so that
Table 3.1: Valuation Transition Probability

<table>
<thead>
<tr>
<th>Initial Valuation</th>
<th>Final Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_L$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>$V_H$</td>
<td>$1 - \alpha$</td>
</tr>
</tbody>
</table>

his own expected surplus is maximized. Specifically, consumers who show up in the advance-selling period compare their expected surplus from buying during the advance-selling period with the expected surplus from postponing the purchasing decision to the selling period. Essentially, the consumer calculates the expected surplus of buying the product in the first period, accounting for the possibility his valuation will change. He then compares this value with the expected surplus he may achieve if he waits for the second period. This future surplus accounts for three elements: the price that the consumer expects to encounter during the selling period, his future valuation of the product, and product availability, i.e. the probability he will be able to obtain a unit of the product in the selling period. We denote by $\hat{p}_2$, the rational expectations (RE) price estimate that all the consumers deduce that the seller will announce in the selling period. This accounts for the seller’s goal of maximizing overall revenues when setting prices for the two periods. We also denote by $H(\cdot)$, the RE availability probability of the product, that is, the probability that a consumer who wants to buy a unit in the second period will not face a stock-out. Note that $H(\cdot)$ may depend on the initial valuations, the prices, the consumer arrivals (i.e., market sizes) in the two periods, the initial inventory, as well as the valuation fluctuation probabilities. Using these notation, we summarize in Table 3.2 the surplus comparison that a consumer would carry out in the advance-selling period. This comparison accounts for the three types of risk described above. Note that if a consumer postpones his purchasing decision to the selling period, at which the second period price, $p_2$, is announced, the decision the consumer faces becomes trivial—purchase the product if the immediate surplus is non-negative (that is, if the valuation is equal to, or exceeds, $p_2$).

Note that RE enables us to conclude that all consumers of the same type should

---

2By assuming RE we refer to the following implicit assumption: the consumers assume that given the observed advance selling price, the retailer will price optimally in the selling period.
Table 3.2: Surplus Comparison: Buying In Period 1 vs. Waiting For Period 2

<table>
<thead>
<tr>
<th>Initial Valuation</th>
<th>Surplus From Buying In Period 1:</th>
<th>Surplus From Buying In Period 2:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_L$</td>
<td>$-p_1 + \alpha V_L + (1 - \alpha) V_H$</td>
<td>$(-\hat{p}_2 + \alpha V_L + (1 - \alpha) V_H) H(\cdot)$</td>
</tr>
<tr>
<td>$V_H$</td>
<td>$-p_1 + \beta V_L + (1 - \beta) V_H$</td>
<td>$(-\hat{p}_2 + \beta V_L + (1 - \beta) V_H) H(\cdot)$</td>
</tr>
</tbody>
</table>

Note that the consumer will purchase in the second period only if the realized surplus is non-negative.

behave in the same manner, i.e., they either all buy immediately or all wait for the second period. This simplifies the analysis that the seller needs to carry out.

3.2.2 The Seller’s Pricing Decisions

We now turn our attention to the pricing decisions made by the revenue maximizing seller. Recall that a consumer that arrives in the advance-selling period compares his expected utility from purchasing the product immediately with the expected utility from deferring his purchasing decision to the second period. The seller’s pricing decision in the first period, $p_1$, induces one of three different possible consumer behaviors. If the first period price is sufficiently high, then all consumers who arrive in the advance-selling period refrain from purchasing the product in that period. If the price is sufficiently low, then all the consumers who are present in the first period purchase the product. Finally, if the price is within some intermediate range, then only some of the consumers who arrive in the first period purchase the product.

In order to determine the price ranges that determine the aforementioned high, low, and intermediate prices, we note the following. There is a unique threshold price, denoted by $\bar{p}_1$, such that if $p_1 \leq \bar{p}_1$ all consumers present in the first period buy the product immediately. Further, there is a unique threshold price, $\bar{p}_1$, such that if $p_1 > \bar{p}_1$, all consumers in the advance-selling period defer their purchasing decision to the second period. Intermediate pricing, $p_1$, $\bar{p}_1 < p_1 \leq \bar{p}_1$, induces mixed behavior where some consumers buy in the first period whereas the others wait for the selling period.3

It can be shown that a choice of an intermediate price may induce only con-

---

3Note that inducing postponement of purchase behavior by all consumers can be easily done by setting $p_1 > V_H$. Similarly, inducing purchasing behavior such that all consumers buy in the first period, can be achieved by setting $p_1 < V_L$. 

30
sumers with initial valuation $V_L$ (resp. $V_H$) to purchase in the first period, if the valuation fluctuation probabilities are such that $\beta > \alpha$ (resp. $\beta < \alpha$). As the pricing decision, $p_1$, induces a unique consumer behavior in the advance-selling period, we can exploit this segmentation behavior to determine the remaining inventory for the selling period. Consequently, we can determine the optimal pricing decisions made by the seller in both the advance-selling and selling period, as we show below.

The seller’s pricing decision is such that she maximizes her total revenue from the sale of the $Q$ available units. We also assume that a stock-out, if occurs, is experienced only in the selling period. That is, there are sufficient units to satisfy demand in the first period. The seller, therefore, solves the following optimization problem:

$$\max_{(p_1, p_2(p_1, n_1))} E\{p_1 n_1 + p_2(p_1, n_1)n_2\}$$

Subject to

$$n_1 + n_2 \leq Q \quad (3.1)$$

$$p_1, p_2(p_1, n_1) \geq 0,$$

where $n_1$ and $n_2$ are the actual sales quantities in each period. These quantities are functions of the pricing decisions as described above, as well as of the actual realizations of the consumer arrivals.

### 3.3 Model Analysis

We use backward induction to solve for the pricing decisions that the seller makes. We state the following lemma about possible prices in the second period (all proofs are provided in the appendix).

**Lemma 1** The price set by the seller in the selling period, $p_2$, is either $V_L$ or $V_H$.

The intuition that drives this conclusion is as follows. After valuation uncertainty is resolved the consumers are pooled into two distinct valuation groups - those with valuation $V_H$ and those with valuation $V_L$. The seller can choose to set a low price that attracts all consumers, or alternatively she can set a high price such
that only high valuation consumers purchase the product in the second period. If
the seller decides to sell only to the high valuation group, she extracts all surplus
from those consumers by selling at a price \( V_H \). In this case, the seller may end up
with some unsold units. Alternatively, the seller can target all consumers present in
the selling period by announcing a selling price of \( V_L \). By pricing at \( V_L \), the seller
extracts all surplus from the low valuation consumers who are able to obtain a unit
of product—note that stock-outs may occur—and the high valuation consumers
who are able to obtain a unit of product are left with some positive surplus. These
high valuation consumers retain a surplus of \((V_H - V_L)\). Any other choice of pricing
in the second period is inferior to those suggested above. Pricing at a price differ-
ent from either \( V_L \) or \( V_H \) in the second period leaves the consumers with surplus
that the seller could have extracted without effecting the consumers’ purchasing
decision.

We now use Lemma 1 to analyze the seller’s decision in the selling period. The
seller needs to compare her expected revenues from setting \( p_2 = V_L \) or \( p_2 = V_H \).
We denote the seller’s expected revenue in the second period by \( \pi_2 \). Then, we have:

\[
\pi_2 = \max_{p_2} \{ p_2 \cdot \min (Q - n_1, E[Demand_2(p_2)]) \}. \tag{3.2}
\]

The expected demand in the selling period, denoted \( E[Demand_2(p_2)] \), can be de-
duced using Lemma 1. Consider Table 3.3 below wherein we develop the expres-
sions for \( \pi_2 \) for each of the possible combinations of the first period pricing de-
cision (and the corresponding segmentation policy) and the second period pricing
decision (where \( p_2 \in \{V_L, V_H\} \)).

<table>
<thead>
<tr>
<th>Advance-Selling Price</th>
<th>Advance-Selling Segmentation*</th>
<th>Expected Revenue in Selling Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_1 &gt; p_l )</td>
<td>No One</td>
<td>( V_L E[\min(Q, A_1 + A_2)] )</td>
</tr>
<tr>
<td>( p_l &lt; p_1 \leq p_l ) and ( \beta &gt; \alpha )</td>
<td>Those with initial valuation ( V_L )</td>
<td>( V_L E[\min(Q - \gamma_1 A_1, (1 - \gamma_1) A_1 + A_2)] )</td>
</tr>
<tr>
<td>( p_l &lt; p_1 \leq p_l ) and ( \beta &lt; \alpha )</td>
<td>Those with initial valuation ( V_H )</td>
<td>( V_L E[\min(Q - (1 - \gamma_1) A_1, \gamma_1 A_1 + A_2)] )</td>
</tr>
<tr>
<td>( p_1 \geq p_l )</td>
<td>All</td>
<td>( V_L E[\min(Q - A_1, A_2)] )</td>
</tr>
</tbody>
</table>

\*This column characterizes the consumers who buy in advance-selling period.
Note that Table 3.3 enables the computation of all expected revenues, and from this the seller, and consumers, can conclude the optimal pricing policy the seller deploys when maximizing her expected revenues.

Since all parameters are common knowledge, both the seller and all (strategic) consumers can infer the second period price for each possible scenario. The subsequent analysis that needs to be carried out is that of the first period pricing. We denote the maximum expected revenue in the selling period by $\pi^*_2$, and the decision the seller makes in the first period is then:

$$p^*_1 = \arg\max_{p_1} E \{ p_1 n_1 + \pi^*_2 \}.$$  (3.3)

We note that $n_1$ is a function of both $p_1$ and $N_1$. We can replace (3.3) with a comparison of values based on the parameters of the model. Evaluating $H(\cdot)$ explicitly can be quite difficult (as discussed in Yin et al., 2009). However, the main insight stemming from this model does not require an explicit solution for $H(\cdot)$. The value of $H(\cdot)$ is derived numerically in Section 3.5 where we present a numerical study of the model.

We proceed with the investigation of the effect of valuation uncertainty on pricing decisions. While it is not very surprising, the introduction of a possible rise in consumer valuation (from $V_L$ to $V_H$) brings about the possibility of price increases between the advance-selling period and the selling period. Indeed, this result has already been demonstrated under certain properties such as “degrees of patience” (i.e., how patient consumers are with respect to making the decision regarding their purchase) as in Su (2007), or what may be referred to as end of planning horizon effects as in Levin et al. (2010). In contrast, our conclusion stems from a rather intuitive and objective notion of valuation uncertainty. We first state this result.

**Proposition 2** For each possible segmentation policy induced by the optimal advance-selling period pricing decision, $p^*_1$, there exists a set of parameters under which $p_2 > p^*_1$.

This result stems from valuation uncertainty, and is driven by the fact that there is a probability that consumers may experience a product valuation increase between the first and second periods. Further, this result reveals that some consumers,
arriving in the advance-selling period, are willing to accept higher prices for the product in the selling period after resolving uncertainty regarding their valuation for the product, rather than pay a lower price when still facing some uncertainty. While this result may be somewhat intuitive, and may be regarded to as a notion of value of information, it is surprising that the same type of analysis may yield, under different combination of valuation fluctuation probabilities, a result that is exactly the opposite. This means that some consumers may be willing to risk facing strictly negative surplus in order to obtain a unit of product. This result is presented in the following proposition:

**Proposition 3** Assume the seller commits to $p_2 = V_L$, then there exists a set of parameter values such that a strategic consumer who arrives in the advance-selling period, with an initial valuation $V_L$, chooses to buy in the advance-selling period at a price $p_1^* > V_L$.

This result stems from the trade-off between expected positive utility, and the probability of obtaining the product, $H(\cdot)$. However, it is still somewhat counter intuitive that a strategic consumer forgoes a future surplus, which is guaranteed to be non-negative, for an immediate negative surplus and a future surplus that may not be positive.

We next turn our attention to the related issue of the effects of the presence of strategic consumers on the seller’s revenue. Before proceeding, we first define a reference point to our analysis — the case where consumers are myopic rather than strategic. These myopic consumers are short sighted and either ignore or are not aware that their valuation may change from the first to the second period. When facing a purchase decision, they only consider their current valuation and the current posted price. That is, myopic consumers do not optimally time their purchase, like the strategic consumers. When faced with myopic consumers, any first period pricing decision, $p_1$, set by the seller that induces consumers with valuation $V_L$ to buy, will induce consumers with valuation $V_H$ to buy as well. Therefore, the seller cannot segment the myopic consumers in the first period to the same degree she is able to when the consumers are strategic. This is an important insight that leads to the following theorem.
Theorem 2 There exists a set of parameters under which the seller is better off facing strategic, rather than myopic, consumers.

We illustrate this with an example. Assume that the seller faces strategic consumers and that the parameters are such that the optimal decisions the seller should make are such that consumers with initial valuation of $V_L$ buy the product in the advance-selling period, while consumers with initial valuation of $V_H$ defer their purchasing decision to the second period. By contrast, assume that the seller faces myopic consumers. In the advance-selling period, this segmentation (where only low valuation consumers purchase the product in the first period) cannot be achieved since myopic consumers with initial high valuation always buy the good if low valuation consumers buy the good as well. Hence, several segmentation policies are excluded when a seller faces myopic, rather than strategic consumers.

Further, in the presence of strategic consumers, there are scenarios in which the seller’s pricing decision for the advance-selling period may be higher than for the case of myopic consumers. This stems from the fact that myopic consumers do not account for future valuation when making their purchasing decision in the advance-selling period. Selling to myopic consumers with initial valuation $V_L$ implies an advance-selling period price of $p_1 = V_L$. However, if the same consumers were to be strategic and account for their future valuation, the selling price would be higher (assuming a positive probability of valuation increase). To conclude, these examples demonstrate that some segmentation possibilities available when facing strategic consumers are not available when facing myopic consumers, and limit the ability of the seller to benefit from these policies.

The main conclusion from Theorem 1 is that facing strategic consumers may give rise to new possibilities when segmenting a market of consumers with uncertain valuation which would be beneficial to the seller.\(^4\) Indeed, contrary to current main stream operations and revenue management literature, the presence of strategic consumers with uncertain valuation could be beneficial to the seller as compared to the situation where she faces myopic consumers.

\(^4\)Specifically, the seller can attempt to raise awareness to future valuation fluctuations. For example, a targeted marketing campaign may induce certain consumers to behave in a strategic rather than myopic manner, a positive and beneficial behavior for the seller in some cases.
The next natural step in our analysis is to examine the common practices used to mitigate the negative effects that emerge due to the presence of strategic consumers. This is important as we now focus on enhancing the possible positive effects strategic consumers offer. The literature consider two well studied practices: Price guarantees and return policies. Price guarantee is a mechanism used by sellers to impact the decision undertaken by consumers by providing the consumers with a promise for compensation in case of a future price decrease. For an overview of price guarantees we refer the reader to Aviv et al. (2010), and Nalca et al. (2010). The second mechanism—a return policy—allows consumers to return a product for some refund (see Su, 2009 for a review). A returns policy focuses on mitigating valuation uncertainty. It attempts to induce purchasing when valuation uncertainty exists. Since returns policies are related to valuation uncertainty, rather than the prospect of creating excess surplus in case prices decline, we analyze in the next section its incorporation into our framework.

### 3.4 The Effect of Consumer Return Policies

As stated in Su (2009), revenue management in the presence of strategic consumers is closely related to the topic of consumer return policies. Further, valuation uncertainty is one of the key drivers of the analysis presented in Su (2009). In this section we analyze how incorporating return policies in our model, effects the seller’s optimal pricing decisions. Our analysis differs from the one in Su (2009) in three main aspects: (i) we examine a two period model, and not a single period model, as in Su (2009); (ii) we allow for units return to be resold as unused units (this is relevant, for example, in the case of tickets for a sporting events) rather than sell them as used units; and (iii) we restrict our analysis to a choice between two discrete valuations (namely $V_L$ and $V_H$) rather than a continuum of valuations.

Our analysis in this section is based on the model we presented in section 3.3 with the following adjustments: When announcing the advance-selling price, $p_1$, the seller also announces the refund value, $r$, that she will reimburse consumers that choose to return the product. The return of products is made prior to the initiation of the selling period, and before the seller announces the selling period price, $p_2$.

While it is evident that the possibility to return the product may put the seller in
a compromising position (e.g., having excess inventory by the end of the season), it may also be viewed as a mitigating mechanism for consumers’ valuation uncertainty by providing the consumers with some insurance in the case that the realized valuation is not as high as they anticipate, or in the case the surplus generated is negative. Further, the ability to offer returned products for resale, enables the seller to explore possibilities that are not available in the case returns are not allowed. This means that a finer segmentation policy, compared to the ones analyzed in our model, may be available for the seller to exploit. Our conclusion, later formally stated as part of Proposition 4, is that there exist a set of parameter values such that the seller benefits from providing consumers with the ability to return the product for some refund $r > 0$.

The intuition behind this result is as follows. A returns policy allows the seller to extract some surplus from high-valuation consumers that arrive in the selling period, on account of consumers that bought the product in the first period, for some price $p_1 > V_L$, and later realized their valuation for the product is actually $V_L$, and therefore chose to return the product for a refund. The proposition reveals that there are scenarios that enable the seller to enhance the positive effects of facing strategic consumers by offering the ability to return products for a refund.

We turn to analyze the extent to which the seller should offer the refund in the case of returns. Both Xie and Gestner (2007) and Su (2009) find that there exist conditions where offering the ability to return products for a partial refund is profitable for the seller. While Xie and Gestner (2007) only analyze partial refunds, Corollary 1 of Su (2009) states that partial refunds are always superior to full refunds. In our setting, similar to Xie and Gestner, the seller can resell returned units of product and create a positive surplus. Due to this possibility we are able to state the following.

**Proposition 4** There exist a set of parameter values such that the seller benefits from providing consumers with the ability to return the product for a refund. Further, there is a subset of these parameter values such that the seller is better off when providing consumers with a full, rather than partial refund upon the return of the product.

The result in Proposition 4 stems from the trade-off between the price the seller
is able to charge in the advance-selling period, and the extra surplus that she may achieve in the selling period. This additional surplus is based on the pricing decision that accounts for reselling of the returned units and the refund paid to consumers returning the product. As full refunds become optimal according to Proposition 4, one may be tempted to consider more than full refunds. However, we elect to ignore such refunds, as they may induce speculative behavior by consumers, an issue that is beyond the scope of our work.

The main conclusion of our analysis regarding returns policy in the presence of strategic consumers with uncertain valuation for the product is that this practice may benefit the seller, as it acts as an enhancer of the beneficial effects that strategic consumers may impose. Further, providing consumers with a full, as opposed to partial refund in case of returns may be beneficial to the seller.

3.5 Numerical Study

As mentioned above, the seller’s pricing decisions and the consumers’ purchasing decisions are inter-related. This interdependency is of particular interest strategic consumer are present. A key element in this interdependency is the probability that a consumer who defers a purchasing decision to the future is able to obtain a unit of product in the future if he does choose to purchase. As extensively discussed in Yin et al. (2009), finding this probability, $H(\cdot)$, is a tedious and complex task. Moreover, in our setting, this task encounters further complexities due to valuation uncertainty—an issue not addressed in Yin et al. (2009). Further, noting that all consumers of the same type should make the same decision, results in four possible consumer surplus levels that need to be analyzed. Thus, while it is essential to find $H(\cdot)$ in order to achieve equilibria decisions, these complicating circumstances prevent us from presenting an explicit formulation of $H(\cdot)$. Therefore, we resort to a numerical analysis to illustrate our insights.

We were able to avoid the aforementioned difficulties by using the following approach. We use the thresholds on pricing decisions described in Table 3.3, and the purchasing decisions that are induced, which provides us with bounds on the revenues achievable under each possible policy. We then check which of the resulting prices leads to an equilibrium outcome by numerically comparing all possible
cases of consumer behavior. For each set of parameters, we calculate the maximal profits for each segmentation policy. The first period pricing decisions for each policy are the maximal prices that induce this policy, and we then verify that the resulting optimal policy for the seller (i.e., the one resulting in the highest revenue) is an equilibrium, i.e., no consumer class is better off by changing its decision. In this case we used the resulting expected number of consumers making a purchase for the evaluation of $H(\cdot)$ and compared all possible cases for each policy to ensure this is indeed an equilibrium.

To demonstrate that no single policy universally dominates, in Figure 3.1 we depict the revenues generated under each of the segmentation policies as a function of the initial inventory. Note that for each value we verified that this is indeed an equilibrium.
Figure 3.1: Revenue Generated For Each of the Pricing Policies as a Function of Initial Inventory: $V_L = 1$, $V_H = 2$, $\alpha = 0.9$, $\beta = 0.95$, $E(N_1) = 40$, $E(N_2) = 50$, $\gamma_1 = 0.1$, $\gamma_2 = 0.75$. To distinguish between the pricing policies, a two-letter notation is employed, the first represents the first period purchasing segmentation: A for All, B for consumers with initial valuation $V_L$, C for consumers with initial valuation $V_H$, and D for None; the second letter stands for the price in the second period: L for $V_L$ and H for $V_H$. 
Figure 3.1 depicts several segments based on the initial inventory, $Q$. In Segment I, where the initial inventory is low, the optimal policy is to price the product so that no consumers buy the product in the first period, and then price the product at $V_H$, clearing all inventory. The intuition behind this policy is that due to the low levels of inventory the seller can generate a revenue of $V_H \cdot Q$, as enough consumers, i.e. at least $Q$, will be willing purchase the product in the selling period at $p_2 = V_H$. In Segment II - the pricing decision is such that only consumers with initially low valuation for the product purchase it immediately. The posted price in the advance-selling period is equal to the expected valuation of the low valuation consumers. In the selling period the product is sold at a price $p_2 = V_H$. The optimal policy in this segment is such that it trades-off guaranteed advance-selling revenues with the possibility of a higher per-product revenue in the selling period but also the risk of unmatched supply at the selling period price. The optimal pricing policy suggested in segment III is such that it aims to induced advance-selling of the product to consumers with initially high valuation for the product. For the analyzed set of parameters this suggested policy is not an equilibrium as it is dominated by the strategy AH, in which ALL consumers arriving in the first period purchase the product, and the second period price is $p_2 = V_H$. AH, is also the equilibrium optimal policy in segment IV. In segment V the optimal policy is such that it induces the purchase of the product by all consumers in the advance-selling period. Due to the abundance of inventory remaining in the selling period, the optimal pricing policy is a market clearing one, where $V_L$ is the selling period price.

Recall that we assume the seller’s costs are normalized to zero, and therefore in our analysis we treat revenue maximization and profit maximization as one. We claim that our numeric analysis regarding revenue maximization can help make optimal capacity decisions that would maximize profits. This can be achieved by one of two possible courses of action. The first is to incorporate the costs into the expressions being plotted, i.e. plot profits rather than revenues, and select the optimal course of action, ensuring it constitutes an equilibrium purchasing decision for the consumers. Alternatively, we can plot the cost curve together with the revenue curves. The optimal decisions would be inventory levels for which the derivatives of both curves is equal (as we recall from basic microeconomics). Note that there may be numerous optima, and the difference between them would be the
resulting optimal pricing policy and initial inventory levels.

When costs are normalized to zero, the optimal pricing policy and the corresponding purchasing behavior of consumers is subject to model parameters. However, for certain inventory levels the optimal pricing policy is independent of model parameters. Specifically, for very low initial inventory it is always optimal to price such that all consumers defer their purchase, and in the second period the price is $V_H$. Above a certain level of inventory, it is optimal to segment the market so that at least part, if not all of the first period consumers purchase the product in the first period, and the optimal second period price is $p_2 = V_H$. Finally, there is always a high enough inventory level such that the second period pricing policy is $p_2 = V_L$. This behavior is consistent with intuition and with the example depicted in Figure 3.1.

Next, we compare the profits derived in two contrasting cases. In one, the seller faces only strategic consumers. In the other, she faces only myopic consumers. In Figure 3.2 we depict the profits associated with optimal pricing policy for each of the two cases, for a given set of parameters (valuations, arrival rates, and fluctuation probabilities). It follows from Figure 3.2 that with low inventory levels ($Q \leq Q^*$) the seller is better off facing myopic, rather than strategic consumers. She is able to capture all consumer surplus from high valuation consumers in both periods, which she is not able to do when facing strategic consumers, as the optimal $p_1$ is strictly less than $v_H$. For higher initial inventory ($Q > Q^*$) the seller is better off selling to strategic, rather than myopic consumers. The intuition behind this result is as follows. When facing strategic consumers, the seller is able to extract, upfront and from all consumers, the surplus resulting from the possibility of future high valuation. Further, the seller is able to sell off all inventory in the second period. This is in contrast with the case the seller faces myopic consumers, where all consumer surplus for high valuation consumers is extracted, but some inventory remains unsold. This example also proves Theorem 1.

3.6 Conclusions

We have shown that when a seller faces consumers who have uncertainty regarding their final product valuation, she may choose to increase the price of the product
Figure 3.2: Profit When All Consumers Are Myopic and When All Consumers Are Strategic; $V_L = 1$, $V_H = 2$, $\alpha = 0.05$, $\beta = 0.1$, $E(N_1) = 40$, $E(N_2) = 5$, $\gamma_1 = 0.75$, $\gamma_2 = 0.75$.

Our main insights provide a new perspective on the influence of strategic consumers in revenue management. It may provide some justification to revisit well known results and incorporate the effect of valuation uncertainty on revenue management decisions in the presence of strategic consumers. Further, our study motivates further research into the benefits of targeting consumers. More specifically, the seller may benefit from targeting certain consumers, turning their attention to the possibility of valuation uncertainty, and, as a result, induce more strategic behavior.

This paper can be used as a basis for future research in the area of strategic consumers. First, finding bounds on model parameters to satisfy the main results could provide useful when applying dynamic pricing when facing strategic consumers. Next, this model assumes a fixed capacity of stock. Future research can
incorporate replenishment possibilities between the two periods or incorporation of an initial stocking decision to the model.

3.7 Proofs

In this section we provide proofs not provided as part of the body of the essay.

Proof of Lemma 1.

We show that any feasible value of $p_2$ is dominated either by $p_2 = V_H$ or by $p_2 = V_L$. Note that if $p_2 > V_H$, no consumers buy the product; therefore, $p_2 = V_H$ dominates $p_2 > V_H$. If $p_2 \leq V_L$, all consumers in the market in the second period attempt to buy a unit of product. However, setting $p_2 = V_L$ trivially generates more profit than setting $p_2 < V_L$. For the remaining interval of $V_L < p_2 < V_H$, notice that any price his range does not result in purchase of the product by low valuation consumers. Therefore, if the seller seeks to attract low valuation consumers she must price the product at $p_2 = V_L$. If the seller is not interested in attracting low valuation consumers, she will simply set $p_2 = V_H$. Hence, $p_2 = V_L, V_H$.

Proof of Proposition 1.

Considering Table 3.3, it is evident that there exists a set of parameters such that for any given $p_1$ there is a distinguishable advance-selling purchasing behavior by the consumers that arrive in the first period, and further, there exist a set of model parameters such that the seller’s optimal second period pricing decision is to set $p_2 = V_H$. To prove our proposition, we show that there exist a set of parameters such that the first period price is lower than $V_H$ and that this first period price still results in the correct purchasing behavior by the consumers.

Since we are looking for model parameters such that $p_2 = V_H$, we can assume that the consumers arriving in the first period face an expected surplus of zero in the second period. In order to induce the required consumer behavior, the seller needs the consumers in the first period to yield a positive surplus so that they purchase the product in the first period. We consider the three price levels and the (four) resulting purchasing segmentation policies presented in Table 3.3 separately.

Pricing scheme $p_1 \geq \bar{p}_1$, which results in a advance-selling purchasing decision by all consumers arriving in first period. In order for this to hold, consumers with initial valuation $V_L$ need to obtain a positive surplus relative to the pricing scheme:
−p_1 + V_L + (1 − \alpha)(V_H − V_L) ≥ 0 \Rightarrow \alpha V_L + (1 − \alpha)V_H ≥ p_1. Consumers with initial valuation \( V_H \) also need to obtain a positive surplus: 

\[-p_1 + V_H + \beta(V_H − V_L) ≥ 0 \Rightarrow \beta V_L + (1 − \beta)V_H ≥ p_1.\]

For \( p_1 = \text{Minimum} \{ \beta V_L + (1 − \beta)V_H, \alpha V_L + (1 − \alpha)V_H \} \)
both the required constraints hold. Clearly there is a set of parameters for which the seller can set \( p_1 \) as above and \( p_2 = V_H > p_1 \).

Pricing scheme \( p_1 < p_1 ≤ \bar{p}_1 \). We split this case into two distinct cases of purchasing behavior of consumers that arrive in the first period: 1. For \( \beta > \alpha \), those with initial valuation \( V_L \) purchase the product in the first period. 2. For \( \beta < \alpha \), those with initial valuation \( V_H \) purchase the product in the first period.

For case 1: The seller will set \( p_1 \) so that a consumer’s with an initial valuation \( V_L \) buy in the first period and consumers of initial valuation \( V_H \) do not buy. This is achieved by when consumers with initial valuation \( V_L \) have a positive surplus in the first period whereas consumers with initial valuation \( V_H \) obtain a strictly negative surplus. A price \( p_1 \) that achieves this needs to satisfy \( \beta V_L + (1 − \beta)V_H ≤ p_1 ≤ \alpha V_L + (1 − \alpha)V_H \). It is easy to concoct a set of parameters that satisfies this condition along with the condition that the optimal second period price is \( p_2 = V_H \) as in Table 3.3, resulting in the desired result of \( p_2 = V_H > p_1 \).

For case 2: The setting is similar to case 1, but in this case the seller sets \( p_1 \) so that period 1 consumers with initial valuation \( V_H \) buy and consumers of initial valuation \( V_L \) defer their purchasing decision, and the proof is similar to case 1, exchanging \( \alpha \) and \( \beta \).

Pricing scheme \( p_1 > \bar{p}_1 \) induces a first period purchasing behavior such that all consumers defer their purchase to the second period. In order to achieve this we need all first period consumers to obtain a negative surplus, w.r.t. \( p_1 \). This is achieved, in a manner similar to above, by setting \( p_1 \) that satisfies \( p_1 ≥ \max \{ \alpha V_L + (1 − \alpha)V_H; \beta V_L + (1 − \beta)V_H \} \). Choosing \( p_1 \) in the range: \( [\max \{ \alpha V_L + (1 − \alpha)V_H; \beta V_L + (1 − \beta)V_H \}, V_H] \) results in the required consumer behavior as well as \( p_2 = V_H > p_1 \).

**Proof of Proposition 2**

A low valuation consumer purchases in the first period only if: 

\[-p_1 + \alpha V_L + (1 − \alpha)V_H ≥ (1 − \alpha)(V_H − V_L)H(\cdot).\]

This can be rewritten as 

\[-p_1 + V_H ≥ \alpha(V_H − V_L) + (1 − \alpha)(V_H − V_L)H(\cdot).\]

Since \( H(\cdot) < 1 \), we know that \( (V_H − V_L) ≥ \alpha(V_H − V_L) + (1 − \alpha)(V_H − V_L)H(\cdot). \) This, in turn, means that there exists \( p_1 > V_L \) that may satisfy: 

\[-p_1 + V_H ≥ \alpha(V_H − V_L) + (1 − \alpha)(V_H − V_L)H(\cdot).\]
Proof of Theorem 1

The existence of this result follows from the example provided in Section 3.5 (see Figure 3.2).

Proof of Proposition 4

If the seller offers a refund, r, then the consumers’ comparison of expected surplus in the first period must now account for this refund. The resulting comparisons are presented in Table 3.4. Note that we implicitly assume that the refund is only available to consumers who buy in the product first period, and whose final surplus from purchasing in the first period is negative, i.e. only consumers with a low valuation, \( V_L \), for the product in the selling period, if any, return the product for a refund.

Table 3.4: Surplus Comparison: Buying In Period 1 vs. Buying In Period 2

<table>
<thead>
<tr>
<th>Initial Valuation</th>
<th>Surplus From Buying In Period 1</th>
<th>Surplus From Buying In Period 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V_L )</td>
<td>(-p_1 + \alpha (V_L + r_f) + (1 - \alpha) V_H)</td>
<td>((-\hat{p}_2 + \alpha V_L + (1 - \alpha) V_H) H(\cdot))</td>
</tr>
<tr>
<td>( V_H )</td>
<td>(-p_1 + \beta (V_L + r_f) + (1 - \beta) V_H)</td>
<td>((-\hat{p}_2 + \beta V_L + (1 - \beta) V_H) H(\cdot))</td>
</tr>
</tbody>
</table>

Note that the consumer will decide to purchase in the second period only if the realized surplus is non-negative.

While the seller’s comparison for the selling period surplus now needs to account for the returned units when making the pricing decision, it is also apparent that the seller can charge a higher price in the advance-selling period compared to the case of no returns. Specifically, assuming the parameters are such that in the advance-selling period, only low valuation consumers buy the product, \( p_1 \) increases by \( \alpha r_f \), when compared to the case of no return for refund. We denote by \( S_r \) the amount of units returned by consumers (and notice this is easily calculated based on the pooling prices). Therefore, the revenues from the return policy, when the pricing policy is such that \( p_2 = V_H \), is: \( \alpha r_f (n_1 - S_r) + (V_H - r_f)n_2^* \), where \( n_2^* \) is the resulting number of units sold. In this case, the revenues may be an increasing function of \( r_f \), which implies that a refund policy benefits the seller, and further a full refund policy is better than partial refund policy.
Chapter 4

The Multi-Mode Resource-Constrained Cross-Dock Scheduling Problem

4.1 Introduction

Cross-docks are facilities in which inbound materials are rapidly consolidated, distributed, and transferred to outbound vehicles. Cross-docks reduce transportation costs by allowing for higher capacity utilization for inbound and outbound vehicles. Cross-docks also promote just-in-time (JIT) production methods by sending material to final destinations in small batches in the correct sequence to reduce inventory costs. A recent study by the Saddle Creek Corporation (2011) reported a significant increase in the number of firms that utilize cross-docks as part of their supply chain. The same report noted that firms that had recently implemented the use of cross-docks within their supply chain reported a 14.3% reduction in annual transportation costs. The research presented in this chapter is motivated by our work with a software firm that develops container packing software and with one of its clients, a large Canadian retailer, that uses cross-docking as its main means of transferring goods between suppliers and its retail locations.

There are two main operational decisions that drive the majority of operational
costs at a cross-dock: (1) assigning containers to dock-doors; and (2) scheduling containers at the cross-dock (accounting for resource requirements and constraints). Both inbound and outbound containers at a cross-dock must be assigned to a specific dock-door for the processing of freight. Improper assignments of containers to doors can result in excessive travel costs within the cross-dock. The arrival times of containers must also be carefully managed to allow for efficient use of limited resources such as labor and machines. Previous research has limited its focus to either the assignment or the scheduling problem. Our work considers both problems simultaneously, and, to the best of our knowledge, is the first to provide a solution approach, allowing solutions results for real-world problems.

Our solution approach to this problem is a methodological framework which we refer to as the Multi-Mode Resource-Constrained Cross-Dock Scheduling Problem (MRCDSP) method. The MRCDSP can be broken down into four stages: (1) container clustering; (2) dock-door assignment; (3) workflow scheduling; and (4) container scheduling. Each of these four stages of the MRCDSP are described in details in Section 4.3. The flow of information and the relationship between the different stages of the MRCDSP are depicted in Figure 4.1. The container clustering stage uses the material flow between containers data as input and provides disjoint sets of containers as output. The resulting clusters are then used as input to the dock-door assignment phase which provides a container to dock-door allocation as output. These results then serve as inputs to the workflow scheduling stage, along with the detailed material flow data. The outputs of the workflow scheduling stage are a variety of potential modes of execution (as explained in Section 4.3.4) and associated processing times, which serve, along with the layouts achieved in the dock-door assignment stage, as inputs to the container scheduling stage. The final result is a detailed allocation schedule of resources to containers that minimizes the makespan required to process all container clusters, while minimizing the weighted-distance that material travels by optimizing the allocation of containers to dock-doors.

The MRCDSP is the first attempt to consider all of these aspects of cross-dock operations hierarchically, taking into account available resources at the cross-dock. Further, it is the first work to provide a holistic approach for these cross-dock operating decisions. Using this framework achieves an optimal container to
dock-door allocation, and a makespan minimizing schedule of containers to the cross-dock. A comparative numeric study based on test case data from a retail partner shows that our approach reduces, on average, the makespan by 37% and the weighted-distance material travels metric by 45% in comparison to methods commonly used in practice.

The remainder of the chapter is organized as follows. In Section 4.2 we provide an overview of the literature relevant to our work. In Section 4.3 we describe our modeling approach in detail. We present results from a numeric study in Section 4.4, and we conclude with a brief summary in Section 4.5.

### 4.2 Review of Related Literature

Boysen and Fliedner (2010), Vogt (2010), and Van Belle et al. (2012) provide recent surveys of the cross-docking literature. Our work targets a gap in the literature.
mentioned by each of these survey papers: the need to simultaneously address the issues of material flow and container scheduling at cross-docks. As our research is related to both the material flow and scheduling decisions, we briefly discuss relevant papers from the cross-docking literature. Specifically we focus on papers regarding material flow and handling performance at cross-docks, and those targeting operational issues related to the scheduling of containers to be processed at the cross-dock.

Early papers related to cross-dock management include the seminal papers by Tsui and Chang (1990, 1992) who formulate the problem of assigning containers to dock-doors as a quadratic assignment problem (QAP) and suggest a branch-and-bound approach for solving this problem. Gue (1999) relaxes the assumption that at an I shaped cross-dock all inbound containers are assigned along one side of the cross-dock, opposite to the outbound containers. By relaxing this assumption, Gue achieves a significant reduction of the weighted-distance material traveled within the cross-dock, in comparison with previous models. Miao et al. (2009) extend Gue’s work by relaxing the assumption that dock-doors are fixed for specified destinations; the authors also account for a notion of scheduling, as we describe below.

In the aforementioned papers, only very small instances of the QAP are solved to optimality. This is due to the computational complexity of the problem. Therefore, many later papers offer heuristic solution approaches. Bartholdi and Gue (2000) use a simulated annealing heuristic to investigate the effects that incorporating congestion and labor costs within the door assignment problem has on dock-door allocation, arguing that considering the complete cost structure might lead to a reduction in total cost. Bozer and Carlo (2008) present a simulated annealing heuristic to solve the QAP in a cross-dock setting, assuming that the number of containers is exactly equal to the number of dock-doors. Rosales et al. (2009) study the allocation of dock-doors to containers while balancing workload among the dock-doors. They focus on minimizing the total cost, accounting for material flow and workforce resources, under the assumption that a single person is assigned to each cross-dock door with the possibility of overtime and that each outbound shipment location is both fixed and exogenous.

In our study, we use a computationally efficient formulation of the QAP, coupled with a reduction in problem size (achieved via clustering of containers), to ob-
tain an optimal solution. The idea of container clustering in cross-dock operations is not new; it has been noted in both Li et al. (2008) and Shakeri (2011), where the focus is on finding a feasible schedule of containers subject to resource constraints. Rosales et al. (2009) use a similar problem size reduction technique where they divide their inbound containers sets into smaller subsets when the problem size is too large. In our work, we reduce the problem size by assigning containers to disjoint clusters that do not share material flows, thereby enabling an optimal solution to the resulting QAPs.

Another stream of research examines scheduling issues in the context of cross-docks. Specifically, many papers focus on the scheduling of containers for processing, subject to resource constraints, with the objective of optimizing some measure of interest. The problem of scheduling containers and tasks in a resource-constrained environment is NP-hard, as was shown by Li et al. (2004), Boysen et al. (2010), and others. Due to its complexity, most papers in this area rely on heuristic solution approaches, c.f. Boysen and Fliedner (2010).

Chen and Lee (2009) analyze task scheduling in a cross-docking environment. They assume that the cross-dock is set up so that all inbound containers are on one side and outbound containers are on the other side. The authors further assume that only one task can be performed on each side of the cross-dock per time period, taking into account precedence requirements. They show that even at this level of abstraction, the problem is NP-hard, which demonstrates the complexity of the real-world problem being analyzed. Wang and Regan (2008) analyze a variety of scenarios, cross-dock layouts, and performance measures, by simulating rules-of-thumb regarding the scheduling of containers to dock-doors. They conclude that the currently-accepted practice of first-come-first-served policy, or even a simple look-ahead policy, might perform poorly when trying to minimize makespan, compared to other policies that account for processing time a priori.

Choi et al. (2006) develop a genetic algorithm to solve a non-linear program with the objective of balancing the workload at the cross-dock. Their model limits inbound container assignment to a single dock-door and they assume freight sorting time is negligible. Zhang et al. (2010b) minimize three objectives simultaneously; the total handling time of inbound freight, the total weighted distance material travels, and the completion time of processing all outbound containers. They formulate
the problem as a multi-objective mixed-integer linear program with a predefined number of outbound and inbound dock-doors and an implicit assumption of abundant labor and machine resources. They solve the problem for an instance of nine inbound containers. The authors conclude that minimizing the total handling time of inbound freight or the completion time of processing all outbound containers does not provide a good solution with respect to the dock door assignment; instead, first solving the QAP and then optimizing the schedule provides superior performance. This is the same format suggested by our MRCDSP, and as such we see similar improvements in performance. However, we are able to solve much larger instances of the problem.

Lee et al. (2009) focus on minimizing the time required to process a given set of containers. They assume unlimited space and resources at the cross-dock and formulate the problem as a mixed-integer linear program in which the processing time accounts for the weighted distance that material travels. They are not able to solve the model to optimality for practical cases due to the complexity of the model. Miao et al. (2009) account for both container scheduling and dock-door allocation simultaneously. However, they assume that the arrival and departure dates are fixed and the issue of scheduling is treated as a constraint rather than as an objective. The authors conclude that there is a need to extend their study to the case where scheduling is addressed as an optimization study rather than as an input. Our MRCDSP extends their work. We note that all of the aforementioned papers resort to heuristic-solution methods, whereas our approach provides optimal solutions for most stages of the MRCDSP, and bounds on solution quality when optimal solutions are not achieved.

4.3 Model Development

The MRCDSP framework provides a makespan-minimizing resource allocation scheme for shift level operations at a cross-dock. This scheme is constructed subject to a container to dock-door assignment that minimizes the weighted distance of material flows. To achieve this goal, our framework utilizes a hierarchical approach composed of four stages. The foundation of our approach is the reduction of the size of the problem through a data-driven clustering process by which containers
are grouped into independent and disjoint sets. We define this container clustering step as Stage 1 of our MRCDSP. For each container cluster, we solve for the dock-door allocation that minimizes the weighted-distance material travels within the cross-dock in Stage 2. We then solve for the makespan-minimizing workflow schedule for each cluster at several predetermined resource levels in Stage 3. Each of these solutions, which maps resource levels to makespans, serves as a possible mode — i.e. a combination of resources with a consequent makespan — for processing this cluster in Stage 4 of the MRCDSP. In Stage 4 we find the makespan-minimizing schedule of clusters to the cross-dock subject to the container to dock-door allocations and the total available resource levels at the cross-dock. In summary, the stages of the MRCDSP are:

**Stage 1:** Container Clustering

**Stage 2:** Dock-Door Assignment

**Stage 3:** Workflow Scheduling

**Stage 4:** Container Scheduling

### 4.3.1 Stage 1: Container Clustering

In Stage 1 of the MRCDSP, we cluster the containers into disjoint sets based on the flow of materials between containers. The identification of the disjoint subsets of containers utilizes a transshipment matrix as input, producing independent clusters which do not share material flow between clusters. The transshipment matrix describes the flow of material between containers, where the element in row $i$ and column $j$ is the weight of material that travels from inbound container $i$ to outbound container $j$. This matrix is known since we assume that we know the loading plan for all inbound and outbound containers, and we know the origin of any item on all outbound containers. Each resulting cluster is a minimal cardinality set of inbound and outbound containers. Figure 4.2 demonstrates a potential clustering of containers. Figure 4.2a shows a set of inbound and outbound containers, where connecting arcs between the containers represent material flows. From this set of containers, we can create the disjoint set of clusters that do not share material flows.
flows between clusters, as shown in Figure 4.2b. Clustering is accomplished using a simple algorithm such as the one shown in Section 4.7. Through clustering we create multiple smaller problems that can then be addressed in the following stages of the MRCDSP.

4.3.2 Stage 2: Dock-Door Assignment

After clustering the containers into disjoint clusters, the MRCDSP proceeds to assign the containers within each cluster, $C_γ$, to dock-doors. This assignment minimizes the weighted rectilinear-distance traveled by material within the cross-dock; this metric is often used in the literature as the main performance measure for cross-dock efficiency e.g. Tsui and Chang (1990, 1992), Gue (1999), and Bartholdi and Gue (2000). The MRCDSP uses a linearized version of the QAP, as first proposed by Zhang et al. (2010a). The linearized model and the smaller problem instances resulting from Stage 1 of the MRCDSP allow us to find an optimal layout solution per cluster within reasonable time using standard optimization methods and solvers.

Model 1 presents the linearized version of the QAP that is used to assign con-
tainers to the dock-doors. This formulation is adopted from the IPQAPR-IV model in Zhang et al. (2010a). The decision variables used in Model 1 are described below; notation definitions are provided in Section 4.6.

\[
x_{ij} = \begin{cases} 
1 & \text{if container } i \text{ is assigned to dock-door } j \\
0 & \text{otherwise.}
\end{cases}
\]

\[
y_{ijkl} = \begin{cases} 
1 & \text{if container } i \text{ is assigned to dock-door } j \text{ and container } k \text{ to dock-door } l \\
0 & \text{otherwise.}
\end{cases}
\]

Model 1:

\[
\min \sum_{1 \leq i < k \leq |C_\gamma|} \sum_{1 \leq j \neq l \leq |D|} \omega_{ikj} \epsilon_{j l} y_{ijkl}
\]  

s.t.

\[
\sum_{j=1,j\neq l}^{[D]} y_{ijkl} = x_{ij} \quad \text{for all } i, k \in C_\gamma, j \in D, i < k, (i,k) \notin F_0
\]  

\[
\sum_{i=1,i\neq j}^{[D]} y_{ijkl} = x_{kl} \quad \text{for all } i, k \in C_\gamma, l \in D, i < k, (i,k) \notin F_0
\]  

\[
y_{ijkl} \in \{0, 1\}
\]  

\[
x_{ij} \in \{0, 1\}
\]  

\[
\forall i \in C_\gamma, \sum_{j=1}^{[C_\gamma]} x_{ij} = 1 \quad \forall j \in D
\]

\[
F_0 = \{(i,k) \in C_\gamma \times C_\gamma | \epsilon_{ik} \eta_k = 0\}
\]

The efficiency of Model 1 depends on the size of \( F_0 \), the set of all containers not connected through material flow. Generally, the larger the size of \( F_0 \) relative to \( |C_\gamma| \), the cardinality of the container set of a cluster, the faster Model 1 is solved to optimality. Furthermore, we solve this QAP for each cluster as if it is the only cluster to be allocated to the cross-dock, hence formulation (1.1)-(1.5) is equivalent to assigning the cluster of containers to a cross-dock with \( |C_\gamma| \) dock-doors on each side. This does not change the formulation above, however it does reduce the problem size as \( |D| \) need not be the entire set of doors, but rather a restricted portion of this cross-dock. Objective function (1.1) minimizes the weighted distance of the flow of materials from the inbound to the outbound containers within a cluster. Constraints (1.2) and (1.3) require that each dock-door is allocated to a single container, and that each container is allocated to a single dock-door, respectively. Constraints (1.4) and (1.5) ensure that the variables are binary. Model 1 computes the optimal allocation of containers to the dock-doors within a cross-dock. These allocations become constraints in Stage 4 of the MRCDSP - the container schedul-
ing phase - ensuring that we maintain optimal assignments.

4.3.3 Stage 3: Workflow Scheduling

The purpose of this stage is to create the processing time data for multiple execution modes required by Stage 4 of the MRCDSP — the container scheduling stage. Contrary to previous research related to scheduling at cross-docks, which typically assume that the processing time for all containers is constant regardless of the actual resource allocations (c.f. Boysen and Fliedner (2010), Bartholdi and Hackman (2008), and Bartholdi and Gue (2000)), this stage of the MRCDSP finds the makespan-minimizing schedule of the processing of tasks for each cluster subject to resource availability and precedence constraints. In order to provide a more appropriate estimate of processing times, the MRCDSP calculates the processing times for container clusters subject to resource availability as described in Model 2.

Model 2 is an instance of the Multi-mode Resource Constrained Project Scheduling Problem (MRCPSP), c.f. De Reyck and Herroelen (1999), adapted to our cross-dock setting. In our model, each task (unloading, sorting and loading) can be carried out by a single resource type (either labor or machine) which is predetermined based on the nature of the freight associated with this task (e.g. dimensions, weight, packaging, whether the freight is palletized, etc.). In our setting, we allow for the task processing types: (1) regular processing, available to both resource types; and (2) “crash” processing, available only for tasks associated with labor resources. “Crash” processing requires two laborers to execute the task, rather than one. The complete workflow scheduling model is given below in Model 2. The decision variables used in Model 2 are detailed below. Additional notation is defined in Section 4.6.

\[
\begin{align*}
    w_{nt} &= \begin{cases} 
        1 & \text{if job } n \text{ starts at time } t; \\
        0 & \text{otherwise.} 
    \end{cases} \\
    z_{nt} &= \text{number of resource units required for job } n \text{ at time } t; \\
    q_n &= \begin{cases} 
        1 & \text{if two workers are assigned to job } n \text{ where } n \text{ is “crash” processed;} \\
        0 & \text{otherwise.} 
    \end{cases} \\
    \tau &= \text{makespan.}
\end{align*}
\]
Model 2:

\[
\min \tau \\
\tau \geq \sum_t (t + \theta_n) \cdot w_{nt} - \sigma_n q_n \\
\sum_t w_{nt} = 1 \\
\sum_t (t + \theta_n) \cdot w_{nt} - \sigma_n q_n \leq \sum_t w_{nt} \cdot t \\
\sum_{n \in N_g} z_{nt} \leq \alpha_g \\
z_{nt} \geq \sum_{t'=t-\theta_n+1}^{t} w_{nt'} - q_n \\
z_{nt} \geq 2 \sum_{t'=t-\theta_n+\sigma_n+1}^{t} w_{nt'} + q_n - 1 \\
\sum_{n \in \{N_c \cap N_g\}} z_{nt} \leq U_{cg} \\
w_{nt} \in \{0, 1\} \\
q_n \in \{0, 1\} \\
z_{nt} \in \mathbb{Z} 
\]

for all \( n \in N \) \hspace{1cm} (2.1) 
for all \( n \in N \) \hspace{1cm} (2.2) 
for all \( n \in N \) \hspace{1cm} (2.3) 
for all \( (n, n') \text{where } n \text{ is a prerequisite of } n' \) \hspace{1cm} (2.4) 
for all \( t \in T, \ g \in G \) \hspace{1cm} (2.5) 
for all \( n \in N, \ t \in T \) \hspace{1cm} (2.6) 
for all \( n \in N, \ t \in T \) \hspace{1cm} (2.7) 
for all \( t \in T, \ g \in G, \ c \in C \) \hspace{1cm} (2.8) 
for all \( n \in N, t \) \hspace{1cm} (2.9) 
for all \( n \in N \) \hspace{1cm} (2.10) 
for all \( n \in N, t \) \hspace{1cm} (2.11)

The objective function, (2.1), minimizes the makespan which is defined in (2.2). Constraint (2.3) ensures that each task is assigned to be executed exactly once. Constraint (2.4) ensures that the precedence relationships are satisfied. Constraint (2.5) requires that sufficient resources are available for execution of all tasks being processed at each time unit \( t \) (i.e. the resource constraint). Constraints (2.6) and (2.7) define the processing time associated with each task, depending on its processing method (i.e. “crash” or regular processing). Constraint (2.8) enforces a maximum number of resources that can be simultaneously assigned to work on tasks in the same container (due to space limitations, safety requirements, etc.). Constraints (2.9) and (2.10) restrict variables \( w_{nt} \) and \( q_n \) to be binary; constraint (2.11) allows \( z_{nt} \) to be any non-negative integer.

We note that Model 2 is NP-hard, as shown by, among others, Blazewicz et al. (1986). Because multiple instances of this model need to be solved for each cluster, each serving as input for Stage 4 of the MRCDSP, we present a heuristic algorithm to obtain a solution to Model 2. The objective for Stage 3, and more specifically for the employment of our heuristic, is to generate reasonable and relevant processing times, which are used as the processing times associated with the different resource levels denoted as modes in Stage 4. We note that if exogenous processing times for clusters given a set of resources are available they may be used as input to Stage 4.
of the MRCDSP, eliminating the need for Stage 3.

Our heuristic, as with the model formulation above, assumes the availability of the following information from the loading plans of the inbound and outbound containers: (1) The set of tasks that need to be processed; (2) processing times for each task; and (3) the precedence relationships between tasks; (4) the total available resource levels per shift.

Because we assume that the freight locations within the container, dimensions and weight specifications are known, then the resource requirements for each task can be determined. Once resource requirements are determined, processing times are calculated based on the type of resource associated with each task. We assume that only tasks that are processed by resource type labor may be “crashed” to reduce the time needed for material handling.

Precedence relationships between tasks are provided by detailed container packing plans. We assume that once unloading of freight has started for a particular container, it must be completed before sorting of its freight can begin. For sorting, tasks are divided into three groups corresponding to their location in the outbound container: Front, middle, and back. This grouping allows us to add a dummy task which serves as a unique predecessor of numerous sorting and loading tasks.1 The division into three groups for sorting tasks stems from safety issues regarding container packing to ensure proper weight distribution; however, in general any number of groups may be used in our algorithm.

In describing our heuristic, we use the following notation for container types based on the container contents. Type-1 containers hold freight that require both machine and labor resources; Type-2 containers require only laborers; and Type-3 containers can be processed entirely by machine resources. For each container, each task is associated with a pre-defined processing time determined by the resource associated with processing. The total processing time for all tasks on a

1For example, consider a single outbound container, and two inbound containers. Once all freight that is intended to be loaded into the front of the outbound container has been unloaded from the inbound containers, a dummy task will be marked as completed, and hence the sorting of this freight can begin. Once all sorting of this freight has been completed, a dummy task will be marked as completed, and loading of the freight can begin. For the freight bound for the back of the outbound container, the dummy task will need to account for the completion of loading the middle of the outbound container, before it can be marked as completed, noting that for loading of the middle portion the front portion must first complete loading.
container depends on the allocation of resources to that container.

We propose the following heuristic algorithm to compute the processing makespan. Assign the maximum available resources to the set of inbound Type-1 containers. This resource allocation must account for the predefined upper limit on the level of resources that can work simultaneously on a single container (see constraint (3.8)). After allocating resources to Type-1 containers, allocate any labor resources still available to inbound Type-2 containers, and any machine resources still available to inbound Type-3 containers. This allocation process continues until the resources or set of relevant jobs are exhausted. At each time period, $t$, the level of unassigned resources and remaining jobs on hand is updated.

Upon completion of all unloading tasks, we turn to the sorting tasks. The process of allocating resources to tasks is such that all available resources are assigned sorting tasks associated with freight that is to be loaded onto the front one-third of the container. Upon completion of sorting for the front group, resources are allocated to the freight associated with the middle one-third of the container, and then to the back one-third. Finally, all resources are assigned to loading tasks until all containers have been loaded.

This algorithm is described by the following pseudocode. Additional notation is defined in Section 4.6.
Algorithm 2: Computing Minimal Makespan.

Inputs: resource availability levels; tasks yet to be processed; predecessor relationship matrix; processing times for each task.

Step 0. Initialize: Set time \( t = 1 \), and define the following lists:
- \( RA(t) \) – Resources Available at time \( t \) - initially all resources allocated to the cluster;
- \( JR(t) \) – Jobs Remaining at time \( t \) - initially all tasks in the cluster;
- \( JIP(t) \)– Jobs in Progress at time \( t \);
- \( PRM \) – Predecessor Relationship Matrix;

Step 1. Resource Allocation: While \( (JR(t) \neq \emptyset \) and \( JIP(t) \neq \emptyset) \)

Step 1.a. Unloading jobs in \( JR(t) \) associated with Type—1 containers:
- If \( Ucg < RA(t) \): assign maximum levels of machines and labor subject to \( Ucg, PRM \);
- Else: assign resources proportionately to job types on containers, s.t. \( RA(t) \) and \( PRM \);
- Update \( JR(t), JIP(t), \) and \( RA(t) \), based on resource allocation and processing times.

Step 1.b. Unloading jobs in \( JR(t) \) associated with Type—2 and Type—3 containers:
- Assign maximum levels of resources s.t. \( Ucg, RA(t), \) and \( PRM \).
- Update \( JR(t), JIP(t), \) and \( RA(t) \), based on resource allocation and processing times.

Step 1.c. Sorting jobs in \( JR(t) \):
- Assign maximum levels of resources s.t. \( RA(t) \) and \( PRM \).
- Update \( JR(t), JIP(t), \) and \( RA(t) \) based on resource allocation and processing times.

Step 1.d. Loading jobs in \( JR(t) \):
- Assign maximum levels of machines and labor s.t. \( RA(t) \) and \( PRM \).
- Update \( JR(t), JIP(t), \) and \( RA(t) \) based on resource allocation and processing times.

Step 2. Time update:
- \( t = t + 1 \).
- Go to Step 1 with updated lists.

Step 3. End While.

Output: \((t - 1)\), the makespan for a cluster, given a set of resources.
Algorithm 2 can be used to solve realistically-sized problems within seconds. We have solved instances of simulated data consisting of over a hundred containers within seconds. Furthermore, we are able to bound the gap between the optimal solution and the solution generated by this algorithm. This bound tends toward zero as the number of tasks increases.

To construct our bound, we recall the following assumptions: once resources are allocated to unloading a container they remain assigned until the unloading process of that container has been completed; only tasks carried out by labor can be crashed; all precedence relationships are known; and each unit of freight passes through a path of unloading, sorting, and loading. We recall the notation, \( g = \{\text{Labor}, \text{Machine}\} \), is the number of resources of each type, available to us. We also know the number of tasks per container \( c \), denoted \( N_c \), and the total number of tasks to be undertaken using labor and machine resources, denoted \( N_{\text{Labor}} \) and \( N_{\text{Machine}} \).

We assume that tasks carried out by labor in regular mode require \( k \cdot t \) time units, where \( k > 1 \), and when carried out in “crash” processing, tasks require \( t \) time units to complete.

The worst-case solution for Algorithm 2 occurs when a single available laborer is assigned to unload the very last fully-loaded, Type-2 container, denoted \( c^* \), which contains jobs that need to be loaded in the front segment of all outbound containers. An upper bound on the optimality gap is the difference between the time required to unload and sort the task in this Type-2 container, and the time needed to sort all other tasks after which the resources are idle. The bound on the optimality gap is therefore,

\[
\text{Max} \left\{ 2 \cdot k \cdot |N_{c^*}| \cdot t - \max \left\{ \left( \frac{N_{\text{Labor}} - N_{\text{Labor}} \cap N_{c^*}}{N_{\text{Labor}}} \right) \cdot \frac{N_{\text{Machine}}}{\alpha_{\text{Machine}}} \cdot t \right\}, 0 \right\}.
\]

The first term in the expression, \( 2 \cdot k \cdot |N_{c^*}| \cdot t \), is the amount of time required for unloading and sorting of all freight on container \( N_{c^*} \). The term \( \left( \frac{N_{\text{Labor}} - N_{\text{Labor}} \cap N_{c^*}}{N_{\text{Labor}}} \right) \cdot \frac{N_{\text{Machine}}}{\alpha_{\text{Machine}}} \cdot t \) is the minimal amount of time required by labor resources for processing (sorting and loading) all freight not on container \( N_{c^*} \) after unloading of this container has commenced. Finally, the term \( \frac{N_{\text{Machine}}}{\alpha_{\text{Machine}}} \cdot t \) is the amount of time required by machine resources to complete processing of freight. For the case in which the number
of tasks is much larger than the available set of resources, i.e. \( N_g \gg \alpha_g \) for \( g = \{\text{Labor, Machine}\} \), then our bound tends toward zero. Recall that each unit of freight is associated with three tasks (unloading, sorting and loading), and therefore \( N_g \) will often be significantly larger than \( \alpha_g \) in practice, unless we are dealing with an instance containing very few containers in the cluster, or one where there are very few units of freight on each container.

### 4.3.4 Stage 4: Container Scheduling

The final stage of the MRCDSP creates a makespan-minimizing schedule of container clusters in the cross-dock subject to resource constraints. Stage 4 uses the results from Stages 2 and 3 as inputs to find an optimal makespan-minimizing schedule of containers to the cross-dock. The final schedule allocates resources — labor, machines, and dock-doors — to particular container clusters over the planning horizon. The decision variables for Stage 4 are as follows.

\[
\begin{align*}
  k_{\gamma m d t} &= \begin{cases} 
    1 & \text{if cluster } \gamma \in \Gamma \text{ starts in mode } m \in M \text{ at door } d \in D \text{ in time } t \in T; \\
    0 & \text{otherwise;}
  \end{cases} \\
  \tau &= \text{makespan.}
\end{align*}
\]

**Model 3:**

\[
\begin{align*}
  \text{min } \tau \\
  \tau &\geq \sum_m \sum_d \sum_t \left( t + s_{\gamma m} \right) \cdot k_{\gamma m d t} \quad \text{for all } \gamma \in \Gamma \\
  \sum_m \sum_{d=1}^{\Delta-\delta+1} \sum_t k_{\gamma m d t} &\leq \sum_{t=0}^{T} \left( t + s_{\gamma m} \right) \cdot k_{\gamma m d t} \quad \text{for all } \gamma \in \Gamma \\
  \sum_\gamma \sum_m \sum_{d=1}^{\Delta-\delta+1} \sum_{t=1}^{T} k_{\gamma m d t} &\leq \alpha_g \quad \text{for all } t, g \\
  \sum_\gamma \sum_m \sum_{d=1}^{\Delta-\delta+1} \sum_{t=1}^{T} k_{\gamma m d t} &\leq 1 \quad \text{for all } t, d \in D \\
  k_{\gamma m d t} &\in \{0, 1\} \quad \text{for all } \gamma \in \Gamma, m \in M, d \in D, t
\end{align*}
\]

The objective function (3.1) minimizes the makespan of processing all container clusters. The makespan defined by (3.2) evaluates the total time required to complete processing of all container clusters. Constraint (3.3) ensures that every cluster is processed exactly once. Constraint (3.4) guarantees that sufficient resources are available to the container clusters being processed at a given time. Constraint (3.5) ensures that sufficient contiguous doors are assigned to container clusters, so that the layout achieved in Stage 2 is implemented. Constraint (3.6) requires all decision variables to be binary.
Similar to Model 2, Blazewicz et al. (1986) have shown that this binary-optimization model is NP-hard. However, we are able to solve realistically-sized problems encountered in practice due to reduction in problem size achieved through container clustering. Further, we use two common cutting-plane methods to further reduce the problem size. The first type of cutting plane is generated by solving the model using a weak resource constraint, i.e. large $\alpha_g$ values, which produces a lower bound on the makespan. This type of bound is particularly useful when convergence to optimality is slow, i.e. the integer solution is easily found, but the convergence process is slow. The second type of cutting plane is generated by solving the problem in an iterative fashion, starting with a small subset of container clusters and subsequently adding an additional container cluster until the entire set of container clusters have been scheduled. This type of cutting plane is particularly useful for reducing computer memory requirements. The general idea is to first solve smaller problems, and slowly increase the size of the problem being solved by gradually adding clusters to be scheduled. For each new larger problem being solved we use the result of the previously solved problems as lower bound constraints and reduce the required search space.

4.4 Numerical Study

In order to evaluate the performance of our MRCDSP formulation, we applied our full MRCDSP to a set of simulated data. We first describe the assumptions used in our numerical example as well as the data generating process for our simulated data. We next evaluate Stages 2 and 3 of our MRCDSP method using these simulated data. Finally, we compare the results of our container cluster scheduling stage (Stage 4) with a schedule generated using a random assignment rule, which is commonly employed in practice. All of our numerical results were generated and analyzed using an Intel 3GHz Quad Core PC with 16 GB of RAM. We use GAMS with Gurobi solver and MATLAB to implement our MRCDSP method.

4.4.1 Description of Simulation-Generated Data and Assumptions

Due to confidentiality concerns, we are not able to present results of our MRCDSP method using exact data from the implementation of the method with our industrial
partner. However, our simulated dataset was generated through extensive consultation with our industrial partner and its largest customer to closely imitate real world data. The simulated data are based on approximations of our observations at various cross-dock facilities associated with our industry partner. Using simulated datasets also allows us to create a much larger set of instances for our numerical results: we simulate 1,000 datasets consisting of 100 containers each - a reasonable amount of containers to be processed within a 10 hour shift.

We use the following assumptions when generating our simulated data.

1. **Cross-dock Layout**: We assume an I-shaped cross-dock facility, with evenly spaced doors on each side of the dock (where the distance between any two adjacent doors - both to the sides and exactly across the dock - is normalized to 1 unit of distance in our numerical results).

2. **Container to Dock-Door Assignment**: We assume that any container may be assigned to any dock-door when allocating containers; however, once allocated, dock-doors remain associated with a container until all tasks within the container’s cluster have been processed.

3. **Resources**: We assume that resource levels are known for all sets of resources: dock-doors, labor and machines.

4. **Processing Times of Tasks**: We assume that all tasks (unload, sort, and load) processed by the same type of resource require equal time. Extensive discussion with foremen and shift managers at several cross-dock facilities verified that this assumption is reasonable. For illustrative purposes, we assume that each time unit, $t$, is 5 minutes.

5. **Containers**: Each inbound container has up to 30 units of freight (i.e. pallets) that require processing. Each outbound container has up to 40 units of freight that require processing. (Container sizes typically differ because inbound containers often arrive from sea vessels and outbound containers are transported by truck). Each inbound container has material destined for between two and ten outbound containers, and this value follows a uniform distribution. This results, in Stage 1, in container clusters of five to fifteen
containers per cluster. We assume that units of freight are sorted into three designated segments — front, middle, and back — on outbound containers. For the simulated data, assume that Type-1 containers, requiring both labor and machine resources, have an equal amount of tasks requiring labor and machine resources. Further, we assume, for constraint (2.8), that the maximum number of resources allowed to simultaneously process loading and unloading tasks on a specific container is: two laborers and two machines for Type-1 containers; three laborers for Type-2; and three machines for Type-3 containers.

4.4.2 Evaluating MRCDSP Stage 2: Optimal Dock-Door Assignment

To assess the efficiency of Stage 2 of our approach in being able to intermix inbound and outbound dock-doors, we compare the following scenarios: in Scenario 1 we solve Model 1, as defined in Equations (1.1) to (1.5), where any container can be allocated to any of the dock-doors; in Scenario 2, we solve Model 1 with the additional constraint, frequently used in practice, that inbound containers are assigned to doors along one side of the cross-dock and outbound containers are assigned to the opposite side. Unsurprisingly, the less constrained model solution (Scenario 1) outperforms the solution in Scenario 2 in terms of the weighted distance of material flow objective.

We analyze an I-shaped cross-dock with 20 dock-doors on each side (i.e. a total 40 dock-doors). Our cluster sizes range from five to fifteen containers, and we randomly select twenty-five clusters for each cluster size, from our clustered data. These cluster sizes represent actual cluster sizes based on data from our industrial partner. We solve the linearized version of the QAP for each of these clusters using three different approaches. (1) Using Stage 2 of the MRCDSP, which we refer to here simply as QAP. (2) Using Stage 2 of the MRCDSP with an additional across-the-dock constraint, limiting inbound containers to one side of the cross-dock and the outbound containers to the other side, which we refer to as cross-dock QAP (cdQAP). (3) Using Stage 2 of the MRCDSP while limiting the number of doors to precisely the number of containers in the cluster by allocating a segment of the cross-dock of size $\frac{|C|}{2}$ to the cluster (if $|C|$ is odd we add one dummy trailer); we
Table 4.1: Comparison of QAP, cdQAP, and mdQAP

<table>
<thead>
<tr>
<th>Cluster Size</th>
<th>Average Gap Between QAP and cdQAP</th>
<th>Average Gap Between QAP and mdQAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>45.57%</td>
<td>0.00%</td>
</tr>
<tr>
<td>6</td>
<td>48.46%</td>
<td>0.67%</td>
</tr>
<tr>
<td>7</td>
<td>49.15%</td>
<td>0.00%</td>
</tr>
<tr>
<td>8</td>
<td>44.08%</td>
<td>1.75%</td>
</tr>
<tr>
<td>9</td>
<td>45.14%</td>
<td>0.68%</td>
</tr>
<tr>
<td>10</td>
<td>40.87%</td>
<td>1.47%</td>
</tr>
<tr>
<td>11</td>
<td>40.15%</td>
<td>0.77%</td>
</tr>
<tr>
<td>12</td>
<td>42.25%</td>
<td>1.19%</td>
</tr>
<tr>
<td>13</td>
<td>43.12%</td>
<td>0.83%</td>
</tr>
<tr>
<td>14</td>
<td>42.93%</td>
<td>1.21%</td>
</tr>
<tr>
<td>15</td>
<td>43.71%</td>
<td>0.89%</td>
</tr>
<tr>
<td>All Clusters</td>
<td>44.13%</td>
<td>0.86%</td>
</tr>
</tbody>
</table>

refer to this third approach as the minimal door QAP (mdQAP). The results are reported in Table 4.1.

From Table 4.1, we see that the QAP outperforms the cdQAP by approximately 45% on average. This implies that the current practice in which all inbound containers are located on one side of the cross-dock and all outbound containers on the opposite side increases material flows substantially compared to the QAP, which allows containers to be allocated to either side of the cross-dock. Furthermore, we see that the mdQAP achieves a weighted distance material flow value that is on average only 0.86% worse than the QAP. This implies that it is sufficient to allow a cluster to be allocated to the cross-dock in a fashion that it is only allocated a contiguous number of doors (facing each-other), rather than allowing for all possible optimal allocations of containers to dock-doors. This allows us to achieve a close-to-optimal allocation, which utilizes contiguous dock-doors, which is important, as dock-doors may be a bottleneck resource, and in that case we would like to refrain from leaving these dock-doors idle.2 Recall that we solved the QAP in Stage 2 of the MRCDSP for a number of dock-doors equal twice the number of containers in the cluster. Our results here suggest that we can reduce the problem

2It is easy to see that in any optimal allocation of containers to dock-doors, there may be a maximum of 33% unused dock-doors.
size, using a number of dock-doors equal to the number of containers in the cluster, and solve the mdQAP without significantly increasing the weighted-distance material travels.

4.4.3 Evaluating MRCDSP Stage 3: Workflow Scheduling

In this section, we analyze the efficiency of our heuristic approach for solving the makespan-minimizing work schedule within clusters in Stage 3 of our MRCDSP. We analyze the efficiency of Stage 3 by comparing our solution to an alternative approach commonly used in practice in which tasks are randomly assigned to resources.

The random assignment benchmark we use is as follows. At each time unit, all available tasks, regardless of their container of origin, but accounting for predecessor relationships, have an equal probability of being assigned a resource for processing. In our implementation of a random allocation, we account, as in the MRCDSP, for the predefined upper limit on the level of resources that can work simultaneously on a single container. We note that the benchmark used here is a rather optimistic measure compared with another commonly used policy where assignments are made on first-come-first-served basis. The random assignment method inherently assumes that containers are clustered, which may not be the case in reality, where a container may be unloaded yet its freight cannot be loaded onto an outbound container because their predecessors have yet to be allocated to the cross-dock. In order to compare our heuristic to the benchmark we need to: select different resource levels (i.e. modes); compute the makespan for both our heuristic and the random approach; use these results to solve Model 3. The choice of modes used for each cluster analyzed is as follows. We begin by searching for the resource level required to minimize the makespan of the cluster without waste (i.e. the minimum level of resources that allows for a makespan equal to the non-capacitated resource level). Subsequently, the resources are reduced to a minimum such that the cluster will be completed within a reasonable time window (e.g. half the length of a shift). The remaining modes that we use for calculation are chosen at fixed intervals between these two points.

The minimum level of resources that sets a makespan equal to the non-capacitated
case is, in general, not easy to establish. Due to the low computational require-
ments of our heuristic, we are able to gain insight into the relationship between
resource levels and processing times by carrying out an extensive analysis for each
of the clusters. For example, in Figure 4.3, we present a single cluster with 9 in-
bound and 6 outbound containers (containing 160 units of freight that are to be
processed). This cluster is analyzed 1,600 times, with varying levels of resources,
to determine the processing time required for each resource level. The result is a
production-function type relationship such as is shown in Figure 4.3. This rela-
tionship illustrates, as expected, that as additional resources become available, the
processing time decreases. However, Figure 4.3 also suggests that there is a limit
to this improvement. This limit is due to the precedence relationships among dif-
ferent tasks which constrains the number of tasks that are available for processing
per time period, as well as to the constraint on the number of resources that can
simultaneously process tasks on the same container.

Figure 4.3: Cluster Processing Time as a Function of Allocated Resources
Figure 4.4 displays the average efficiency gain (percentage) of the MRCDSP Stage 3 over the random assignment approach. It is generated by comparing the makespans resulting from the two approaches for 1,000 clusters, randomly chosen from the simulated dataset. For each of these clusters we compute the makespan required both under the MRCDSP Stage 3 and the random assignment for ten resource levels. The 100% resource level is, for a specific cluster, the minimum number of resources that achieves the minimal makespan (i.e. the non-capacitated resource case). The other resource levels (i.e. modes) are percentages with respect to this 100% level. The solid line in Figure 4.4 represents the average efficiency gap, which increases as the resource percentage level increases. The dashed lines represent a 95% confidence interval for the efficiency gain. Our results indicate that the MRCDSP achieves an average efficiency gain of 29% for the makespan of tasks within a cluster when compared to the benchmark commonly used in practice of random assignment.

![Figure 4.4: Efficiency Gains - Cluster Makespans: MRCDSP Stage 3 vs. Random-Assignment](image-url)
4.4.4 Evaluating MRCDSP Stage 4: Container Scheduling

Figure 4.5 displays three makespan-minimizing schedules for one illustrative example of our simulated data containing 13 clusters of containers. Figures 4.5a, 4.5b, and 4.5c correspond to different level of resources available at the cross-dock for the duration of the shift. In this example we analyze four potential resource levels (i.e. modes of execution) per cluster. The processing times per mode (resource level) are calculated using the heuristic algorithm presented in Section ???. Mode 1 is defined as the minimum set of resources required to complete the processing of a cluster within half a shift and mode 4 corresponds to the 100% resource level, as described above. Modes 2 and 3 are set respective to modes 1 and 4 at one-third and two-thirds of the gap between these upper and lower resource levels. Figure 4.5a presents the solution assuming 180 laborers and 90 machines are available during the shift. Figure 4.5b presents the result assuming 90 laborers and 45 machines are available. Figure 4.5c presents the result for 60 laborers and 20 machines. Each figure presents a schedule of clusters (denoted $c_1, c_2, c_3, ...$) to the dock-doors, and the mode of processing for the cluster (denoted $m_1, m_2, m_3, m_4$) followed by the associated processing time for the cluster in this mode.

Figure 4.5a shows that all containers are processed in mode 4 which utilizes 100% of the resources for each of the clusters. This is not surprising because labor and machine resource constraints are relatively weak in this scenario of abundant resources. In Figure 4.5b, resources become a binding constraint, hence some container-clusters are executed in modes other than mode 4. Figure 4.5c displays the extreme case of very low resource levels, hence clusters in this setting require substantially longer times to be completed. Comparisons of potential solutions, as shown in Figure 4.5, enable managers to reveal the bottlenecks in the system, whether dock-doors, labor, or machinery, and respond accordingly to adjust the bottleneck, by scheduling appropriate labor and machine levels.

After presenting an illustrative example for a specific data set, we turn to evaluating the expected improvements that may occur by implementing Stage 4 of the MRCDSP. We compare the makespan required using the MRCDSP Stage 4 with that generated by a random assignment of resources to container clusters, and report the average difference between the two solutions.
The random assignment is such that a non-processed cluster of containers is randomly chosen, and all available resources are allocated to it, processing the cluster in the mode resulting in the quickest processing time subject to the level of available resources. Resources are allocated iteratively, to randomly selected clusters of containers, until cross-dock resources are exhausted. Once resources complete the processing of a cluster, they are released back to be used for the remaining clusters. This procedure is repeated until all such clusters have completed processing.

For purposes of comparison, we use the simulated container data set. Initially,
layouts are found, using the linearized QAP formulation which identifies the number of doors required by each of the clusters. Next we apply Stage 3 of the MRCDSP to generate four modes of processing for each container cluster (in the same manner carried out in the example above). Finally, we solve for the minimum makespan of the clusters of containers subject to a number of different levels of resources available at the cross-dock using both MRCDSP Stage 4 and the random approach. This was carried out for each of our 1,000 simulated data sets, solving the problem using 121 combinations of resource and door levels for each data set (11 levels of available resources at the cross-dock levels and 11 levels of available doors). For each data set the corresponding minimum number of doors is equal to the largest number of doors required by all clusters within this set. Similarly, the maximum number of doors is equal to the sum of the number of doors required by all clusters in the data set. We then divide the range between these two values equally to provide us with 11 levels of available dock-doors. We relate 11 levels of resources in a similar manner, defining the 100% level for resources as the one associated with the sum of resources required to process, simultaneously, all container clusters at a resource level (mode) that provides the shortest makespan. Similarly, the minimal resource level is such that it enables the processing of any container cluster.

Our results indicate that the MRCDSP Stage 4 provides a makespan that is, on average, 11% shorter than that provided using a random allocation of resources. The most significant impact, over 20% average improvement in makespans, is achieved for moderate-sized cross-docks and moderate resource levels. The comparison, displayed in Figure 4.6, shows that scenarios in which the cross-dock is not resource constrained with respect to either dock-doors or labor and machinery resources, the random assignment heuristic performs as well as the optimal solution. This is also true for settings where the cross-dock is extremely resource constrained. If the system is extremely constrained, the container clusters must be processed in a sequential manner and available processing modes are limited.
4.5 Conclusions

In this research we present a novel approach, the MRCDSP, to address managerial decisions regarding cross-dock operations. Specifically, our approach results in an optimal makespan-minimizing schedule of containers to doors at a cross-dock subject to resource availabilities. Simultaneously, our approach minimizes the weighted distance that material must travel in the cross-dock and allows for resource-allocation dependent freight processing times.

We exploit the availability of predefined container loading plans to decompose the overall problem into smaller problem instances by clustering containers into disjoint sets which are not connected by the flow of materials between clusters. For each identified container cluster, we then solve for an optimal container to door allocation which minimizes the weighted distance that material travels within the cluster. By reducing the overall size of this combinatorially hard problem, we are able to solve the smaller instances to optimality. We then heuristically estimate the processing time required for each container cluster, based on the amount of resources allocated to the cluster. We show that our heuristic provides reasonable
estimates of the optimal required processing time for each cluster. Next, we de-
velop a container-sequencing schedule that assigns resources to container clusters
in order to minimize the required makespan. Again, through our use of clustering
to reduce the problem size, we are able to obtain optimal solutions.

We evaluate the benefits achievable from the MRCDSP by comparing it to com-
mon cross-dock management practices used in industry. Using a simulated data set
based on our industry partner’s actual cross-dock facility operations, we show that
our MRCDSP method outperforms current practice resulting in an average reduc-
tion in makespan of 37% and an average reduction in the weighted distance mate-
rial travels of 45%. Once implemented, the MRCDSP method provides managers
with an ability to conduct sensitivity analysis regarding the effects of resource lev-
els at the cross-dock on the overall schedule; this can help managers understand
the impact of resource level assignments on overall cross-dock operations. Our
MRCDSP method can be implemented using software such as MS-Excel, Matlab,
and dedicated optimization software such as Gurobi or CPLEX.

We close by noting a future research possibility. During our visits to cross-
dock facilities, as part of this project, we observed that some of these facilities
offer a warehousing and storage service to complement their cross-dock services.
Further investigation suggests that this practice is not uncommon; a recent study
by the Saddle Creek Corporation (2011) reports that over 20% of firms that use
cross-docks store their freight at the cross-dock facility for more than two days.
Therefore, we find some motivation in future exploration of the incorporation of
warehousing and storage at a cross-dock into the MRCDSP framework.
4.6 Model Notation

<table>
<thead>
<tr>
<th>Notation</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>set of all containers</td>
</tr>
<tr>
<td>D</td>
<td>set of doors</td>
</tr>
<tr>
<td>Γ</td>
<td>set of clusters</td>
</tr>
<tr>
<td>(C_\gamma)</td>
<td>set of all containers in a cluster (\gamma \in \Gamma)</td>
</tr>
<tr>
<td>N</td>
<td>set of all tasks/jobs to be processed (unloading, sorting, loading)</td>
</tr>
<tr>
<td>(N_c)</td>
<td>set of jobs belonging to container (c \in C) where (N_c \subseteq N) and (\cup_c N_c = N)</td>
</tr>
<tr>
<td>G</td>
<td>set of all resource types (clamp, forklift, worker/labor)</td>
</tr>
<tr>
<td>(N_g)</td>
<td>set of jobs handled by resource type (g \in G) where (N_g \subseteq N) and (\cup_g N_g = N)</td>
</tr>
<tr>
<td>M</td>
<td>set of possible modes in which a cluster may be processed</td>
</tr>
<tr>
<td>T</td>
<td>set of discrete time slots, (t = 1, \ldots, T)</td>
</tr>
<tr>
<td>(F_0)</td>
<td>set of all unconnected containers in a cluster</td>
</tr>
<tr>
<td>(A)</td>
<td>set of all inbound containers, (A \subset C)</td>
</tr>
<tr>
<td>(B)</td>
<td>set of outbound containers, (B \subset C)</td>
</tr>
<tr>
<td>(A(\gamma))</td>
<td>set of inbound containers in cluster (\gamma \in \Gamma), (A(\gamma) \subseteq A)</td>
</tr>
<tr>
<td>(B(\gamma))</td>
<td>set of outbound containers in cluster (\gamma \in \Gamma), (B(\gamma) \subseteq B)</td>
</tr>
<tr>
<td>(\sigma_{ik})</td>
<td>amount of material to be transferred from container (i) to container (k), ((i, k) \in C)</td>
</tr>
<tr>
<td>W</td>
<td>matrix of (\sigma_{ik}) values</td>
</tr>
<tr>
<td>(\varepsilon_{jl})</td>
<td>rectilinear distance from door (j) to door (l), ((j, l) \in D)</td>
</tr>
<tr>
<td>(\rho_{mg})</td>
<td>number of type (g \in G) resource units used by cluster (\gamma \in \Gamma) in mode (m \in M)</td>
</tr>
<tr>
<td>(s_{pm})</td>
<td>processing time of cluster (\gamma \in \Gamma) in mode (m \in M) (in units of (t \in T))</td>
</tr>
<tr>
<td>(\delta_{\gamma})</td>
<td>number of doors needed for cluster (\gamma)</td>
</tr>
<tr>
<td>(\Delta)</td>
<td>number of doors available at the cross-dock = (</td>
</tr>
<tr>
<td>(\theta_n)</td>
<td>time required to complete job (n \in N) (based on Stage 2 dock-door assignment)</td>
</tr>
<tr>
<td>(\sigma_n)</td>
<td>time saved using extra resource of type worker to complete job (n \in N) (=0 for other resource types)</td>
</tr>
<tr>
<td>(\alpha_g)</td>
<td>available units of resource (g \in G)</td>
</tr>
<tr>
<td>(U_{cg})</td>
<td>limit on number of resources (g \in G) simultaneously utilized at (c \in C)</td>
</tr>
</tbody>
</table>
4.7 Algorithm for Cluster Generation

INPUTS: \( W, \mathcal{A} \) and \( \mathcal{B} \).

Step 0. Initialize: Set cluster counter \( \gamma = 0 \).

Define the set \( Q \) as an interim set aiding in the construction of the clusters;

Step 1. Cluster initialization:
If \( \mathcal{A} \neq \emptyset \)
\[ \gamma = \gamma + 1; \]
\[ B(\gamma) = \mathcal{B}_{\min} \text{ (where } \mathcal{B}_{\min} \text{ is the element of } \mathcal{B} \text{ with the lowest index yet to be chosen)}; \]
\[ Q = \emptyset; \]
Go to step 2.

Else, End. At this stage the algorithm returns all clusters.

Step 2. Cluster Construction:
If \( Q \neq B(\gamma) \)
\[ Q = B(\gamma); \]
\[ A(\gamma) = A(\gamma) \cup \{ \text{All inbound containers in } \mathcal{A} \text{ connected to } B(\gamma) \}, w_{ij} > 0; \]
\[ B(\gamma) = B(\gamma) \cup \{ \text{All outbound containers in } \mathcal{B} \text{ connected to } A(\gamma) \}, w_{ij} > 0; \]
\[ \mathcal{B} = \mathcal{B} / B(\gamma); \]
\[ \mathcal{A} = \{ \mathcal{A} / A(\gamma) \}; \]
Go to Step 2.

Else, Go to Step 1.

OUTPUTS: \( A(\gamma), B(\gamma), \) disjoint clusters of containers \( \forall \gamma \in \Gamma \).


