OPTIMAL SUPPLY CHAIN PLANNING PROBLEMS WITH NONLINEAR REVENUE AND COST FUNCTIONS

By

SEMRA AĞRALI

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2009

To my husband, my parents and my sisters,

who have always encouraged me to pursue my dreams

ACKNOWLEDGMENTS

First, I would like to thank my advisor, Dr. Joe Geunes, for all his help, guidance and continuous support. He paid attention to every step of this dissertation and helped me improve my writing. I feel fortunate to have him as my advisor.

I would like to acknowledge Dr. Cole Smith, Dr. Edwin Romeijn and Dr. Siva Srinivasan for participating in my dissertation committee and their insightful comments. I also thank my recommendation letter writers, Dr. Edwin Romeijn, Dr. Cole Smith and Dr. Joe Hartman. Special thanks go to Dr. Cole Smith for his continuous support and invaluable advice during my job search process.

I would like to acknowledge my friends who, directly or indirectly, helped me. I especially thank Gonca Yıldırım for being such a wonderful friend. She listened to all my complaints and gave the support that I needed. I will miss her a lot. I thank Fadime Üney Yüksektepe for her invaluable friendship and all her help especially during my job search process. I would like to thank my dear colleague, Chase Rainwater for not only being a great officemate but also being a great friend. I also thank Candace Rainwater for the songs that she sang for us. I will miss our "AmeriTurks" band.

Moreover, I thank my parents, Nilüfer and İlhan, and my sisters, İlknur, Semanur and Gül for always believing in me and for their support at every step of my life. I am very lucky to be part of such a wonderful family. I could not have done anything without their patience, encouragement and love.

Last, but not least, I thank my beloved husband, Caner. He is my best friend, a great colleague, and the one that I admire most. I thank him for giving me brilliant ideas when I feel lost in my research, debugging my code with me and everything that he has brought to my life. Without his support and encouragement, this dissertation would not be complete.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

OPTIMAL SUPPLY CHAIN PLANNING PROBLEMS WITH NONLINEAR REVENUE AND COST FUNCTIONS

By

Semra Ağralı

August 2009

Chair: Joseph Geunes
Major: Industrial and Systems Engineering

This dissertation studies problems arising in certain stages of a supply chain. We specifically focus on problems that have nonlinearity in revenue or cost functions, and problems that can be written as mixed-integer linear programming problems. There are four main chapters that provide contributions to the supply chain operations literature.

We first consider the allocation of a limited budget to a set of investments in order to maximize net return from investment. In a number of practical contexts, the net return from investment in an activity is effectively modeled using an S-Curve, where increasing returns to scale exist at small investment levels, and decreasing returns to scale occur at high investment levels. We formulate the problem as a knapsack problem with S-Curve return functions and demonstrate that it is $\mathcal{NP}$-Hard. We provide a pseudo-polynomial time algorithm for the integer variable version of the problem, and develop efficient solution methods for special cases of the problem. We also discuss a fully-polynomial-time approximation algorithm for the integer variable version of the problem. Then, we consider a stochastic knapsack problem with random item weights that follow a Poisson distribution. We assume that a penalty cost is incurred when the sum of realized weights exceeds capacity. Our aim is to select the items that maximize expected profit. We provide an effective solution method and illustrate the advantages of this approach.

We then consider a supply chain setting where a set of customers with a single product are assigned to multiple uncapacitated facilities. The majority of literature on

such problems requires assigning all of any given customer's demand to a single facility. While this single-sourcing strategy is optimal under certain cost structures, it will often be suboptimal under the nonlinear costs that arise in the presence of safety stock costs. Our primary goal is to characterize the incremental costs that result from a single-sourcing strategy. We propose a general model that uses a cardinality constraint on the number of supply facilities that may serve a customer. The result is a complex mixed-integer nonlinear programming problem. We provide a generalized Benders decomposition algorithm to solve the model. Computational results for the model permit characterizing the costs that arise from a single-sourcing strategy.

Finally, we consider a multi-period component procurement-planning and product-line design problem with product substitutions and multiple customer segments. Each customer segment has a preferred product and a set of alternative products. If a customer's preferred product is not made available, demand can be satisfied using an alternative product at a substitution cost. We assume each product is assembled-to-order from a set of components, and inventory is held at the component level. Our aim is to determine a product portfolio, substitution plan, and procurement plan in order to maximize profit. We develop a large-scale mixed-integer linear programming formulation, prove that the problem is $\mathcal{NP}$-Hard and propose a Benders decomposition-based exact algorithm. We conclude the dissertation by discussing our contributions to the literature, and provide some future research directions based on our results.

CHAPTER 1
INTRODUCTION

In today's competitive business environment, companies that use their supply chain systems effectively gain economic advantages over competitors. The primary purpose of a supply chain is to satisfy customer needs while maximizing the overall value generated. The value generated by a supply chain is measured by the difference between the final product value and the overall cost incurred across the supply chain.

At each stage of a supply chain two metrics are used to assess the overall value generated: revenue obtained and cost incurred. These metrics usually cannot be expressed as linear functions of the activity levels in real life problems; instead they are usually expressed as complex nonlinear functions. In this dissertation, we consider complexities that arise in particular functional decision areas of a supply chain, such as manufacturing operations and distribution, where nonlinear revenue or cost functions arise.

Stages of a supply chain are often provided with a limited budget that can be invested in competing activities. When the budget limit is the only constraining factor, then the resulting problem falls in the well-known class of knapsack problems. The goal is to maximize the overall return obtained from activity investments. In a number of practical contexts (e.g., advertising), the net return from investment in an activity is effectively modeled using an S-Curve, where increasing returns to scale exist at small investment levels, and decreasing returns to scale occur at high investment levels. In Chapter 2, we analyze knapsack problems with S-Curve return functions that consider the allocation of a limited budget to a set of activities or investments in order to maximize net return from investment.

Another type of knapsack problem that we study considers stochastic item sizes and deterministic capacity. Such problems arise in a variety of resource-allocation contexts when the resource capacity must be allocated to tasks with non-deterministic capacity consumption. In Chapter 3, we assume that item weights are random and follow a Poisson

distribution. When the sum of realized weights of items allocated to the knapsack exceeds the knapsack's deterministic capacity, a penalty cost is incurred per unit of overflow. Our goal is to select the items to include in the knapsack that provide the maximum expected net profit. This problem type has many applications in operations planning and assignment problems. For example, for a job-to-machine assignment problem, the knapsack capacity might correspond to the regular working time of the machine, the weight of the item to the processing time of the job on the machine, and the penalty to the overtime cost associated with using the machine.

Transportation is another important function of a supply chain. One aim in transportation problems is to determine the assignment of customers to supply facilities that minimizes total transportation cost while obeying supply limits and meeting customer demands. Transportation costs depend on the locations of facilities. While some planners decide on the facility locations and transportation from these locations to customers separately, when they are considered together, the overall value generated can be increased. Moreover, when we consider contexts with uncertain demands, it is important to consider the impacts of safety stock costs. Therefore, inventory related costs at supply facilities should be accounted for when making decisions on location and allocation. In Chapter 4, we study a supply chain setting where multiple uncapacitated facilities serve a set of customers with stochastic demands. Since the demand is stochastic, some amount of safety stock is held at facilities. The majority of literature on such problems requires assigning all of any given customer's demand to a single facility. While this single-sourcing strategy is optimal under linear (or concave) cost structures, it will often be suboptimal under the nonlinear costs that arise in the presence of safety stock costs. Our primary goal is to characterize the incremental costs that result from a single-sourcing strategy.

An important stage of any supply chain is the production of items. In this stage, one should decide on which products to offer to the market, and how to satisfy the demand. Offering a small number of products will reduce the cost associated with each product,

such as design and holding costs; however, offering a wide selection of products will increase the customer service level since more customers will find products that fit their requirements. In some cases, customers may be willing to accept substitutes if offered some incentive, while, in some cases, customers leave the system without buying anything. Therefore, deciding which products to offer is a critical decision. There are several complicating issues that affect the optimal degree of product variety, one of which is the product architecture. If a product is assembled-to-order from a set of components, then the problem includes an embedded component procurement problem. In Chapter 5, we consider a multi-period component procurement-planning and product-line design problem with product substitutions and multiple customer segments. We have a set of products, each with a fixed design cost and multiple customer segments, where each has demands for ideal products, and sets of alternative products that include all substitutes for their ideal products. If a customer's ideal product is not made available, demand can be satisfied in many cases using an alternative product at a substitution cost. We assume the demand is lost if a customer leaves the system without buying anything. Each product has a profit margin that is customer-segment dependent. Moreover, each product is assembled-to-order from a set of components, and inventory is held at the component level. Hence, our aim is to determine a product portfolio, substitution plan, and procurement plan in order to maximize profit.

In the remainder of this dissertation, we first present our algorithms for knapsack problems that often arise in advertising budget allocation in Chapter 2. We demonstrate that the resulting knapsack problem with S-Curve return functions is $\mathcal{NP}$-Hard, provide a pseudo-polynomial time algorithm for the integer variable version of the problem, and develop efficient solution methods for special cases of the problem. We also discuss a fully-polynomial-time approximation algorithm for the integer variable version of the problem.

In Chapter 3, we study a class of stochastic knapsack problems with Poisson resource requirements. We provide a polynomial-time solution method for the continuous relaxation of this problem, a customized branch-and-bound algorithm for its exact solution, and illustrate the advantages of this solution approach via a set of randomly generated problem instances.

In Chapter 4, we analyze a supply chain setting where multiple uncapacitated facilities serve a set of customers with a single product. We propose a general model that uses a cardinality constraint on the number of supply facilities that may serve a customer. The result is a complex mixed-integer nonlinear programming problem. We provide a generalized Benders decomposition algorithm for the case in which a customer's demand may be split among an arbitrary number of supply facilities. The Benders subproblem takes the form of an uncapacitated, nonlinear transportation problem, a relevant and interesting problem in its own right. We provide analysis and insight on this subproblem, as well as computational results for the general model that permit characterizing the costs that arise from a single-sourcing strategy.

In Chapter 5, we study a multi-period component procurement-planning and product-line design problem with product substitutions. We develop a large-scale mixed-integer linear programming formulation, prove that the problem is $\mathcal{NP}$-Hard and propose a Benders decomposition-based exact algorithm.

Chapter 6 concludes this dissertation by discussing the first four chapters and providing future research directions related to these chapters.

In this dissertation, we provide solution algorithms to problems that arise in certain decision processes within a supply chain. Contributions to the literature are as follows: (1) we show that the continuous knapsack problem with non-identical S-curve return functions is $\mathcal{NP}$-hard, provide potential global optimization approaches for solving this difficult problem, and provide both a pseudo-polynomial time algorithm and a fully polynomial time approximation scheme for the discrete version of the problem; (2) we provide an

effective solution method to a knapsack problem with random item weights following a Poisson distribution, and illustrate the advantages of this approach; (3) we propose a general model for a problem where multiple customers are assigned to multiple facilities that hold safety stock for the assigned customers, and provide an exact solution algorithm to solve the proposed model; and (4) we model a multi-period complex product-line design problem with product substitutions, in which products are assembled from a set of components ordered from an outside supplier, prove that this problem is $\mathcal{NP}$-Hard, and propose an exact algorithm to solve the proposed model.

CHAPTER 2
ALGORITHMS FOR KNAPSACK PROBLEMS WITH S-CURVE RETURN FUNCTIONS

2.1 Introduction and Motivation

The allocation of budget to competing activities occurs in nearly all business applications. When the budget limit is the only constraining factor (when, for example, funds may be invested in various instruments), then the resulting problem falls in the well-known class of knapsack problems. The standard $0-1$ knapsack problem considers a number of items, each with a known weight and value, with a goal of maximizing the value obtained by selecting a subset of the items whose collective weight does not exceed a given capacity limit (see, e.g., Martello and Toth (1990)). In certain contexts (e.g., investment in various financial instruments) the effective weight of an item may itself be a decision variable. That is, if we are free to invest any nonnegative amount up to some upper limit in each element of a set of investment instruments, then we have a continuous version of the knapsack problem. When the value of the instrument is linear in the amount invested, then the resulting problem is a continuous knapsack problem that can be solved by inspection: simply sort instruments in nonincreasing order of per-unit revenue, and insert items into the knapsack until the capacity is exhausted (the more general standard continuous knapsack problem may employ a capacity consumption factor per-unit of the decision variable value, in which case we simply sort items in nonincreasing order of the ratio of per-unit value to per-unit capacity consumption).

If the value of the investment instrument is not a linear function of the investment level, then the resulting nonlinear knapsack problem is not necessarily easily solved (see Bretthauer and Shetty (2002a) for a comprehensive review of the literature on nonlinear knapsack problems). In a number of practical applications, including portfolio selection and advertising budget allocation, the return on investment function may take a nonlinear form leading to complex classes of nonlinear knapsack problems. The relationship between advertising budget allocation and sales response has served as the topic of many studies

in marketing. Simon and Arndt (1980) surveyed the characteristics of sales-advertising response functions. Their survey of the literature showed that the majority of research on advertising response subscribes to one of two proposed shapes of the response function: (1) a nonnegative concave-downward curve and (2) an S-curve. Thus, if a supplier's sales response to advertising in each member of a set of markets follows one of these forms, the supplier faces the challenge of determining the amount of a limited budget to allot to each market in order to maximize sales. In sales-force time-management contexts, a similar phenomenon occurs, where the frequency of sales calls to a client affects the sales response. Lodish (1971) characterizes this response as following an S-curve shape as a function of sales visit frequency. The salesperson must therefore allocate the number of available visits during a planning horizon to each member of a set of clients in order to maximize sales revenue.

Burke et al. (2008) analyzed a related problem that considers the case of concave-downward response functions (which are characterized by nonnegative concave functions with zero return at the origin). They focus on a setting in which a buyer must purchase a fixed quantity from a number of capacitated suppliers, and where each supplier offers a (concave) quantity discount structure. In contrast, we focus on the commonly employed S-curve return functions where increasing returns to scale exist at small investment levels, and decreasing returns to scale occur at high investment levels. Figure 2-1 illustrates an example of the shape of the S-curve return functions we consider.

We examine a budget allocation problem requiring the best allocation of an available budget A among N independent instruments. The return of instrument i is given by the function $\tilde{\mu}_i(a_i)$ where a_i is the investment level allocated to instrument i. The objective is to maximize total net return from budget allocation to the different instruments while not exceeding the limited budget (we later define the term *net return* more precisely in Section 2.3). We recognize the potential uncertainties existing in such application areas, and our model can be employed in such contexts when each function $\tilde{\mu}_i(a_i)$ represents the expected

return for a given investment level, and when the objective is to maximize net expected return. We first show that the problem of allocating a budget among competing activities while maximizing the net return is NP-hard when the return function takes the form of an S-curve. We then consider global optimization methods for solving the problem, analyze a special case in which all S-curves have the same shape, and show that this special case can be solved in polynomial time. Following this, we examine the practical special case in which the investment levels must come from a discrete set of values, and provide a pseudo-polynomial time algorithm for this case, as well as a fully polynomial time approximation scheme (FPTAS).

The remainder of this chapter is organized as follows. In Section 2.2, we review related past literature on budget allocation problems and applications. We define the problem and model formulation in Section 2.3, and discuss a polynomially solvable special case in Section 2.4. In Section 2.5, we consider the integer variable version of the problem, providing a pseudo-polynomial time algorithm as well as a fully polynomial time approximation scheme.

2.2 Literature Review

The allocation of resources among different activities is a critical issue in almost all sectors, which has led to a multitude of research papers on this topic. We discuss related papers on work that deals with sales/advertising contexts, and also consider related work on nonlinear knapsack problems. Zoltners and Sinha (1980) provide a literature review and a conceptual framework for sales resource allocation modeling. They develop a general model for sales resource allocation which simultaneously accounts for multiple sales resources, multiple time periods and carryover effects, non-separability, and risk. Moreover, they discuss several actual applications of the model in practice, which illustrates the practical value of their integer programming models.

When the sales response or costs are not known with certainty, they are often characterized using probability distributions. Holthausen and Assmus (1982) discuss a

model for the allocation of an advertising budget to geographic market segments, when the sales response to advertising in each segment is characterized by a probability distribution. Their model derives an efficient frontier in terms of the expected profit and the variance resulting from alternative budget allocations. Norkin et al. (1998) propose a general stochastic search procedure for the optimal allocation of indivisible resources, which is posed as a stochastic optimization problem involving discrete decision variables. The search procedure develops a branch-and-bound method for this stochastic optimization problem.

The problem of resource allocation among different activities, such as allocating a marketing budget among sales territories is analyzed by Luss and Gupta (1975). They assume that the return function for each territory uses different parameters, and derive single-pass algorithms for different concave payoff functions (based on the Karush-Kuhn-Tucker, or KKT, conditions) in order to maximize total returns for a given amount of effort. A number of efficient procedures have been developed subsequent to this for solving single-resource-allocation problems under objective function and constraint assumptions that lead to convex programming problems, including Zipkin (1980), Bitran and Hax (1981), Bretthauer and Shetty (1995, 2002b), and Kodialam and Luss (1998). In addition, several papers have focused on nonlinear knapsack problems satisfying these convexity assumptions, when the variables must take integer values, including Hochbaum (1995), Mathur et al. (1983), and Bretthauer and Shetty (1995, 2002b).

Surprisingly little literature exists on continuous knapsack problems involving the minimization of a concave objective function (where the KKT conditions are not sufficient for optimality). Moré and Vavasis (1990) provide an efficient method for finding locally optimal solutions for this class of problems assuming objective function separability. Burke et al. (2008) consider a problem in which a producer must procure a quantity of raw materials from a set of capacitated suppliers. The producer seeks to obtain its required materials at minimum cost, where each supplier provides a concave quantity discount

cost structure. The resulting problem takes the form of a continuous knapsack problem involving the minimization of the sum of separable concave functions. They provide a pseudo-polynomial time algorithm and a fully-polynomial-time approximation scheme for the general version of the problem. Sun et al. (2005) provide a partitioning method for the integer version of this problem that uses a linear underestimation of the objective function to provide lower bounds at each iteration. Romeijn et al. (2007) consider the minimization of a specially structured nonseparable concave function over a knapsack constraint, and provide an efficient algorithm for solving this problem.

The literature on knapsack problems in which the objective function is nonconvex (and nonconcave) is somewhat limited. Ginsberg (1974) was the first to consider a knapsack problem with S-curve return functions, which he referred to as "nicely convex-concave production functions". He characterized structural properties of optimal solutions assuming differentiability of the return functions, and predominantly assuming identical return functions. Lodish (1971) considered a nonlinear nonconvex knapsack problem in a salesforce planning context in which the response function is defined at discrete levels of salesforce time investment. He approximated this problem using the upper piecewise linear concave envelope of each function, and provided a greedy algorithm for solving this problem (this greedy algorithm provides an optimal solution for certain discrete knapsack sizes, but not for an arbitrary knapsack size). Freeland and Weinberg (1980) addressed the continuous version of this problem and proposed solving the approximation obtained by using the upper concave envelope of each continuous return function. Zoltners et al. (1979) consider general response functions and also propose an upper concave envelope approximation, along with a branch and bound procedure, that permits successively providing better approximations of the continuous functions at each branch. We discuss a similar method for solving the continuous version of the problem with S-curve return functions that takes advantage of the specialized structure of these return functions. Morin and Marsten (1976) devised a dynamic programming approach

for discrete, nonlinear, and separable knapsack problems, where no convexity or concavity assumptions are made on the objective function, which they assume to be nondecreasing in the decision variables. The storage requirements for determining dominated solutions in their approach grow exponentially, however, in the number of items. Romeijn and Sargut (2009) recently considered a nonconvex, continuous, and separable knapsack problem, which results as a pricing subproblem in a column generation approach for a stochastic transportation problem. They use a sequence of upper bounding functions that permits solving a sequence of specially-structured convex programs such that, in general, the procedure converges to an optimal solution in the limit (we discuss a similar approach for solving the continuous version of our problem in the next section).

Knapsack problems with non-convex (and non-concave) objective functions, such as those mentioned in the previous paragraph, fall into the difficult class of global optimization problems (see Horst et al. (1995)), which require specialized search algorithms that often cannot guarantee finite convergence to a globally optimal solution. The S-curve functions we consider fall into this category, although we are able to exploit the special structure of these functions to provide effective methods for solving the discrete version of this problem. As we later discuss in greater detail, the continuous version of the problem we consider falls into the class of monotonic optimization problems (Tuy (2000)), and specialized methods developed for this class of global optimization problems thus provide a viable option for providing good solutions.

Our primary contributions relative to this body of previous research include showing that the continuous knapsack problem with non-identical S-curve return functions is NP-Hard, providing potential global optimization approaches for solving this difficult problem, and in providing both a pseudo-polynomial time algorithm and a fully polynomial time approximation scheme for the discrete version of the problem.

2.3 Problem Description, Formulation, and Solution Properties

We consider a set $I = \{1, \ldots, N\}$ of marketing instruments, indexed by i, such that the expected return from investing a_i dollars in instrument i is given by the function $\tilde{\mu}_i(a_i)$ for each $i \in I$. We assume that the function $\tilde{\mu}_i(a_i)$ is nonnegative and everywhere locally Lipschitz continuous for all $a_i \geq 0$, and that $\tilde{\mu}_i(a_i)$ is a convex nondecreasing function for $0 \leq a_i \leq \beta_i$ and is concave for $a_i \geq \beta_i$ for some nonnegative β_i. These functional properties allow for modeling S-curve return functions that arise in a number of applications. We define the net return function from investing a_i in instrument i as $\mu_i(a_i) = \tilde{\mu}_i(a_i) - a_i$ and we assume $\mu_i(0) \geq 0$ for all $i \in I$. Note that the function $\mu_i(a_i)$ inherits the convexity and concavity properties of $\tilde{\mu}_i(a_i)$. Let $\partial\mu_i(a_i)$ denote the set of subgradients of $\mu_i(a_i)$ at a_i. Because $\mu_i(a_i)$ is neither everywhere convex nor everywhere concave, we define these subgradients as follows. For $\tilde{a}_i \in [0, \beta_i]$ (in the convex region of $\mu_i(a_i)$), if $\xi \in \partial\mu_i(\tilde{a}_i)$, we have $\mu_i(a_i) \geq \mu_i(\tilde{a}_i) + \xi(a_i - \tilde{a}_i)$ for any $a_i \in [0, \beta_i]$. For $\tilde{a}_i \in [\beta_i, \infty)$ (in the concave region of $\mu_i(a_i)$), if $\xi \in \partial\mu_i(\tilde{a}_i)$, we have $\mu_i(a_i) \leq \mu_i(\tilde{a}_i) + \xi(a_i - \tilde{a}_i)$ for any $a_i \in [\beta_i, \infty)$.

We wish to allocate a budget of A dollars to the marketing instruments in order to maximize total expected return. We formulate this knapsack problem with S-curve return functions (KPS) as follows.

$$\textbf{[KPS]} \qquad \text{Maximize} \quad \sum_{i=1}^{N} \mu_i(a_i)$$

$$\text{Subject to:} \quad \sum_{i=1}^{N} a_i \leq A, \tag{2–1}$$

$$a_i \geq 0, \qquad i = 1, \ldots, N. \tag{2–2}$$

Note that we can apply a nonnegative weight c_i to any item i in the objective function (e.g., $\mu_i(c_i a_i)$) by simply redefining our μ_i function (i.e., $\hat{\mu}_i(a_i) = \mu_i(c_i a_i)$), and the resulting functions retain the S-curve shape (we then need to redefine our β_i and γ_i values accordingly). We can also accommodate a nonnegative weight c_i in the constraint (e.g., $\sum_{i=1}^{N} c_i a_i \leq A$) using the variable substitution $a'_i = c_i a_i$ and redefining the net return function using $\hat{\mu}_i(a'_i) = \mu_i(a'_i/c_i)$.

We next show that we can assume that the S-curves we will deal with are all nondecreasing and nonnegative, without loss of generality. Let γ_i equal the minimum between A and the minimum value of a_i in the concave portion of $\mu_i(a_i)$ at which point this function reaches its maximum value. We can assume that if $\gamma_i = A$, then $\mu_i(a_i) = \mu_i(\gamma_i)$ for all $a_i \geq A$, and we therefore have that $0 \in \partial\mu_i(\gamma_i)$ for all $i \in I$. Note that an optimal solution exists such that $a_i \leq \gamma_i$ for all $i \in I$, and we assume that $\mu_i(\gamma_i) \geq 0$. We can then assume without loss of generality that each $\mu_i(a_i)$ is nondecreasing on $[0, \gamma_i]$ (any decreasing portion of the function must occur in the convex segment of the function, and we can redefine $\mu_i(a_i)$, if necessary, using $\mu_i'(a_i) = \max\{\mu_i(0), \mu_i(a_i)\}$; the resulting functions continue to obey our S-Curve properties). Since we assume that $\mu_i(0) \geq 0$ we therefore need only consider nonnegative, nondecreasing functional forms for $\mu_i(a_i)$ for all $i \in I$. We next provide a key result on the complexity of [KPS].

Theorem 1. *Problem [KPS] is NP-Hard.*

Proof: Consider an instance of the NP-Hard $0-1$ knapsack problem:

$$
\begin{array}{lll}
[\mathbf{KP}_{0,1}] & \text{Maximize} & \sum_{i=1}^{n} r_i x_i \\
 & \text{Subject to:} & \sum_{i=1}^{n} w_i x_i \leq A, \\
 & & x_i \in \{0,1\}, \qquad i = 1, \ldots, N,
\end{array}
$$

where $A, r_i, w_i > 0$ and A, w_i integer for all i. Given an instance of $[\mathrm{KP}_{0,1}]$, we can construct an instance of [KPS] as follows. Given item i define

$$
\mu_i(a_i) = \begin{cases} 0, & 0 \leq a_i \leq w_i - 1, \\ r_i(a_i - (w_i - 1)), & w_i - 1 \leq a_i \leq w_i, \\ r_i, & w_i \leq a_i. \end{cases}
$$

Note that this definition of each $\mu_i(a_i)$ is consistent with our assumptions on the $\mu_i(a_i)$ functions in the definition of the problem [KPS] (with β_i taking any value on the interval

$[w_i - 1, w_i]$). Observe that for any $a_i \leq w_i - 1$, item i's contribution to the objective function of [KPS] is zero, while for any $a_i \geq w_i$, item i contributes r_i to the objective function. An optimal solution for this special case of [KPS] exists with all a_i integer (because A is integer and each $\mu_i(a_i)$ is piecewise linear with integer breakpoints), and any optimal solution can thus be modified to a solution with equivalent objective function value where each a_i equals 0 or w_i. Therefore, an optimal solution to this special case of [KPS] provides an optimal solution for the corresponding instance of [$\text{KP}_{0,1}$], which implies the NP-Hardness of problem [KPS]. □

Solving [KPS]. The nonconvexity of the S-curve functions and the above result imply that we need to draw on global optimization methods for solving [KPS]. We will discuss two such approaches: the first employs recent results on monotonic optimization problems, while the second exploits the special structure of the S-curve functions we are considering.

As mentioned in the previous section, problem [KPS] falls into the class of monotonic global optimization problems (Tuy (2000)), because we are maximizing a nondecreasing function subject to a nondecreasing constraint limited by an upper bound (and where the variables are nonnegative). Tuy (2000) demonstrates the intuitive result that, for such problems, an optimal solution exists on the boundary of the feasible region. He proposes a so-called polyblock algorithm, which performs a search over a sequence of hyper-rectangles. We next briefly describe the application of this approach for solving [KPS]. Let a denote the vector of a_i values ($i = 1, \ldots, N$), and let a^L and a^U denote lower and upper bound vectors on a (initially we have $a_0^L = 0$ and a_0^U is the vector of γ_i values, where the subscript 0 corresponds to an iteration counter). Define $\mathcal{A}$ as the set of all $a \in \mathbb{R}^N$ that satisfy the budget constraint (2–1). Beginning with the initial interval (or polyblock) $P_0 = [a_0^L, a_0^U]$, it is clear that (a) if $a_0^U \in \mathcal{A}$, then this solution is optimal (because of the monotonicity and boundary solution properties), and (b) if $a_0^L \notin \mathcal{A}$, then the problem is infeasible. Assuming that neither of these holds, we wish then to bisect this

polyblock into two smaller polyblocks along one of the variable dimensions. For example, if j denotes the index of the item with the maximum value of γ_i, suppose we consider the two polyblocks $P_{1,1} = [a^L_{1,1}, a^U_{1,1}]$ and $P_{1,2} = [a^L_{1,2}, a^U_{1,2}]$, where $a^L_{1,1} = a^L_0$ for all i, $a^U_{1,1} = a^U_0$ except for the jth element, which equals $\gamma_j/2$. Similarly, $a^L_{1,2} = a^L_0$ except for the jth element, which equals $\gamma_j/2$, while $a^U_{1,2} = a^U_0$. We now have two polyblocks whose union equals the initial polyblock P_0 (and whose intersection is empty).

Given any polyblock $P_k = [a^L_k, a^U_k]$, then clearly if $a^L_k \notin \mathcal{A}$, we can eliminate (prune) the polyblock; on the other hand, if $a^U_k \in \mathcal{A}$ then this solution provides both an upper and lower bound for the best possible solution in the polyblock. If neither of these holds, then a^L_k serves as a lower bound on the best solution in the polyblock, and we utilize an upper bounding method for the best solution in the polyblock (this can be obtained, for example, by establishing the upper concave envelope of each of the functions $\mu_i(a_i)$ in [KPS], replacing these functions with this upper concave envelope function in [KPS], and solving the resulting convex program; to do this, we simply determine the smallest point on the concave portion of $\mu_i(a_i)$ such that $\mu_i(a_i)/a_i \in \partial\mu_i(a_i)$, and connect a line from the origin to this point). We therefore have all of the elements we need for a branch-and-bound type of algorithm, where branching corresponds to bisecting a variable (and thus splitting a polyblock in two), and fathoming a polyblock with index k is done by either (a) verifying that a^U_k is feasible and therefore the best possible solution for the polyblock; (b) verifying that a^L_k is infeasible, and thus pruning the polyblock, or (c) verifying that the polyblock's upper bound solution value is inferior to the best known solution value. This polyblock algorithmic approach will either terminate with an ϵ-optimal solution (where ϵ is a predetermined optimality tolerance), or will converge to an optimal solution value in the limit (Tuy (2000)).

While the polyblock algorithm has been shown to be effective for monotonic optimization problems, the S-curve functions we consider have a special structure that we may exploit to provide alternative global optimization approaches for [KPS]. The following

theorems (2 and 3) provide important properties that we will utilize in developing an additional global optimization solution approach as well as solution methods for various special cases of [KPS]. In particular, Theorem 3 demonstrates that an optimal solution always exists such that at most one instrument i will exist with positive investment at a level less than β_i (i.e., in the convex portion of the $\mu_i(a_i)$ function). This theorem generalizes a similar result provided by Ginsberg (1974) who considered the differentiable case with nonzero second derivatives (i.e., strict concavity in the concave portion and strict convexity in the convex portion of the function).

Theorem 2. *In an optimal solution a^* for [KPS], given any items (i, j) such that $a_i^*, a_j^* > 0$, we must have $\partial\mu_i(a_i^*) \cap \partial\mu_j(a_j^*) \neq \emptyset$.*

Proof: The proof follows from the necessity of the generalized Karush Kuhn-Tucker (KKT) conditions (which are provided in Appendix A); in particular, for all i such that $a_i^* > 0$ the generalized KKT conditions require the existence of a nonnegative w such that $w \in \partial\mu_i(a_i^*)$. □

Theorem 3. *An optimal solution exists for [KPS] with $0 < a_i < \beta_i$ for at most one instrument i.*

Proof: Consider an optimal solution a^* with objective function value z^* such that $0 < a_i^* < \beta_i$ and $0 < a_j^* < \beta_j$ for some $\{i, j\} \in I$. Note that by Theorem 2 we must have $\partial\mu_i(a_i^*) \cap \partial\mu_j(a_j^*) \neq \emptyset$. Consider a solution with $a_k = a_k^*$ for all $k \in I\backslash\{i, j\}$, $a_j = a_j^* + \delta$, and $a_i = a_i^* - \delta$ for some $\delta \leq \min\{a_i^*, \beta_j - a_j^*\}$, denote the objective function value of this new solution by z_n, and let ξ^* denote an element of $\partial\mu_i(a_i^*) \cap \partial\mu_j(a_j^*)$. By the convexity of $\mu_i(a_i)$ for $0 \leq a_i \leq \beta_i$ (and of $\mu_j(a_j)$ for $0 \leq a_j \leq \beta_j$), we have

$$\mu_j(a_j^* + \delta) \geq \mu_j(a_j^*) + \delta\xi^*,$$

$$\mu_i(a_i^* - \delta) \geq \mu_i(a_i^*) - \delta\xi^*.$$

Considering the difference in objective function values between the two solutions given, we have

$$
\begin{aligned}
z_n - z^* &= \mu_i(a_i^* - \delta) - \mu_i(a_i^*) + \mu_j(a_j^* + \delta) - \mu_j(a_j^*) \\
&\geq \delta\left\{\xi^* - \xi^*\right\} \\
&= 0.
\end{aligned}
$$

Since $z_n \geq z^*$, the new solution is optimal. Because this holds for any $\delta \leq \min\{a_i^*, \beta_j - a_j^*\}$, we can set $\delta = \min\{a_i^*, \beta_j - a_j^*\}$, which results in an optimal solution in which either $a_i = 0$ or $a_j = \beta_j$ (or both). The arbitrary selection of the indices i and j and the repeated application of this argument to any pair of items implies that the theorem holds. □

Theorem 3 allows us to eliminate the part of the feasible region where multiple items may take positive values strictly between 0 and β_i in the convex portion of the net return function. This property becomes particularly useful in providing solution methods for a practical special case of problem [KPS] in Section 2.4. It can also aid in a more efficient application of global optimization techniques for [KPS]. We next discuss such a global optimization approach (similar approaches were suggested by Zoltners et al. (1979) for a nonlinear sales resource allocation problem, and by Romeijn and Sargut (2009) for solving a singly-constrained nonlinear pricing problem embedded in a stochastic transportation problem).

Recognizing that at most one instrument exists with an optimal value in the convex portion of the return function, we can thus solve a set of N subproblems, where the ith subproblem requires $0 \leq a_i \leq \beta_i$ and $\beta_j \leq a_j \leq \gamma_j$ for all $j \neq i$. Observe that for the ith subproblem, each of the functions $\mu_j(a_j)$ is concave on the feasible region, with the exception of item i. For this item, we initially approximate $\mu_i(a_i)$ using a line with slope $\mu_i(\beta_i)/\beta_i$ (see the picture on the left in Figure 2-2). The resulting convex programming problem serves as a root node problem for a branch and bound solution approach for the ith subproblem, and the solution provides an upper bound on the optimal solution of the ith subproblem. Suppose that the optimal value of a_i in this initial upper bounding

problem results in $a_i = a_i^*$, and note that the resulting solution is feasible for [KPS], and we can evaluate this solution using the original objective function to provide a lower bound on the best solution for subproblem i. We then begin branching, with one branch considering solutions with $0 \leq a_i \leq a_i^*$ and the other solutions with $a_i^* \leq a_i \leq \beta_i$. As in the solution at the root node, we use an upper linear approximation of the function within each of these intervals, as shown in the figure on the right in Figure 2-2.

Observe that each time we branch, we provide a closer (piecewise linear) approximation of the convex portion of $\mu_i(a_i)$ for the ith subproblem, and that each problem considered at a node in the branch and bound tree is a convex program (and can therefore be efficiently solved using a commercial solver, for example). Moreover, at each node, we obtain both upper and lower bounds on the ith subproblem solution. We can therefore use this branch and bound procedure in search of an ϵ-optimal solution for each of the N subproblems (we would propose using a breadth-first strategy, iterating between the different subproblems, in order to fathom as many of the different subproblems as quickly as possible). As with the polyblock algorithm, given a value of ϵ, this method will find an ϵ-optimal solution in a finite number of steps, and is guaranteed to converge to a global optimal solution in the limit, although not finitely. However, with either approach, we can generate a multitude of feasible solutions in reasonable time, with bounds on the deviation of each solution from optimality.

The following section discusses a special case in which the response functions obey certain strict relationships. These assumed relationships lead to a polynomial-time solution, and also allow us to explore the generalized KKT conditions for [KPS], which are necessary for *local optimality* of a solution (see, e.g., Hiriart-Urruty (1978)).

2.4 Polynomially Solvable Special Case

This section considers a special case in which the return functions obey certain properties that permit creating a preference rank ordering for different instruments. We first consider the case in which all of the return functions are identical, i.e., $\mu_i(a_i) = \mu(a)$

with an inflection point occurring at β, and a maximum value occurring at γ such that $\beta \leq \gamma < A$ for all $i \in I$. We restrict our focus to differentiable functions in this section (except possibly at β), noting that similar results can be obtained for functions with points of non-differentiability. Although this special case appears to be simple at first, its analysis allows us to illustrate the potential complexities of this problem class, and it also paves the way for characterizing the complexity of more general cases.

Under identical revenue curves, Theorem 2 implies that all instruments whose investment level is positive and falls in the concave part of the curve will have identical values of a_i at optimality. Moreover, Theorem 3 allows us to arbitrarily select any instrument as one whose a_i value may be positive and fall in the interval $(0, \beta)$. We employ the necessary KKT conditions (see Appendix A) to analyze this problem. We first suppose that the KKT multiplier associated with the knapsack constraint, denoted by w, is zero. In this case we have that $\lambda_i = d\mu(a_i)/da_i$ and $\lambda_i a_i = 0$ for all $i \in I$, where λ_i is a KKT multiplier associated with the ith nonnegativity constraint. Thus if a_i is positive, we have that $d\mu(a_i)/da_i = 0$ at a KKT point when $w = 0$. Because we assume (without loss of generality) that the return functions are nondecreasing, any zero derivative point in the convex portion of the curve must have a return function equal to $\mu(0)$, and we can thus ignore stationary points in the convex portion of the curve. Noting that $d\mu(\gamma)/da = 0$, and letting $\bar{n} = \lfloor A/\gamma \rfloor$, we have that any solution such that $\bar{n}$ of the a_i values are set to γ serves as a candidate for an optimal solution (because each of these has objective function value $\bar{n}\mu(\gamma)$, we need only consider one such solution).

We next consider the case in which $w > 0$, which implies that the knapsack constraint must be tight at any associated KKT point. Such a KKT point must satisfy the following system of equations:

$$w = \frac{d\mu(a_i)}{da_i}, \quad \forall i \in I : a_i > 0,$$

$$\sum_{i=1}^{N} a_i = A,$$

$$w \geq 0,$$

$$a_i \geq 0, \quad \forall i = 1, \ldots, N.$$

(Note that for our S-curves, $\frac{d\mu(a)}{da} \geq 0$ for all $a \geq 0$ and therefore $w \geq 0$ is redundant in the above system.) Recall that at most one variable can have an a_i value that falls in the convex portion of the return function. Let us first consider cases in which no variable takes a value in this range. If we suppose that we know that n variables take a positive value, then we seek a positive (single variable) a such that $a \geq \beta$ and $na = A$. For $n = 1, \ldots, N$, we can quickly determine whether the solution with $a = A/n$ satisfies $d\mu(a)/da \geq 0$ with $a \geq \beta$. If such a solution exists, then it serves as a candidate for an optimal solution. We next suppose that one of the variables may take a value in the interval $(0, \beta)$. Letting a^l denote the value of the single variable in this interval, and letting a^u denote the value of the variable(s) falling in the concave portion, and noting that at most N such solutions may exist (with $n = 0, \ldots, N-1$), we seek solutions satisfying the following system of equations:

$$\begin{aligned} \frac{d\mu\left(a^l\right)}{da^l} - \frac{d\mu(a^u)}{da^u} &= 0, \\ a^l + na^u &= A, \\ 0 \leq a^l &\leq \beta, \\ \beta \leq a^u. \end{aligned} \tag{2–3}$$

The difficulty of finding a solution to this system of equations depends on the functional form of the derivative function. In cases where the equation $d\mu\left(a^l\right)/da^l - d\mu\left(a^u\right)/da^u = 0$ intersects the line $a^l + na^u = A$ only once in the interval $0 \leq a^l \leq \beta$, for the given value of n we can perform a line search to determine the unique solution satisfying the above system of equations. When the $\mu(a)$ function takes a second-degree polynomial form on both the convex and concave intervals, then this provides a sufficient condition for having at most one solution to the above system (note that if the equation $d\mu\left(a^l\right)/da^l - d\mu\left(a^u\right)/da^u = 0$ is linear, then it cannot be collinear with the equation $a^l + na^u = A$

because $d\mu(a^u)/da^u \geq 0$, eliminating the possibility of an infinite number of candidate solutions satisfying the KKT conditions). In such cases, the first constraint in the above system is linear, and we have two linearly independent equality constraints in two variables.

Assuming the previously stated conditions for a unique solution to the above system of equations, we then perform a line search for each possible value of n along the line $0 \leq a^l = A - na^u \leq \beta$ to determine (at most) N additional candidate solutions. Note that if $\mu(a)$ is not strictly positive for all $a \in (0, \beta)$, then letting $\tilde{a}$ denote the largest value of a such that $\mu(a) = 0$, we can limit our search to the interval $(\tilde{a}, \beta)$. The complexity of this line search is $\mathcal{O}(\log \beta)$. The overall complexity of this approach is therefore $\mathcal{O}(N \log \beta)$. The following algorithm summarizes our approach for solving [KPS] with identical response functions, assuming the system of equations (2–3) has at most one solution for any value of n.

Algorithm 1 Solve [KPS] with Identical Response Functions.

Initialize $LB = \bar{n}\mu(\gamma)$, where $\bar{n} = \left\lfloor \frac{A}{\gamma} \right\rfloor$
for $n = 1$ to N **do**
  Set $a_n = \frac{A}{n}$
  if $a_n \geq \beta$ and $n\mu(a_n) > LB$ **then**
    $LB \leftarrow n\mu(a_n)$
  Solve system of equations (2–3)
  if a feasible solution exists for (2–3) with $a^l = a^l_n$, $a^u = a^u_n$, and $\mu(a^l_n) + n\mu(a^u_n) > LB$ **then**
    $LB \leftarrow \mu(a^l_n) + n\mu(a^u_n)$
Optimal Solution Value $z^* = LB$

Note that this solution approach applies not only when all N curves are strictly identical, but generalizes to the case in which each pair of curves differs by a constant value for all $a \geq 0$. In such cases, we cannot arbitrarily select an item whose value falls in the convex portion, but we need to separately consider each of the N curves as a candidate for taking a value in the convex portion. Moreover, given that some n items take values in the concave portion of the curve, we now have a dominance order in which

to select these n items from among the N (i.e., those with higher values of $\mu_i(0)$ are preferred to those with lower values). For this case, the complexity therefore increases by a factor of N, leading to a worst-case bound of $\mathcal{O}(N^2 \log \beta)$.

For more general versions of the problem, where the $\mu_i(a_i)$ functions have no relationship, we showed in the previous section that the problem takes the form of a difficult global optimization problem. Although the KKT conditions are necessary for optimality, it is impractical in the general case to try to enumerate all KKT points in order to account for all local minima, in an attempt to find a global minimum. We next illustrate this complexity for the differentiable case. For the case in which the KKT multiplier for the knapsack constraint is zero (i.e., $w = 0$), we still require $d\mu_i(a_i)/da_i = 0$ for all $a_i > 0$. Because $d\mu_i(\gamma_i)/da_i = 0$, we require finding a subset $\bar{I}$ of $\{\gamma_1, \gamma_2, \ldots, \gamma_N\}$ such that $\sum_{i \in \bar{I}} \gamma_i \leq A$ with the maximum value of $\sum_{i \in \bar{I}} \mu_i(\gamma_i)$. This problem is itself a $0-1$ knapsack problem and, therefore, identifying candidate solutions using the KKT conditions does not lead to a polynomial-time solution approach. Additionally, the case in which $w > 0$ requires finding *all solutions* of a set of N (possibly nonlinear) equality constraints (while also satisfying $2N$ nonnegativity conditions), for each of the N choices of the variable which may take a value in the convex portion of the curve. We note here that nonlinear programming methods used in a number of commercial solvers (such as conjugate gradient methods; see, e.g., Bazaraa et al. (2006)) can be utilized in an attempt to identify a locally optimal point, although these methods cannot guarantee finding a globally optimal solution for global optimization problems.

We next focus on the practical case where all a_i variables must take integer values, where we can employ our previous properties of optimal solutions and provide algorithms of practical use that lead to solutions with provable bounds on performance.

2.5 Model with Integer Variable Restrictions

In the majority of practical contexts where problem [KPS] applies, it is acceptable (or even necessary) to limit the set of possible a_i values to a discrete set (where, e.g., each a_i

value is denominated in some currency). Therefore, we henceforth consider problem [KPS] with the added restriction that each a_i value must be integer, denoting this problem as [KPSI] (note that our solution approaches can easily be extended to any discrete candidate set of a_i values). We assume that the piecewise-linear function obtained by connecting successive values of $\mu_i(a_i)$ at integer values of a_i with a line segment continues to satisfy the the S-curve properties we have defined. As the following theorem shows, we can work with these continuous piecewise-linear functions in order to solve [KPSI].

Let [KPSPL] denote the restricted version of [KPS] in which all of the $\mu_i(a_i)$ functions are piecewise-linear functions with integer breakpoints.

Theorem 4. *An optimal solution for [KPSPL] exists in which all a_i take integer values.*

Proof: Assume we have an optimal solution a^* in which there are two or more fractional values of a_i. Consider two of these fractional items a_j^* and a_k^* where $l_j < a_j^* < u_j$ and $l_k < a_k^* < u_k$, where (l_i, u_i) define the integer breakpoints on either side of a_i^* for any $i \in I$. Let $a_i = a_i^*$ for $i \neq j, k$ and $\overline{A} = A - \sum_{i \neq j,k} a_i$. Consider the following linear program (LP):

$$\begin{aligned} \text{Maximize} \quad & \mu_j(a_j) + \mu_k(a_k) = \rho_j a_j + \rho_k a_k \\ \text{Subject to:} \quad & a_j + a_k \leq \overline{A} \\ & l_j \leq a_j \leq u_j \\ & l_k \leq a_k \leq u_k \end{aligned}$$

where ρ_j and ρ_k are the slopes of the return functions of instrument j and k at the points a_j^* and a_k^*, respectively. An optimal solution exists for this LP such that at least one of the variables j and k falls at one of its (integer) bounds (because at most one of these variables can be basic) and a resulting objective function value at least as high as that of the solution a^*. Therefore, we either have that the original solution is suboptimal (a contradiction), or an alternative optimal solution is obtained with one less fractional variable. If two or more fractional variables remain, we can repeat this procedure until two fractional variables remain, whose sum must be integer because of the integrality of A.

An optimal solution for this final LP exists with integer values for both variables, as they must sum to an integer value, and at least one of them must take an integer value. □

Theorems 3 and 4 together imply that we can solve the integer variable version of the problem using the continuous piecewise-linear function obtained by connecting successive values of $\mu_i(a_i)$ at integer values of a_i with a line segment for all i, and an optimal solution will exist with at most one value of a_i strictly between 0 and β_i. This permits the construction of a pseudo-polynomial time algorithm for solving [KPSI] as we next discuss.

2.5.1 Pseudo-Polynomial Time Algorithm

Given that an optimal solution exists with at most one instrument in the convex part of the S-curve, we can use the following approach. Suppose we arbitrarily select any instrument and assume that the investment amount for this instrument will be at some value in the convex portion of the function. That is, given some variable j, suppose we set $0 < a_j < \beta_j$. This implies that any other variable i must either equal zero or fall in the interval $[\beta_i, \gamma_i]$. Given some value of a_j between 0 and β_j, say a'_j, then define $\overline{A}_j = A - a'_j$. The remainder of the problem reduces to the following generalization of the $0-1$ knapsack problem:

$$\textbf{[KPEI]} \qquad \text{Maximize} \quad \sum_{i \in I \backslash \{j\}} \mu_i(a_i) \tag{2–4}$$

$$\text{Subject to:} \quad \sum_{i \in I \backslash \{j\}} a_i \leq \overline{A}_j, \tag{2–5}$$

$$\beta_i x_i \leq a_i \leq \gamma_i x_i, \qquad \forall i \in I \backslash \{j\}, \tag{2–6}$$

$$a_i \in \mathcal{Z}_+, x_i \in \{0, 1\}, \quad \forall i \in I \backslash \{j\}, \tag{2–7}$$

where $\mathcal{Z}_+$ is the set of nonnegative integers. Balakrishnan and Geunes (2003) provided a pseudo-polynomial time algorithm for the above problem when each $\mu_i(a_i)$ has a fixed plus linear structure (i.e., a fixed reward for including item i, plus a variable contribution to profit per unit weight). They referred to this problem as a knapsack problem with

expandable items. Constraint (2–5) serves as a simple knapsack constraint. Constraint set (2–6) forces an item's weight to zero if the item is not included in the knapsack (when $x_i = 0$) and requires the item's weight to fall between some prespecified upper and lower bounds if the item is included. The objective function (2–4) maximizes the net return from filling the knapsack. The dynamic program used to solve [KPEI] in Balakrishnan and Geunes (2003) is a straightforward generalization of the standard dynamic program used for solving knapsack problems, where all integer feasible values of each a_i are implicitly enumerated. The worst-case running time for this dynamic programming approach for a given instrument j assumed to have an investment level between 0 and β_j and a given value of a_j is $\mathcal{O}(NAT)$, where $T = \max_{i \in I}\{\gamma_i - \beta_i\}$.

Using Theorems 3 and 4, we can solve [KPSI] by using this dynamic programming approach to solve [KPEI] for each possible value of a_i such that $0 < a_i < \beta_i$ and for all $i \in I$. That is, we select an instrument j, assume that this instrument has an investment level between 0 and β_j (recall that this can be true for at most one instrument), and solve the associated problem [KPEI]. Because there are no more than $\overline{\beta} = \max_{i \in I}\{\beta_i\}$ possible values for a_j, and because we must solve an instance of [KPEI] for each of these values of a_j and for each of the N items, the worst-case computational effort for this approach is $\mathcal{O}(N^2 A\overline{\beta}T)$.

2.5.2 Fully Polynomial Time Approximation Algorithm

We next develop a fully polynomial-time approximation scheme (FPTAS) for [KPSI]. Given an $\epsilon > 0$, an FPTAS is polynomial in N and $1/\epsilon$, and results in an objective function value of no less than $(1 - \epsilon)z^*$, where z^* denotes the value of the optimal solution. The approach we use is related to the approach Van Hoesel and Wagelmans (2001) employed for capacitated economic lot-sizing.

We begin with a "profit-based" dynamic program for solving KPSI, letting $F_i(\pi)$ denote the minimum amount of capacity in the constraint consumed while providing a profit of at least π, and including all instruments up to (and including) instrument i.

Thus, the the largest value of π such that $F_N(\pi) \leq A$ is the optimal solution value. We assume a given upper bound on profit of Π (we will subsequently describe our method for deriving a valid value for this upper bound).

The dynamic program first solves the single-item problem:

$$F_1(\pi) = \min\{a_1 \mid \mu_1(a_1) \geq \pi, 0 \leq a_1 \leq \gamma_1\}, \quad \pi = 0, \ldots, \Pi.$$

Note that because $\mu_1(a_1)$ is a univariate nondecreasing function, the feasible region of the above problem is convex (assuming it is non-empty). If $\mu_1(0) > \pi$, then $\mu_1(a_1) \geq \pi$ for all $a_1 \in [0, \gamma_1]$ and the constraint $\mu_1(a_1) \geq \pi$ is redundant (because $\mu_1(a_1)$ is nondecreasing), and in this case we have $F_1(\pi) = 0$. If this is not the case, let a_1^l denote the smallest value on the interval $[0, \gamma_1]$ such that $\mu_1(a_1^l) = \pi$, and thus $F_1(\pi) = a_1^l$. We can, therefore, easily determine $F_1(\pi)$ for a well defined nondecreasing function f_1, by finding the (smallest) root of $\mu_1(a_1) - \pi = 0$. This can be done (assuming a root exists, which occurs if $\mu_1(\gamma_1) \geq \pi$) in $\mathcal{O}(\log \gamma_1)$ time using binary search. Next consider the two-item problem:

$$F_2(\pi) = \min_{\alpha=0,1,\ldots,\pi} \left\{ \min_{a_2=0,1,\ldots,\gamma_2} \{F_1(\alpha) + a_2 \mid \mu_2(a_2) \geq \pi - \alpha\} \right\}.$$

Given a value of α, the inner minimization problem can be solved in $\mathcal{O}(\log \gamma_2)$ time. The general recursion can be stated as

$$F_i(\pi) = \min_{\alpha=0,1,\ldots,\pi} \left\{ \min_{a_i=0,1,\ldots,\gamma_i} \{F_{i-1}(\alpha) + a_i \mid \mu_i(a_i) \geq \pi - \alpha\} \right\},$$

with the inner minimization problem for a given α and π having worst-case complexity $\mathcal{O}(\log \gamma_i)$. Given i and π, we can compute $F_i(\pi)$ in $\mathcal{O}(\pi \log \Gamma)$ operations (where $\Gamma = \max_{i \in I}\{\gamma_i\}$). Because we have Π values of π and N values of i, this recursion takes $\mathcal{O}(N\Pi^2 \log \Gamma)$ time. We thus have an algorithm that is pseudo-polynomial in the profit upper bound value Π.

We can set a suitable value of Π as follows. Letting $\mu^{\max} = \max_{i \in I} \{\mu_i(\gamma_i)\}$, then $\Pi = N\mu^{\max}$ provides an upper bound on z^*, the optimal solution value. Note also that

any single value of γ_i is feasible (i.e., less than or equal to A), and we therefore have that $\mu^{\max} \leq z^*$, which implies that $\Pi \leq Nz^*$.

Profit scaling.. In the recursion we have described, suppose that, instead of evaluating every integer value of π, we evaluate every K^{th} value. That is, for $i \in I$ and $\pi \in \{0, K, 2K, \ldots, \lceil \Pi/K \rceil K\}$, define $G_i(\pi)$ as the minimum capacity that can be consumed using the first i instruments, with a profit of no less than π, when the profit allocated to each instrument is a multiple of K. Computing $G_i(\pi)$ for $i \in I$ and $\pi \in \{0, K, 2K, \ldots, (\lceil \Pi/K \rceil)K\}$ then requires $\mathcal{O}(N(\Pi/K)^2 \log \Gamma)$ operations. The following proposition shows that using this approach provides a feasible solution, and a bound on the objective function value.

Proposition 1. *At least one element $\pi \in \{0, K, 2K, \ldots, \lceil \Pi/K \rceil K\}$ exists with $G_N(\pi) \leq A$, i.e., evaluating every K^{th} value of π leads to a feasible solution if one exists. Moreover, the maximum π in this set with $G_N(\pi) \leq A$ equals at least $z^* - NK$.*

Proof. Suppose we have an optimal solution with objective function value z^*, and denote r_i as the profit contribution of item i in this solution. Suppose we allocate a profit equal to $\lfloor r_i/K \rfloor K \leq r_i$ to item i (for all values of i). Then, in the resulting solution, each value of a_i will be less than or equal to its corresponding value in the optimal solution. The resulting solution under the scaling algorithm is therefore feasible under these profit allocations. Since $\sum_{i \in I} r_i \leq \Pi$, the following inequalities show that we account for these profit allocations in the algorithm:

$$\sum_{i \in I} \left\lfloor \frac{r_i}{K} \right\rfloor K \leq \sum_{i \in I} \left(\frac{r_i}{K} \right) K \leq \left(\frac{\Pi}{K} \right) K \leq \left\lceil \frac{\Pi}{K} \right\rceil K.$$

Finally, by the definition of r_i, we have

$$\sum\nolimits_{i \in I} \lfloor r_i/K \rfloor K = \sum\nolimits_{i \in I} (\lceil r_i/K \rceil - 1) K \geq \left\lceil \sum\nolimits_{i \in I} r_i/K \right\rceil K - NK = \lceil z^*/K \rceil K - NK.$$

This last term is greater than or equal to $z^* - NK$, which implies the stated result. □

Selecting an appropriate value of K.. Suppose we select $K = \max\left\{1, \left\lfloor \frac{\epsilon\Pi}{N^2} \right\rfloor\right\}$. From Proposition 1, we have a lower bound on the solution value from the rounding procedure of $z^* - NK$. Because $NK = N\left\lfloor \frac{\epsilon\Pi}{N^2} \right\rfloor \leq \frac{\epsilon\Pi}{N}$ and because $\Pi \leq Nz^*$, we have $NK \leq \epsilon z^*$, which implies that our lower bound satisfies $z^* - NK \geq z^* - \epsilon z^*$, i.e., that our scaling approach finds a solution with value at least $z^*(1-\epsilon)$. Recall our worst case complexity bound of $\mathcal{O}(N(\Pi/K)^2 \log \Gamma)$. If $\Pi/K < N$, then this results in a polynomial-time algorithm. Suppose, conversely, that $\Pi/K > N$. If $\frac{\epsilon\Pi}{N^2} > 1$, then $K = \left\lfloor \frac{\epsilon\Pi}{N^2} \right\rfloor > \frac{\epsilon\Pi}{2N^2}$; if $\frac{\epsilon\Pi}{N^2} \leq 1$, then $K = 1$. In either case we have $K \geq \frac{\epsilon\Pi}{2N^2}$, which implies $\frac{\Pi}{K} \leq \frac{2N^2}{\epsilon}$, and our worst-case complexity becomes $\mathcal{O}\left(\frac{N^5}{\epsilon^2} \log \Gamma\right)$, which is polynomial in N and $1/\epsilon$.

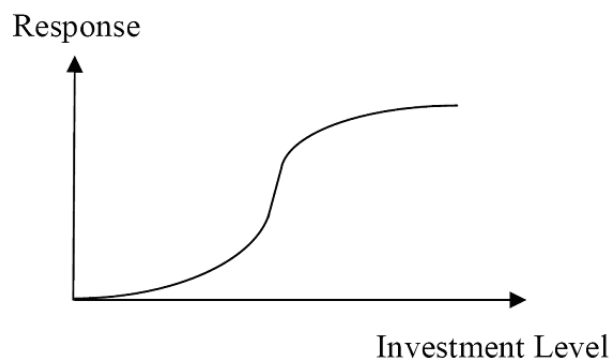

Figure 2-1. An S-curve response function

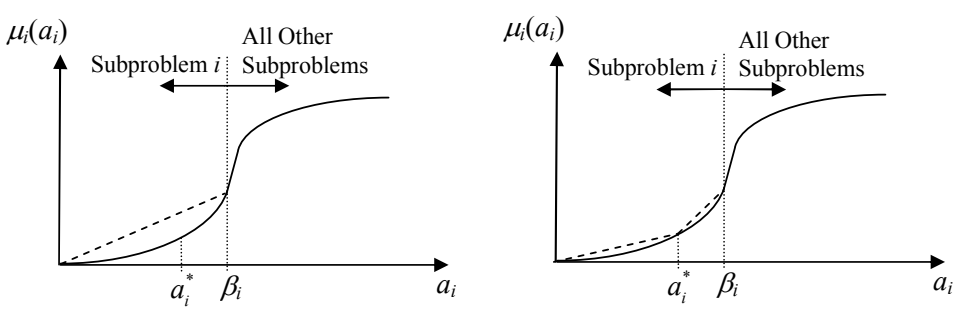

Figure 2-2. Iterative bounding functions for the model

CHAPTER 3
A SINGLE-RESOURCE ALLOCATION PROBLEM WITH POISSON RESOURCE REQUIREMENTS

3.1 Introduction

This chapter studies a generalization of the classical knapsack problem that addresses the problem with random item sizes and a deterministic capacity. Such problems arise in a variety of resource-allocation contexts when the resource capacity must be allocated to tasks with non-deterministic capacity consumption. This problem is defined as follows. Consider a set of N items indexed by j, where the size of item j (e.g., weight) follows a known probability distribution. The value (or expected value) of item j per unit size is r_j and we assume that all items are statistically independent. We also assume, without loss of generality, that items are sorted in nonincreasing order of r_j. We consider a resource with capacity B (expressed in the same units as item sizes). The goal is to assign items to the resource in order to maximize the expected value of items assigned to the resource, less expected overflow costs, where each unit of capacity overflow is assessed a penalty of π. In the special case in which the capacity overflow cost is infinite and the probability distributions of item weights are degenerate (i.e., weights are deterministic), this problem is equivalent to a standard 0-1 knapsack problem. In contrast, this chapter will focus on the previously unstudied case in which the weight of item j follows a Poisson distribution with parameter λ_j and the penalty cost is finite.

To formalize this model, let x_j equal 1 if item j is assigned to the resource, and let $V(x)$ denote the random variable for the aggregate size of items assigned to the resource, where x denotes the N-vector of x_j values. Note that $V(x)$ is Poisson distributed with parameter $\nu = \sum_{j=1}^{N} \lambda_j x_j$ (we will find it convenient to use the continuous variable ν in the problem formulation, although this variable can, of course, be substituted out of the formulation).

We can formulate the static stochastic knapsack problem with Poisson distributed item sizes as (SKPP). Observe that if r_j is an expected unit revenue, then our formulation

implicitly assumes that an item's unit revenue and weight are uncorrelated random variables with an expected value $r_j\lambda_j$ (if r_j is a deterministic unit revenue, then this term also correctly specifies the expected revenue of an item). Note also that the Poisson distribution is undefined for $\nu = 0$. When $\nu = 0$, however, no items are selected, and the expected overflow cost, as well as the expected revenue, will equal zero. We therefore propose solving the problem with the additional constraint $\nu \geq \underline{\lambda}$, where $\underline{\lambda} = \min_{j=1,\ldots,N}\{\lambda_j\}$. If the resulting solution produces positive profit, then clearly the solution with $\nu = 0$ is not an optimal solution. If, on the other hand, the resulting solution has negative expected profit, then the solution with $\nu = 0$ (and all $x_j = 0$) is optimal.

$$
\begin{array}{llll}
\text{[SKPP]} & \textbf{Maximize} & \displaystyle\sum_{j=1}^{N} r_j\lambda_j x_j - \pi \sum_{i=B+1}^{\infty} (i-B)\frac{e^{-\nu}\nu^i}{i!} & \\
 & \textbf{Subject to:} & \nu = \displaystyle\sum_{j=1}^{N} \lambda_j x_j & \\
 & & x_j \in \{0,1\}, \qquad \forall j = 1,\ldots,N. &
\end{array}
$$

It is straightforward to show that an optimal solution exists for the continuous relaxation of [SKPP] with at most one fractional variable, i.e., with at most one item j such that $0 < x_j < 1$. To show this, note that for any given value of ν, the continuous relaxation of [SKPP] is a linear program with a single constraint (i.e., a continuous knapsack problem). We will exploit this property of optimal solutions to provide a polynomial-time solution method for the continuous relaxation of [SKPP] and, therefore, an effective upper bounding method for use in a customized branch-and-bound algorithm.

Problem [SKPP] has many applications in operations planning and assignment problems. For example, for a job-to-machine assignment problem, the knapsack capacity might correspond to the regular working time of the machine, the weight of the item to the processing time of the job on the machine, and the penalty to the overtime cost associated with using the machine. Another example would be a customer-package-pickup-to-vehicle assignment problem. If the size of each customer's pickup requirements is a

Poisson random variable (and the vehicle capacity corresponds to the knapsack capacity), then the resulting assignment of customer pickups to the vehicle is an [SKPP] (in this setting the overflow cost might correspond to the cost of sending an extra vehicle or finding an alternate pickup method when the sum of assigned customer pickups exceeds vehicle capacity). As a third example, a salesperson may be allocated a stock of inventory (of size B), and may be assigned sales calls to a certain subset of potential customers to whom s/he attempts to sell this inventory. In this context the penalty cost would correspond to the cost of having insufficient inventory, i.e., the cost of a lost sale.

In the first two motivating examples, capacity violations are permitted at a cost. In these motivating examples, observe that in practice, the overflow capacity itself may have a hard limit (in the first example, overtime availability is limited by the total hours in the day less regular time, and in the second example, an extra delivery vehicle itself has a capacity). Because the Poisson distribution is defined from zero to infinity (as are many commonly employed probability distributions), this implies that our model cannot ensure obeying such a hard capacity limit with probability one. By setting the overflow cost appropriately, however, the model can ensure that the probability of exceeding this capacity is negligible (when overflow capacity is limited, our model is, therefore, more appropriate for situations in which this penalty cost is relatively high, leading to a low probability of exhausting overflow capacity). Note that in the third sales/inventory example, no such hard limit exists on effective overflow capacity, because demand overflows correspond to lost sales.

The remainder of this chapter is organized as follows. In Section 3.2, we review related past literature on the stochastic knapsack problems. We define the problem and formulate its mathematical model in Section 3.3, and discuss a special case. We propose a solution approach to the problem in Section 3.4 and provide the computational study in Section 3.5.

3.2 Literature Review

The class of stochastic knapsack problems contains a number of variants in which either item values or sizes (or both) are random variables. The goal in such problems is typically to maximize the values (or expected values) of items included in the knapsack while obeying certain constraints (such as a constraint on the probability of overflow). Dean et al. (2004) consider a version of the stochastic knapsack problem in which item sizes are independent random variables, while the values of items are fixed. An item's size is revealed immediately upon determining whether or not to allocate it to the knapsack. The goal is to design an algorithm that selects the items, one at a time, until the knapsack capacity is exceeded. Goel and Indyk (1999) study knapsack problems with Poisson item sizes, where the objective is the maximization of the sum of values of items included in the knapsack, subject to a constraint on the maximum probability of overflow. They provide a polynomial time approximation scheme for this problem via a simple reduction to the deterministic case.

Several past papers have considered the case in which item sizes are fixed but item values are random (see, e.g., Henig (1990), Carraway et al. (1993), and Steinberg and Parks (1979)); in contrast, we consider probabilistic item sizes.

An additional problem class worth mentioning is the class of dynamic stochastic knapsack problems, where items arrive dynamically over time. The values and the sizes of the items are random and become known at the time of the arrival. The goal is to find a control policy for accepting or rejecting arriving objects (as a function of the current state of the system) in order to maximize the total value of items accepted in the knapsack. (Kleywegt and Papastavrou (1999), Kleywegt and Papastavrou (2001)), Papastavrou et al. (1996), and Ross and Tsang (1989) provide examples of problems falling in this class. In this chapter, however, we study a static knapsack problem without a time dimension and where random item realizations are not revealed until after all assignments are made.

The most closely related work to ours addresses the same class of static stochastic knapsack problems that we do, but under the assumption of normally distributed item sizes. Barnhart and Cohn (1998) identified special cases of the general problem that lead to simple solution methods, and provided dominance rules for use in an implicit enumeration approach. Kleywegt et al. (2001) provided a sample-average approximation scheme for solving this problem approximately. More recently, Merzifonluoğlu et al. (2009) provided a polynomial-time solution algorithm for the continuous relaxation of this problem under normal item sizes, and used this solution to provide strong upper bounds in a branch-and-bound scheme. Our consideration of Poisson distributed item sizes broadens the set of tools available for this problem class. Moreover, the Poisson distribution has the added benefit of being defined only for nonnegative values, in contrast to the normal distribution, which is often used as an approximation for random variables that cannot take negative values (note that a Poisson distribution with parameter λ is equivalent to the sum of λ independent Poisson random variables, each with parameter 1; by the central limit theorem, the Poisson distribution tends to a normal distribution as λ increases). To the best of our knowledge, no prior method exists in the literature for addressing this relevant problem class under an assumption of Poisson distributed item weights.

3.3 Problem Analysis and a Solution Method for a Special Case

In this section we analyze structural properties of [SKPP] and provide solution algorithms for the continuous relaxation as well as the binary version of the problem. First, we consider the nonlinear component of the objective function, i.e., the expected capacity overflow. Note that the expected overflow term in the objective can be rewritten as a finite summation as follows:

$$\begin{aligned}\sum_{i=B+1}^{\infty}(i-B)\frac{e^{-\nu}\nu^i}{i!} &= \sum_{i=0}^{\infty}(i-B)\frac{e^{-\nu}\nu^i}{i!}+\sum_{i=0}^{B}(B-i)\frac{e^{-\nu}\nu^i}{i!}\\ &= \nu-B+\sum_{i=0}^{B}(B-i)\frac{e^{-\nu}\nu^i}{i!}.\end{aligned}$$

Letting $\hat{r}_j = r_j - \pi$ and dropping the constant term πB from the objective function, we can rewrite our optimization problem as:

$$\textbf{Maximize} \quad \sum_{j=1}^{N} \hat{r}_j \lambda_j x_j - \pi \sum_{i=0}^{B} (B-i) \frac{e^{-\nu}\nu^i}{i!} \tag{3–1}$$

$$\textbf{Subject to:} \quad \nu = \sum_{j=1}^{N} \lambda_j x_j \tag{3–2}$$

$$\nu \geq \underline{\lambda} \tag{3–3}$$

$$x_j \in \{0,1\}, \qquad \forall i = 1, \ldots, N. \tag{3–4}$$

Letting $g(\nu) = \sum_{i=0}^{B}(B-i)\frac{e^{-\nu}\nu^i}{i!} = \sum_{i=0}^{B}(B-i)P\{V(x) = i\}$, we next characterize some properties of this function. The first and second derivatives of $g(\nu)$ can be written as

$$\begin{aligned} g'(\nu) &= \sum_{i=0}^{B}(B-i)\frac{e^{-\nu}}{i!}(i\nu^{i-1} - \nu^i) \\ &= -\sum_{i=0}^{B-1}\frac{e^{-\nu}\nu^i}{i!} \\ &= -P(V(x) \leq B-1). \end{aligned}$$

$$\begin{aligned} g''(\nu) &= -\sum_{i=0}^{B-1}\frac{e^{-\nu}}{i!}(i\nu^{i-1} - \nu^i) \\ &= -\sum_{i=0}^{B-2} P\{V(x) = i\} + \sum_{i=0}^{B-1} P\{V(x) = i\} \\ &= P\{V(x) = B-1\} \geq 0. \end{aligned}$$

This implies that $g(\nu)$ is convex in ν, which implies that the objective function of [SKPP] is concave. This also implies that the continuous relaxation of [SKPP] is a convex program with a linear constraint set, which means that the KKT conditions are necessary and sufficient for optimality of this relaxation. Note that we can also show that $g(\nu)$ is a convex function of x (this can be shown by verifying that the Hessian of $g(\nu)$ is positive semi-definite after substituting out the ν variable).

We define θ and γ as KKT multipliers for constraints (3–2) and (3–3), respectively, and let α_j and β_j denote KKT multipliers for the constraints $x_j \leq 1$ and $-x_j \leq 0$, respectively, for $j = 1, \ldots, N$. The associated KKT conditions can be written as:

$$\hat{r}_j \lambda_j - \alpha_j + \beta_j = -\lambda_j \left(\pi \sum_{i=0}^{B-1} \frac{e^{-\nu} \nu^i}{i!} + \gamma \right) \quad j = 1, \ldots, N \tag{3–5}$$

$$\gamma, \alpha_j, \beta_j \geq 0 \quad j = 1, \ldots, N \tag{3–6}$$

$$\alpha_j (1 - x_j) = 0 \quad j = 1, \ldots, N \tag{3–7}$$

$$\beta_j x_j = 0 \quad j = 1, \ldots, N \tag{3–8}$$

$$\gamma \left(\nu - \underline{\lambda} \right) = 0 \tag{3–9}$$

$$0 \leq x_j \leq 1 \quad j = 1, \ldots, N \tag{3–10}$$

$$\nu = \sum_{j=1}^{N} \lambda_j x_j. \tag{3–11}$$

Before providing the solution algorithm for the general case, we first analyze a special case.

Identical Revenues. Suppose that $\hat{r}_j = \hat{r} = r - \pi$ for all $j = 1, \ldots, N$, which implies that the (net) revenue term in the objective can be written as $\hat{r}\nu$, and, therefore, that the objective function is a concave function of the single variable ν. Suppose $\hat{r} > 0$ and consider the solution with $x_j = 1$, $j = 1, \ldots, N$, and let $\Lambda = \sum_{j=1}^{N} \lambda_i$. Letting $\gamma^* = 0$ and $\theta^* = \hat{r} + \pi \sum_{i=0}^{B-1} \frac{e^{-\Lambda} \Lambda^i}{i!}$, we can set $\alpha_j = \theta^* \lambda_j$, $\beta_j = 0$, $j = 1, \ldots, N$, and we have a point satisfying the KKT conditions. If, on the other hand, $\hat{r} \leq 0$, then note that any feasible solution must have negative profit, which implies that an optimal solution exists with $x_j = 0$ for all $j = 1, \ldots, N$. Thus, when $\hat{r}_j = \hat{r}$ for all $j = 1, \ldots, N$, an optimal solution selects all items if $\hat{r} > 0$ and selects no items if $\hat{r} \leq 0$. Note that since the optimal relaxation solution is binary, this solves the problem under the binary restrictions as well.

3.4 Solution Approach for Problem [SKPP]

For the general case with item-specific revenue parameters, let us first suppose that $\nu = \underline{\lambda}$ in an optimal solution. The optimal solution in this case can be obtained by

solving a continuous knapsack problem by inserting items into a knapsack of size $\underline{\lambda}$ in nonincreasing order of $\hat{r}_j$ until the knapsack capacity is exhausted. Observe that since $\underline{\lambda} = \min_{j=1,\ldots,N} \lambda_j$, only one item is required to exhaust this capacity, and the optimal solution will set $x_1 = \underline{\lambda}/\lambda_1$.

Next consider the case in which an optimal solution exists that selects more than one item. This implies $\nu > \underline{\lambda}$ and therefore $\gamma = 0$, and we can write the first KKT condition (3–5) as

$$\hat{r}_j - \alpha'_j + \beta'_j = -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu}\nu^i}{i!}, \; j = 1, \ldots, N,$$

where $\alpha'_j = \alpha_j/\lambda_j$ and $\beta'_j = \beta_j/\lambda_j$. Now observe that given any $\nu > 0$, then for any item j such that $\hat{r}_j > -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu}\nu^i}{i!}$, we must have $\alpha_j > 0$, which immediately implies that $x_j = 1$ at a corresponding KKT point. This condition is equivalent to $r_j > \pi \left(1 - \sum_{i=0}^{B-1} \frac{e^{-\nu}\nu^i}{i!}\right) = \pi P\{V(x) \geq B\}$. Because $\pi P\{V(x) \geq B\} \leq \pi$, this implies that for any j such that $r_j > \pi$ (or $\hat{r}_j > 0$), we must have $x_j = 1$ at a KKT point (note that this continues to hold for any $\gamma \geq 0$). Next observe that given any $\nu > 0$, then for any item j such that $\hat{r}_j < -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu}\nu^i}{i!}$, we must have $\beta_j > 0$, which immediately implies that $x_j = 0$ at a corresponding KKT point. This condition is equivalent to $r_j < \pi \left(1 - \sum_{i=0}^{B-1} \frac{e^{-\nu}\nu^i}{i!}\right) = \pi P\{V(x) \geq B\}$. Because $\pi P\{V(x) \geq B\} \geq 0$, this implies that for any j such that $r_j < 0$ (or $\hat{r}_j < -\pi$), we must have $x_j = 0$ at a KKT point. We therefore know that $x_j = 1$ for any j such that $\hat{r}_j > 0$ and $x_j = 0$ for any j such that $\hat{r}_j < -\pi$, and we need only determine the x_j variable values for j such that $-\pi \leq \hat{r}_j \leq 0$. The following proposition enables us to ultimately determine the optimal values of these remaining variables.

Proposition 2. *Suppose an optimal solution exists for [SKPP] such that $x_j > 0$ and that $\hat{r}_k > \hat{r}_j$. Then an optimal solution exists with $x_k = 1$. Conversely, suppose that an optimal solution exists with $x_j < 1$ and that $\hat{r}_k < \hat{r}_j$. Then an optimal solution exists with $x_k = 0$.*

Proof. Suppose an optimal solution exists with $\nu = \nu'$ such that $x_j > 0$ and $x_k < 1$, with $\hat{r}_k > \hat{r}_j$. Because $x_j > 0$, this implies (by (3–5) and (3–6)) that $\hat{r}_j \geq -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu'}\nu'^i}{i!}$,

which then implies $\hat{r}_k > -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu'}\nu'^i}{i!}$. This further implies $\alpha_k > 0$ which, together with $x_k < 1$, violates (3–7) and contradicts the optimality of this solution. Conversely, suppose we have an optimal solution with $\nu = \nu'$ such that $x_j < 1$ and $x_k > 0$. Then we must have $\hat{r}_j \leq -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu'}\nu'^i}{i!}$. If $\hat{r}_k < \hat{r}_j$ we then have $\hat{r}_j < -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu'}\nu'^i}{i!}$, which implies $\beta_j > 0$. This, together with $x_k > 0$, violates (3–8), which contradicts the optimality of the proposed solution. □

Proposition 2 implies a ranked ordering of the attractiveness of items based on the $\hat{r}_j$ (and therefore r_j) values. If we assume that there are not any ties in r_j values, then this, together with the fact that an optimal solution exists with at most one fractional item, suggests a solution method for the relaxation (although this need not necessarily be the unique optimal solution, our methodology that follows ensures finding this optimal solution). That is, suppose that item k takes a fractional value in an optimal solution. Then this implies that $\hat{r}_k = -\pi \sum_{i=0}^{B-1} \frac{e^{-\nu}\nu^i}{i!}$. Moreover, by Proposition 2 we must also have $x_j = 1$ for $j = 1, \ldots, k-1$ and $x_j = 0$ for $j = k+1, \ldots, N$. Let $\nu_1(k) = \sum_{j=1}^{k-1} \lambda_j$. To find the fractional value of x_k we must then solve the following one-dimensional optimization problem:

$$\text{[1D]} \qquad \textbf{Maximize} \qquad \hat{r}_k \lambda_k x_k - \pi \sum_{i=0}^{B} (B-i) \frac{e^{-(\nu_1(k)+\lambda_k x_k)}(\nu_1(k)+\lambda_k x_k)^i}{i!}$$

$$\textbf{Subject to:} \qquad 0 \leq x_k \leq 1.$$

This is a one-dimensional convex program that is easily solved using, for example, bisection search on the interval $[0, 1]$. Note that the derivative of the objective function with respect to x_k equals

$$\left(\hat{r}_k + \pi \sum_{i=0}^{B-1} \frac{e^{-(\nu_1(k)+\lambda_k x_k)}(\nu_1(k)+\lambda_k x_k)^i}{i!} \right) \lambda_k.$$

A stationary point solution is determined by finding the parameter of a Poisson random variable V_k such that

$$P\{V_k \leq B-1\} = -\hat{r}_k/\pi. \tag{3–12}$$

Recall that the only variables to which this one-dimensional optimization problem will apply are those such that $-\pi \leq \hat{r}_j \leq 0$, which implies that the right-hand side of (3–12) will always take a value between 0 and 1. Suppose that the parameter of the Poisson random variable that satisfies (3–12) equals $\hat{\lambda}_k$. This corresponds to a solution in the variable x_k such that

$$\tilde{x}_k = \frac{\hat{\lambda}_k - \nu_1(k)}{\lambda_k}.$$

If $\tilde{x}_k > 1$ then the optimal value of x_k in the one-dimensional optimization problem equals 1; otherwise the optimal value of x_k equals $\max\{0, \tilde{x}_k\}$. If we wish to have a final interval of uncertainty of length ϵ, then we must solve at most N problems of the form [1D], each of which is solved in $\mathcal{O}(\log 1/\epsilon)$, leading to a worst-case complexity of $\mathcal{O}(N \log 1/\epsilon)$ (assuming $1/\epsilon \geq N$ then this term dominates the $\mathcal{O}(N \log N)$ time required for sorting N items).

To summarize our solution algorithm, assuming items are indexed in nonincreasing order of $\hat{r}_j$ values, we first set $x_j = 1$ for all $j : \hat{r}_j > 0$ and set $x_j = 0$ for all $j : \hat{r}_j < -\pi$. Denote $j_{\min}$ and $j_{\max}$, respectively, as the lowest and highest item indices such that $-\pi \leq \hat{r}_j \leq 0$. Then, in increasing order from $k = j_{\min}$ to $k = j_{\max}$, we solve [1D] for x_k and set $x_j = 1$ for all indices lower than k and $x_j = 0$ for all indices higher than k. Note that because the continuous relaxation is a convex program with a linear constraint set (and therefore the KKT conditions are necessary and sufficient for optimality), we can immediately terminate the algorithm upon identifying a KKT point. Recall that we have assumed thus far that no ties exist in $\hat{r}_j$ values. For the continuous relaxation of [SKPP], this assumption is made without loss of generality. That is, if any two items, say j_1 and j_2, have a tie in their $\hat{r}_j$ values, then we may break ties arbitrarily or combine the two items into a single item with parameter value $\lambda_{j_1} + \lambda_{j_2}$ when solving the continuous relaxation (although we cannot combine the variables into one when solving the binary version of the problem).

To solve the binary version of the problem, we use a customized branch-and-bound algorithm that solves the continuous relaxation at each node of the branch-and-bound tree to obtain a good upper bound on the optimal solution at each node. Given a continuous relaxation solution that is not binary at a node, if the continuous relaxation solution value is greater than the best lower bound, we branch on the fractional item and repeat the procedure at each child node. At the child node that forces x_k to zero, we simply exclude item k from the sorted list when solving the continuous relaxation. At the child node that forces x_k to 1, we fix the value of x_k to one, and starting from $j_{\min}$ and going to $j_{\max}$, we apply the same sequential binary search procedure to free items whose values are not yet fixed. Note that at nodes other than the root node, Proposition 2 is not strictly valid for all items, although it still holds among variables whose values are not fixed. At any node that finds a feasible integer solution to the problem we fathom the node and update the lower bound if it exceeds the best lower bound. We terminate the algorithm when we find a KKT point that satisfies the binary restrictions or when all open nodes have been fathomed.

3.5 Computational Study

This section discusses a computational study of our customized branch-and-bound algorithm for solving [SKPP]. The goal of this study is to demonstrate the benefits of our solution approach when compared to a state-of-the-art commercial non-linear integer solver. We implemented our algorithm in Java and performed all tests on a Unix machine with two Pentium 4, 3.2 Ghz processors and 6 GB of RAM. We also modeled [SKPP] in GAMS and solved it using the BARON solver, which is a state-of-the-art commercial solver for general mixed-integer nonlinear problems (clearly the choice of a different benchmark solver will lead to different results; however, as our results show, our algorithm consistently provides optimal solutions in very fast time). We generated a broad set of random test instances for comparing the performance of our algorithm with that of GAMS/BARON.

Table 3-1 summarizes the values of the parameters that we used in generating random problem instances. We generated 24 data sets in order to analyze the effect of the number of items, N, the knapsack capacity, B, and the unit penalty value, π. For each data set, we sampled the value of the expected weight, λ_j, from a uniform distribution on the interval $[0,\ \bar{\lambda}]$, for $\bar{\lambda} = \{0.5, 1, 5, 10\}$, resulting in a total of 96 data sets. We randomly generated 10 instances by sampling the value of the item per unit size, r_j, from a uniform distribution on the interval $[5, 25]$ for each of 96 data sets (for a total of 960 problem instances). These problem instances and parameter values were created to avoid trivial solutions. That is, these parameter values we used ensured the generation of problem instances such that a number of items existed such that $-\pi \leq \hat{r}_j \leq 0$, and, therefore, such that (a) we could not determine the optimal solution of the continuous relaxation by inspection, and (b) the optimal solution to the continuous relaxation was not binary. We provide results from both our customized algorithm implementation and a GAMS/BARON implementation in Table 3-2. Note that for each data set given in Table 3-1, we generated 40 test instances. We limited the running time of GAMS/BARON to 900 CPU seconds. We observed that in each case GAMS/BARON terminated with one of the following three conditions (Rosenthal (2007)):

(1) *Termination with an optimal solution*: The solution found was provably optimal.

(2) *Termination with a resource interruption*: GAMS/BARON could not find a provably optimal solution within the given time limit. The output provided lower and upper bounds for the model at the end of the time limit.

(3) *Termination with an integer solution*: The model terminated with an integer solution before the time limit was reached, presumably due to numerical difficulties with the Poisson distribution function. When the model terminated prematurely with an integer solution, the absolute gap between the lower and upper bound reported by GAMS/BARON was infinity. In other words, GAMS/BARON terminated with an integer solution when it could not find an upper bound, and the solution was, therefore, not

provably optimal (we label such instances as NPO in Table 3-2). The terminating integer solution reported was the lower bound found by GAMS/BARON. Although these solutions were not provably optimal by GAMS/BARON, we were able to compare their solution values to the optimal solution found via our algorithm. We found that these solutions were typically optimal or very close to optimal, with a maximum deviation from optimality of 0.9% among such problem instances.

For each data set from Table 3-1, Table 3-2 shows the percentage of instances (across all values of $\bar{\lambda}$) that were solved to optimality by GAMS/BARON within 900 CPU seconds and the percentage of instances that terminated with integer solutions (not provably optimal). In addition, the table shows the average CPU times of the instances that were solved to optimality and terminated with (not provably optimal) integer solutions (all times are in seconds). We computed the optimality gap of an instance as the difference between upper and lower bounds as a percentage of the best lower bound at the end of 900 CPU seconds for the instances that could not be solved. We provide this average gap in the column labeled "Gap for Unsolved Instances." We computed the gap of an instance that terminated with an integer solution (not provably optimal) as the percentage difference between the objective function value of the integer solution found by GAMS/BARON and the optimal solution we found using our algorithm (since GAMS/BARON was unable to provide a finite upper bound). We provide this gap in the column labeled "Gap for Integer Solutions (NPO)." The average time for our customized branch-and-bound algorithm is provided in the column labeled "Average time" under the "Our Algorithm" column. The table shows that all instances were solved to optimality within a fraction of second when using our algorithm.

All instances in the first data set were solved by GAMS/BARON to optimality in about 85 CPU seconds (on average) while our algorithm found an optimal solution within a fraction of second. In four of the 24 data sets, none of the instances were solved to optimality within 900 seconds using GAMS/BARON. The average optimality gaps of these

instances are significantly high in some data sets, e.g., in sets 5 and 11. We note that in these data sets the difficulty of the problem does not solely depend on the number of items, but also on the combination of the number of items, the knapsack capacity, and the penalty value. For instance, in data set 5, although the number of items was 25 (which is relatively small), the average optimality gap was as high as 42%. In data set 21, however, 67.50% of 40 instances were solved to optimality within a fraction of second, and the average gap of the unsolved instances was 0.46%, although the number of items was 100, which is relatively high.

For problem instances where GAMS/BARON terminated with condition (3), the number of instances that terminated with an integer solution not only depended on the data set from Table 3-2, but also on the value of $\bar{\lambda}$. Figure 3-1 shows the percentage of instances that terminated with condition (3) for the four different $\bar{\lambda}$ values. As illustrated in the figure, for the instances where $\bar{\lambda} = 0.5$, none of the instances terminated with an integer solution, and GAMS/BARON either terminated with an optimal solution or ran for 900 seconds and could not find a provably optimal solution. However, when $\bar{\lambda} = 1$, a total of 42% of the instances terminated with integer solutions. This percentage decreased as the value of $\bar{\lambda}$ increased to 5 and 10, as shown in the figure. As the value of $\bar{\lambda}$ increased, we observed that the percentage of instances GAMS solved optimally increased and the optimality gap for the instances that were not solved also decreased (as we noted previously, as the value of the parameter λ increases, the Poisson distribution tends to a normal distribution, and, in such cases, it may be more effective to model the problem under a normal distribution assumption; see Merzifonluoğlu et al. (2009)). On the other hand, all instances were solved optimally within a fraction of second using our algorithm. As illustrated in Table 3-2, our algorithm outperformed GAMS/BARON across the 960 randomly generated problem instances. Moreover, our algorithm also consistently ensures finding a provably optimal solution, whereas GAMS/BARON ran the full 900 seconds or was unable to determine a finite upper bound in a substantial percentage of instances.

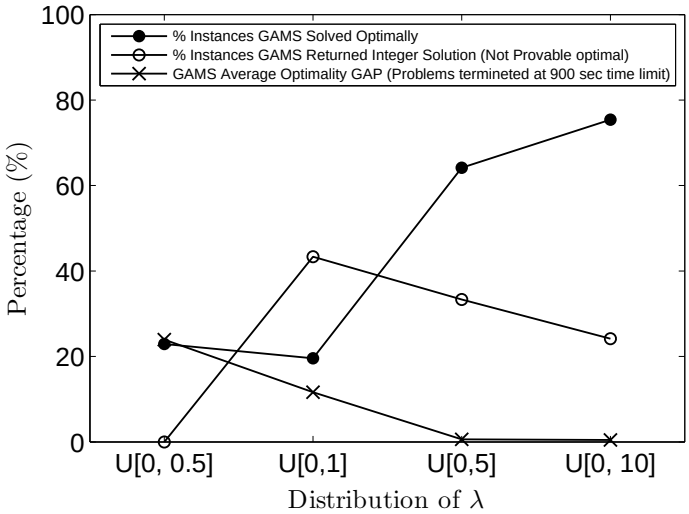

Figure 3-1. Gams/Baron Performance Analysis

Table 3-1. Parameter values for test instances

Data Set	N	B	π	Data Set	N	B	π
1	25	5	20	13	75	5	20
2	25	10	10	14	75	10	10
3	25	10	15	15	75	10	15
4	25	10	20	16	75	10	20
5	25	10	25	17	75	10	25
6	25	15	20	18	75	15	20
7	50	5	20	19	100	5	20
8	50	10	10	20	100	10	10
9	50	10	15	21	100	10	15
10	50	10	20	22	100	10	20
11	50	10	25	23	100	10	25
12	50	15	20	24	100	15	20

Table 3-2. Computational test results

	GAMS/BARON				Our Algorithm
Data Set	% Optimally Solved	% Terminated with Integer Solution (NPO[a])	Avg Time, Optimally Solved Instances	Gap for Unsolved Instances	Average Time
1	100.0%	0.0%	84.450	-	0.065
2	72.5%	27.5%	112.424	-	0.038
3	35.0%	35.0%	0.648	14.20%	0.064
4	10.0%	37.5%	0.662	25.44%	0.086
5	0.0%	55.0%	-	42.43%	0.112
6	5.0%	57.5%	0.467	73.52%	0.115
7	72.5%	0.0%	11.323	6.64%	0.115
8	72.5%	10.0%	3.242	1.09%	0.118
9	50.0%	20.0%	0.456	12.26%	0.147
10	37.5%	15.0%	0.670	22.46%	0.148
11	0.0%	50.0%	-	41.26%	0.258
12	15.0%	47.5%	0.712	37.91%	0.224
13	77.5%	0.0%	8.416	4.66%	0.152
14	57.5%	22.5%	0.513	0.48%	0.132
15	62.5%	5.0%	8.807	1.67%	0.171
16	50.0%	7.5%	0.574	12.98%	0.210
17	0.0%	50.0%	-	29.81%	0.332
18	42.5%	32.5%	0.617	19.58%	0.270
19	87.5%	0.0%	19.854	3.79%	0.146
20	67.5%	7.5%	0.744	0.50%	0.134
21	67.5%	20.0%	0.616	0.46%	0.189
22	50.0%	25.0%	0.645	11.63%	0.218
23	0.0%	50.0%	-	23.02%	0.350
24	60.0%	30.0%	1.096	10.70%	0.312

[a]Not Provably Optimal - Instances terminated with integer solution under condition (3) described above.

CHAPTER 4
A FACILITY LOCATION MODEL WITH SAFETY STOCK COSTS

4.1 Introduction and Motivation

Classical transportation problems determine the assignment of customers to supply facilities in order to minimize total transportation cost while obeying supply limits and meeting (deterministic) customer demands. In the absence of supply capacities, an optimal solution exists for the transportation problem such that each customer's demand is assigned entirely to a single supply facility. The classical uncapacitated facility location problem (UFLP) contains an embedded transportation subproblem, with the addition of fixed costs for open supply facilities (see, e.g., Bilde and Krarup (1977), and Erlenkotter (1978)). Because this problem has a concave cost objective function (such that an extreme point optimal solution exists), we again find that an optimal solution for the UFLP exists such that a given customer's demand is entirely assigned to a single supply facility. More recent work considers practical generalizations of this class of problems that account not only for fixed operating and variable assignment costs, but also for inventory-related costs at facilities. In particular, when we consider contexts with uncertain demands, it is important to consider the impacts of safety stock costs.

Chopra and Meindl (2007) provide illustrations of general trends in supply chain costs as a function of the number of facilities. For example, it is clear that an increase in the number of facilities in a supply chain network results in a corresponding increase in facility costs. Reducing the number of facilities, however, tends to increase outbound transportation costs, which must be balanced against facility and inventory costs. Similarly, Chopra and Meindl (2007) note that an increase in the number of facilities tends to increase total supply chain inventory costs due to the need to increase total system-wide safety stock costs in order to meet customer service level expectations. Conversely, a reduction in the number of facilities that hold safety stock permits a reduction in total safety stock cost as a result of the risk-pooling benefits from aggregating

safety stock in fewer locations. As our results later illustrate, aggregation of safety stock at fewer location is not necessarily required to gain risk-pooling benefits. That is, it is possible to increase the number of supply facilities without a corresponding increase in safety stock cost, even while maintaining a prescribed cycle service level at each facility. As we later discuss, these risk-pooling benefits arise from splitting customer demands among facilities and mixing multiple customer demands within a facility.

Because safety stock costs represent a non-trivial component of overall facility-related costs, recent literature has recognized the need to account for safety stock costs when making facility location decisions (e.g., Shen et al. (2003)). The majority of this work, however, continues to enforce single-sourcing restrictions, which are optimal for the UFLP and uncapacitated transportation problems embedded in these larger inventory-location problems. Unfortunately, safety stock costs cannot be represented, in general, as a linear or concave function of the assignment decision variables. Thus, imposing single-sourcing requirements on such inventory-location problems may be suboptimal when compared to the problem in the absence of this requirement. Our primary goal in this chapter is, therefore, to improve our understanding of the degree of loss that may result from enforcing a single-sourcing requirement.

Clearly there are some benefits to enforcing single-source requirements, although these benefits are typically difficult to quantify. From a practical standpoint, customers often prefer having a single point of contact for delivery and problem resolution. Similarly, suppliers face lower coordination complexity under a single-sourcing arrangement. Algorithmically, heuristic solution approaches are often easier to construct because of the combinatorial nature of solutions to problems that use single-sourcing requirements. In contrast, in the absence of single sourcing, a customer has a built-in backup plan when their demand is split among multiple sources, and one of the sources is unable to deliver. With our goal of understanding the costs of single-sourcing in mind, we address the following problem:

> Given a set of supply facilities, each with some fixed location cost, and a set of customers, each with uncertain demand, determine which supply facilities to open, which customers to assign to which supply facilities and what level of inventory to hold in order to minimize total location, assignment and safety stock costs, while achieving specified service levels and obeying a pre-determined limit on the number of facilities that can supply any given customer.

Note that when the limit on the number of facilities that can supply any given customer equals one, we have the single-sourcing constraint. When this limit equals N (where N is the number of facilities), we effectively have no limit on the number of suppliers that can serve a customer. This problem falls in the class of mixed-integer nonlinear programming problems and is NP-hard (by virtue of generalizing the UFLP). Shen et al. (2003) consider a similar joint location-inventory problem with a single-sourcing requirement that minimizes the cost of facility location, transportation, and holding working process inventory and safety stock. Their model is similar to ours, except that we do not require single sourcing and our model includes a cardinality constraint on the number of sources that can supply a customer. Interestingly, when single-sourcing is required and customer demands are normally distributed, the expression typically used for safety stock cost is concave in the assignment decision variables (when we consider the continuous relaxation of these assignment variables). When single sourcing is not required, however, this expression is instead convex, destroying the concavity of the objective function. Thus, the problem studied by Shen et al. (2003) contains structural properties that are lost when the single-sourcing requirement is dropped. França and Luna (1982) also study a similar problem where demand splitting is allowed (i.e., when a customer's demand may be split among multiple supply facilities). Instead of considering inventory-related costs at the supplier echelon, however, they consider inventory holding and shortage costs at the customer stage, and provide a generalized Benders decomposition algorithm to solve the problem.

In this chapter, we first define and formulate a general model for assigning customers to supply facilities when supplier safety stock costs are considered, demand splitting is

permitted, and customer demand distributions are approximated by a normal distribution (as in Shen et al. (2003). We analyze the special case with zero fixed facility costs, which results in an interesting and practically relevant transportation problem with safety stock costs. We demonstrate important properties of optimal solutions for special cases of this class of transportation problems that, in some cases, lead to closed-form solutions. Moreover, these optimal solution properties provide insight on effective ways to manage risk due to uncertain demand in supply chains. We provide a generalized Benders Decomposition algorithm to solve the general problem with fixed supply-facility operating costs. We then discuss the results of an empirical study intended to characterize the cost of single-sourcing requirements.

The rest of this chapter is organized as follows. Section 4.2 next reviews related literature on location-inventory problems. We define the general problem and model formulation in Section 4.3, and discuss solution methods for special cases in which no fixed cost component exists. Then we present the generalized Benders decomposition algorithm in Section 4.4. Section 4.5 discusses the results of our computational study.

4.2 Literature Review

Since this chapter addresses a location-inventory model, the literature on both facility location and inventory theory is relevant to our work. We thus consider past work in both of these areas, as well as at the intersection of these areas. In the classical facility location problem, the aim is to determine locations of facilities and assignments of retailers to these facilities that minimize the fixed facility location and transportation costs. Thus, the inventory related costs are ignored. We refer the reader to Daskin and Owen (199), Melo et al. (2007), Owen and Daskin (1998), Daskin et al. (2005), and Snyder (2006) for a comprehensive review of facility location problems. On the other hand, inventory theory literature assumes that location decisions have been made beforehand, and, based on this assumption, it evaluates the inventory related decisions. The aim is to find the best

policy that minimizes inventory related costs while providing appropriate service levels at distribution centers or retailers.

Recently, joint location-inventory models have gained attention (see Shen et al. (2003), Vidyarthi et al. (2007), Ozsen et al. (2008a), Ozsen et al. (2008b), Shen and Daskin (2005), Shen (2005), Nozick and Turnquist (1998), Nozick and Turnquist (2001b), Nozick and Turnquist (2001a)). The problem analyzed by Shen et al. (2003) is the most closely related to our work. In particular, Shen et al. (2003) consider a joint location-inventory problem, where multiple retailers−each with stochastic demand−are assigned to distribution centers (DCs). Because of uncertain demand, some amount of safety stock must be carried at distribution centers. In their model, they enforce a single-sourcing requirement, i.e., each customer's demand must be assigned to a single DC. Shu et al. (2005) study a similar problem with one supplier and multiple retailers. Each retailer can serve as a distribution center to achieve risk pooling benefits.

The solution methods applied to these location-inventory models typically depend on the form of the objective function. The form of the objective function, in turn, depends on the decision variable restrictions. For instance, if we have binary assignment variables and an objective function that uses the squared values of these binary variables, then these squared terms can be linearized by simply replacing them with their original binary values (since $x = x^2$ for binary variables). This affects the convexity of the safety stock cost component of the objective function and, therefore, the solution techniques that can be successfully applied. We model our problem as a mixed-integer nonlinear programming problem with continuous assignment variables. We, therefore, need to consider solution techniques relevant to mixed-integer nonlinear programming problems in general, and location-inventory problems in particular.

The majority of past research on location-inventory theory emphasizes the benefits of risk pooling through centralization of inventory, and thus requires such single-sourcing constraints. Recently, Ozsen et al. (2008b) study a logistics system with a single plant, a

set of capacitated warehouses that serve as the direct intermediary between the plant and a set of retailers, each with a stochastic demand. They assume that warehouses order a single type of product from a single plant and carry safety stock to provide appropriate service levels. They relax the single-sourcing requirement. In their model, they assume that each unit of demand of a retailer is randomly assigned to a warehouse among those that are allowed to serve that retailer. The resulting model is an MINLP with an objective function that is neither convex nor concave. They propose a Lagrangian relaxation solution algorithm to solve the model. In this chapter, we also relax the single-sourcing requirement by allowing a customer's demand to be split among supply facilities if it is economical to do so. The main differences of our work from Ozsen et al. (2008b) are that their model has one more stage than our model, at which DCs order products from a single plant and they propose a Lagrangian relaxation algorithm, while we do not account for the order costs from a plant and propose an exact algorithm that uses generalized Benders decomposition.

Lagrangian relaxation based algorithms have been widely used in the location-inventory literature for problems that require single sourcing. Daskin et al. (2002) consider a problem similar to the one addressed in Shen et al. (2003), where they account for both working inventory and safety stock cost terms. They model this problem as a nonlinear integer programming problem with binary assignment variables, and propose a Lagrangian relaxation solution algorithm. Similarly, Sourirajan et al. (2007) apply Lagrangian relaxation to a problem in which a production facility replenishes a single product at multiple retailers. Their model determines the DC locations that minimize total location and inventory costs. Snyder et al. (2007), Ozsen et al. (2008a) and Miranda and Garrido (2006) also propose solution methods based on Lagrangian relaxation for mixed-integer nonlinear models. However, each of these papers assumes that single sourcing is required. Moreover, Lagrangian relaxation based solution methods do not provide strictly better solutions than the continuous relaxation for several important special cases of the problem

we define in this chapter (because of the so-called integrality property; see Geoffrion (1974)).

Several heuristic solution methods have also been proposed in the literature for location-inventory problems. Erlebacher and Meller (2000) consider a problem where products are distributed from plants to DCs and from DCs to retailers. Their aim is to minimize the sum of the fixed operating costs of open DCs, inventory holding costs at DCs, total transportation costs from plants to DCs, and transportation costs from DCs to customers. DCs and customers are located on a grid, and each customer must be assigned to a single DC; thus demand splitting is not allowed. They propose a location-allocation heuristic that uses the better solution obtained using two different approaches. The first approach assigns each customer to its closest DC and then reduces the number of DCs by greedily reassigning customers to other DCs, until reaching a predetermined number of open DCs. The second approach starts by assigning one customer to each open DC (where the number of open DCs equals a predetermined number), and then adds the remaining (unassigned) customers to DCs until all customers are assigned.

The solution method we propose uses generalized Benders decomposition (see Geoffrion (1972)), which has been used effectively for certain classes of mixed-integer nonlinear programming problems. For example, Hoc (1982) considered a transportation and computer communication network design problem with a budget constraint. Hoc (1982) formulated this problem as a mixed-integer nonlinear programming model and proposed an approach using generalized Benders decomposition. França and Luna (1982) also proposed a similar algorithm for a location-inventory problem that is closely related to our work. In their model, they allow backordering with an associated penalty function. Their model considers inventory holding cost at the retail level. In contrast, our model considers inventory costs at the supplier level. The next section formally defines our problem, provides the mathematical model and analyzes two special cases.

4.3 Problem Definition and Mathematical Model

We consider a set $J = \{1, \ldots, N\}$ of potential supply facility locations, indexed by j, such that opening a supply facility at location j results in a fixed cost of F_j for all $j \in J$. We wish to satisfy the demand of a set $I = \{1, \ldots, M\}$ of customers, indexed by i, using some subset of the open facilities. Each customer has a random demand of d_i per time period, and we assume that successive demands in different time increments are independent and identically distributed with mean μ_i and variance σ_i^2. Each supply facility requires achieving a prespecified service level which is supply-facility-dependent. Because customer demands are random, each supply facility carries some amount of safety stock to achieve this service level. The parameters and the decision variables used in the model are as follows.

Parameters

I: set of customers, i.e., $I = \{1, 2, \ldots, M\}$, indexed by i

J: set of supply facilities, i.e., $J = \{1, 2, \ldots, N\}$, indexed by j

c_{ij}: cost of assigning customer i to facility j

$\hat{c}_{ij}$: cost per unit of flow from facility j to customer i

h_j: annual cost of holding a unit of inventory at supply facility j

d_i: random variable for customer i demand per year

μ_i: expected value of d_i

σ_i: standard deviation of d_i

D_j: $\sum_{i \in I} d_i x_{ij}$, i.e., total demand allocated to supply facility j per year

F_j: annualized fixed cost of opening supply facility j

S_j: stock level at supply facility j at the beginning of a year (we assume zero supply lead time)

N_i: maximum number of supply facilities that may serve customer i.

Decision Variables

x_{ij}: proportion of customer i demand allocated to supply facility j

t_{ij}: 1 if any supply is sent to customer i from supply facility j; 0 otherwise

y_j: 1 if supply facility j is opened; 0 otherwise

If we assign the fraction x_{ij} of customer i's demand to supply facility j, then the expected assignment cost equals $c_{ij}x_{ij}$, where $c_{ij} = \hat{c}_{ij}\mu_i$. We assume that all customer demands are independent and normally distributed. Note that the demand seen by supply facility j in a time period has mean $\mu(j) = \sum_{i\in I} \mu_i x_{ij}$ and variance $\sigma^2(j) = \sum_{i\in I} \sigma_i^2 x_{ij}^2$, i.e., $D_j \sim N(\mu(j), \sigma^2(j))$. We assume that supply facility j follows a periodic review inventory policy, and orders up to a stock level S_j at the beginning of every period, such that $\Pr\{D_j \leq S_j\} = \delta_j$; let $z_j^\delta = \frac{S_j - \mu(j)}{\sigma(j)}$ denote the corresponding z value, i.e., $\Phi(z_j^\delta) = \delta_j$. The expected annual safety stock cost at supply facility j is then given by $h_j z_j^\delta \sqrt{\sum_{i\in I} \sigma_i^2 x_{ij}^2}$.

We wish to decide which supply facilities to open and how to allocate the demand of each customer i to at most N_i of these open supply facilities in order to minimize the total expected cost. We formulate this location-inventory problem (ILP) as follows:

$$\textbf{(ILP)} \qquad Z = \text{Minimize} \quad \sum_{j\in J} F_j y_j + \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} + \sum_{j\in J} h_j z_j^\delta \sqrt{\sum_{i\in I} \sigma_i^2 x_{ij}^2} \tag{4–1}$$

$$\text{Subject to} \quad \sum_{j\in J} x_{ij} \geq 1, \qquad \forall i \in I, \tag{4–2}$$

$$\sum_{j\in J} t_{ij} \leq N_i, \qquad \forall i \in I, \tag{4–3}$$

$$0 \leq x_{ij} \leq t_{ij} \leq y_j, \qquad \forall i \in I, j \in J, \tag{4–4}$$

$$y_j, t_{ij} \in \{0, 1\}, \qquad \forall i \in I, j \in J. \tag{4–5}$$

The objective function (4–1) minimizes the sum of the fixed cost of locating supply facilities, the assignment and variable cost from supply facilities to customers, and the safety stock costs. Constraint set (4–2) ensures that each customer's demand is fully assigned to supply facilities. Note that this constraint will be satisfied at equality in an optimal solution. Constraint set (4–3) limits the number of supply facilities that can serve customer i to at most N_i. Constraint set (4–4) permits assigning customer demand only to

open supply facilities, forces y_j to 1 if $t_{ij} = 1$ and t_{ij} to 1 if $x_{ij} > 0$. This constraint also ensures nonnegativity of the assignment proportion variables (x_{ij}'s). Constraint set (4–5) reflects the integrality requirements.

Letting $\phi(x) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{j \in J} h_j z_j^{\delta} \sqrt{\sum_{i \in I} \sigma_i^2 x_{ij}^2}$, the following lemma helps in characterizing the structure of the objective function of (ILP).

Lemma 1. *$\phi(x)$ is convex in x.*

Proof: Let $f_{ij}(x_{ij}) = \sigma_i x_{ij}$ and $F(x) = \sqrt{\sum_{i \in I} [f_{ij}(x_{ij})]^2}$. Now we need to show that $F(x)$ is convex. Let $\vec{F}(x) = [f_{11}(x_{11}), \ldots, f_{ij}(x_{ij})]$. Then $F(x)$ is the l_2 norm of $\vec{F}(x)$, i.e.,$F(x) = \left\| \vec{F}(x) \right\|$.

$$\begin{aligned}
F(\lambda x_1 + (1-\lambda) x_2) &= \left\| \vec{F}(\lambda x_1 + (1-\lambda) x_2) \right\| \\
&= \left\| \lambda \vec{F}(x_1) + (1-\lambda) \vec{F}(x_2) \right\| \quad \text{(because } \vec{F}(x) \text{ is linear in } x_{ij}\text{)} \\
&\leq \left\| \lambda \vec{F}(x_1) \right\| + \left\| (1-\lambda) \vec{F}(x_2) \right\| \quad \text{(triangular inequality)} \\
&= \lambda F(x_1) + (1-\lambda) F(x_2).
\end{aligned}$$

Since h_j and z_j^{δ} are nonnegative constants, $h_j z_j^{\delta} \sqrt{\sum_{i \in I} [f_{ij}(x_{ij})]^2}$ is also convex. Moreover, since the first term of $\phi(x)$ is linear and the second term is the summation of convex functions, $\phi(x)$ is convex in x. □

Lemma 1 implies that (ILP) becomes a convex program, for given y_j and t_{ij} variables. We will use this fact later when constructing a Benders decomposition algorithm. Before discussing a solution technique for the general model, we would like to analyze two special cases of (ILP). Both of these special cases assume that locations are fixed, or equivalently, a fixed value of the vector of y_j variables, which we denote by $\tilde{\mathbf{y}}$ (note that this is equivalent to the assumption of zero fixed costs). These special cases also assume that $N_i = N$ for all $i = 1, \ldots, N$, which permits dropping constraint set (4–3) from the formulation. The resulting problem is an uncapacitated transportation problem with safety stock costs which, to the best of our knowledge, has not been considered in the

literature. While the resulting problem is a convex program (and is therefore readily solved using commercial optimization packages), the analysis of particular special cases of this problem class leads to some interesting structural properties of optimal solutions, and provides insight on managing the tradeoffs between transportation and safety stock costs.

Nonlinear transportation problem. This section analyzes two special cases of the uncapacitated nonlinear transportation problem (where all locations decisions are fixed and no cardinality constraint exists on the number of suppliers that can serve a customer). The first special case assumes identical customer variances and supply-facility-invariant costs, while the second special case considers a two-by-two problem with specially structured assignment and holding costs that lead to a simple closed-form optimal solution.

4.3.1 Identical Supply Costs and Customer Variances

We first consider a special case with identical supply facilities (in terms of supply facility costs) where customer demand variances are identical. For this special case and the one discussed in the following subsection, we assume that locations are fixed, which results in an uncapacitated transportation problem with safety stock costs.

By Lemma 1 we know that the objective function of this special case is a convex function of x. Since all of the constraints of (ILP) are linear in x, the problem with zero fixed costs for facilities is a convex programming problem such that the KKT conditions are necessary and sufficient for optimality for this special case (note that any feasible solution such that $\sum_{i \in I} x_{ij} = 0$ violates the differentiability assumption required for application of the KKT conditions at the associated point; however, we are able to consider such solutions separately in our analysis).

For this special case, we assume the assignment cost is customer-specific and equal to c_i for customer i, i.e., $c_{ij} = c_i$ for all $j \in J$ and for each customer i. We will refer to cases in which transportation costs are facility invariant as cases with symmetric transportation costs. We also assume that the supply facility unit holding costs and required cycle service

levels are identical for all supply facilities, and that all customer demand variances are equal, i.e., $h_j = h$ and $z_j^\delta = z^\delta$ for all $j \in J$ and $\sigma_i^2 = \sigma^2$ for all $i \in I$. Letting μ and β denote the vectors of KKT multipliers for the assignment constraints (4–2) and nonnegativity constraints on the x_{ij} variables, we next analyze the KKT conditions for this special case, which can be written as follows.

$$c_i + hz^\delta \sigma \frac{x_{ij}}{\sqrt{\sum_{i \in I} x_{ij}^2}} - \mu_i - \beta_{ij} = 0, \qquad \forall i \in I, j \in J, \tag{4–6}$$

$$\mu_i \left(1 - \sum_{j \in J} x_{ij}\right) = 0, \qquad \forall i \in I, \tag{4–7}$$

$$\beta_{ij} x_{ij} = 0, \qquad \forall i \in I, j \in J, \tag{4–8}$$

$$\sum_{j \in J} x_{ij} \geq 1, \qquad \forall i \in I, \tag{4–9}$$

$$\mu_i, \beta_{ij}, x_{ij} \geq 0, \qquad \forall i \in I, j \in J. \tag{4–10}$$

Given a solution and any supply facility j, let $I(j)$ denote the set of customers such that $x_{ij} > 0$. Similarly, denote $J(i)$ as the set of facilities such that $x_{ij} > 0$. The following theorem characterizes the structure of optimal solutions for this special case.

Theorem 5. *Any feasible solution such that*

1. $x_{ij} = \frac{1}{\omega_j}$ *for some finite* $\omega_j \geq 1$ $\forall j \in J$, $i \in I(j)$ *(with* $x_{ij} = 0$ $\forall i \notin I(j)$*); and*
2. $\sum_{j \in J(i)} \frac{1}{\omega_j} = 1$ *for all* $i \in I$

satisfies the KKT conditions, and is therefore optimal for the special case we have described.

Proof: When $|I(j)| = 0$, no customers are assigned to supply facility j. Without loss of generality, we assume that this supply facility is not open and we exclude this supply facility from consideration in our problem. Therefore we consider the KKT conditions for $j \in J$ such that $|I(j)| > 0$. Clearly each x_{ij} is between 0 and 1. Since $\frac{1}{\omega_j} > 0$, using condition (4–8) we set $\beta_{ij} = 0$ for all $i \in I(j)$. From condition (4–6), we require

$$c_i + hz\sigma \frac{1/\omega_j}{\sqrt{|I(j)|/\omega_j^2}} - \mu_i = 0, \quad \forall j \in J, i \in I(j),$$
$$\Rightarrow \mu_i = \frac{hz\sigma}{\sqrt{|I(j)|}} + c_i, \quad \forall j \in J, i \in I(j).$$

Thus we have $\mu_i \geq 0$ for all $j \in J$ and $i \in I(j)$. For each $i \notin I(j)$ we set $x_{ij} = 0$ and $\beta_{ij} = c_i$, which ensures that (4–6) holds for all $i \in I$ and $j \in J$. We have therefore constructed a solution satisfying (4–6), (4–8), and (4–10). By assumption we have $\sum_{j \in J(i)} \frac{1}{\omega_j} = 1$ for all $i \in I$, which implies that (4–7) and (4–9) hold, and all KKT conditions are satisfied by the solution we have constructed. □

Theorem 5 implies that any *balanced* solution is optimal under identical supply costs and identical customer variance values. That is, provided that all customers assigned to a supply facility have an equal fraction of their expected demand allocated to the supply facility, the solution is optimal. Thus, for example, an optimal solution exists such that all customers are assigned to a single supply facility, which is consistent with the well known use of inventory aggregation to obtain safety stock risk pooling benefits. Theorem 5 illustrates that we can obtain the same degree of risk pooling benefits in a number of different ways, without requiring inventory aggregation. That is, given a problem with N facilities and N customers, for example, a solution such that all N facilities are open, and $\frac{1}{N}$ of each customer's demand is allocated to each open facility achieves the same degree of risk pooling benefits of aggregating all inventory at a single facility (for this special case). This illustrates the fact that one can achieve risk pooling benefits without physical aggregation by splitting customers' demands among different facilities, and mixing the demands of multiple customers within a facility. Clearly, when accounting for fixed costs of identical facilities, the solution that aggregates all customers at one facility (a single-sourcing solution) is preferred when all customers and facility costs are identical. When neither facilities nor customers are identical, however, solutions that require single

sourcing are often suboptimal, as we later show in our computational results section, and as the special case discussed in the following subsection illustrates.

4.3.2 Specially Structured Assignment and Holding Costs

We next consider a specially structured case with two suppliers and two customers. For this special case, we assume that facility holding costs and service levels are equal, as are customer variances, i.e., $h_j = h$ and $z_j^\delta = z^\delta$ for $j = 1, 2$ and $\sigma_i^2 = \sigma^2$ for $i = 1, 2$. Then, letting $H = hz^\delta\sigma$, we assume the following assignment cost relationship holds for some value α between 0 and 1:

$$c_{11} = c_{12} + Hg(\alpha), \tag{4–11}$$

$$c_{22} = c_{21} + Hg(\alpha), \tag{4–12}$$

where $g(\alpha) = \frac{(1-2\alpha)}{\sqrt{\alpha^2+(1-\alpha)^2}}$. Note that this permits values of $c_{11} \in [c_{12} - H, c_{12} + H]$ and $c_{22} \in [c_{21} - H, c_{21} + H]$. Observe that when $\alpha = \frac{1}{2}$ we have a symmetric transportation cost instance with $c_{11} = c_{12}$ and $c_{22} = c_{21}$, which results in the special case in which assignment costs are facility independent (as in the special case discussed in the previous subsection).

For the two-by-two special case in which facility holding costs and customer variances are equal, and assignment costs obey (4–11) and (4–12), we have the following proposition.

Proposition 3. *For a two-supplier, two-customer problem instance with identical supplier holding costs and service levels, and identical customer variances, when the assignment costs obey the relationships (4–11) and (4–12), an optimal solution exists such that* $x_{11} = x_{22} = \alpha$ *and* $x_{12} = x_{21} = 1 - \alpha$*, with minimum cost* $c_{11} + c_{22} + 2H\frac{\alpha}{\sqrt{\alpha^2+(1-\alpha)^2}} = c_{12} + c_{21} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2+(1-\alpha)^2}}$.

Proof: Please see Appendix B.

Observe that when $\alpha = \frac{1}{2}$, the symmetric cost case, the optimal cost equals $c_{12} + c_{22} + \sqrt{2}H = c_{11} + c_{21} + \sqrt{2}H = c_{11} + c_{22} + \sqrt{2}H = c_{12} + c_{21} + \sqrt{2}H$. In this case, any one of the following solutions is optimal: $(x_{11}, x_{12}, x_{21}, x_{22}) = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)$; $(x_{11}, x_{12}, x_{21}, x_{22}) = (0, 1, 0, 1)$; $(x_{11}, x_{12}, x_{21}, x_{22}) = (1, 0, 1, 0)$. This case is consistent

with the special case covered in the previous section, where an optimal solution exists that allocates $\frac{1}{\omega_j}$ of each customer's demand to each active facility, where ω_j is the number of customers assigned to facility j. When $\alpha = 1$ ($\alpha = 0$), an optimal solution sets $(x_{11}, x_{12}, x_{21}, x_{22}) = (1, 0, 0, 1)$ ($(x_{11}, x_{12}, x_{21}, x_{22}) = (0, 1, 1, 0)$) with an optimal cost of $c_{11} + c_{22} + 2H$ ($c_{12} + c_{21} + 2H$). In this case, the difference in transportation cost does not offset any benefits from risk pooling. While in each of the cases with $\alpha \in \left\{0, \frac{1}{2}, 1\right\}$ an optimal single-sourcing solution exists, the following corollary shows that this is not the case for the remaining values of α on the interval $[0, 1]$.

Corollary 1. *For the two-supplier, two-customer problem class we have described, the objective function value of the minimum-cost single-sourcing solution minus that of the minimum-cost solution with customer demand splitting equals $H \times \rho(\alpha)$, where* $\rho(\alpha) = \min\left\{2\left(1 - \frac{\max\{\alpha, 1-\alpha\}}{\sqrt{\alpha^2+(1-\alpha)^2}}\right); \sqrt{2} - \frac{1}{\sqrt{\alpha^2+(1-\alpha)^2}}\right\}$.

Proof: Please see Appendix B.

Figure 4-1 illustrates the value of $\rho(\alpha)$ for $\alpha \in [0, 1]$. We can show that the peak values occur at the values of α such that the terms in the minimum operator given in the corollary are equal. This occurs at $\alpha = 0.2725$ and $\alpha = 0.7275$, where $\rho(\alpha) = 12.7\%$. At either of these values of α the minimum cost single-sourcing solution exceeds the minimum possible cost by $0.127H$, while the actual percentage cost increase associated with single sourcing depends on the transportation and holding cost parameter values. This analysis illustrates the fact that single-sourcing solutions are either optimal or close-to-optimal when transportation costs are symmetric (as is the case when $\alpha = \frac{1}{2}$) or severely asymmetric (as is the case when $\alpha = 0$ or 1). In the former case, multiple optimal solutions exist (using either one or two facilities) while in the latter case, a single optimal solution exists that uses the dominant facility (in terms of lower transportation costs). For intermediate cases, however (when transportation costs are neither symmetric nor grossly asymmetric), we see that the cost performance of a single-sourcing strategy can be worse than a demand splitting strategy by a non-trivial amount. Our computational tests on the

general model with location decisions (and associated costs), presented later in Section 5, illustrate this phenomenon further, by showing cost increases associated with single sourcing on the order of $2-6\%$.

Next consider an asymmetric transportation cost case in which $c_{12} = c_{21} = \tilde{c}$, such that $c_{11} = c_{22} = \tilde{c} + Hg(\alpha)$. Note that in this case, the average value of c_{ij}, which we denote by $E[c]$, equals $\tilde{c} + \frac{Hg(\alpha)}{2}$. We are interested in how the maximum percentage cost savings from demand splitting (relative to single sourcing) behaves as a function of the ratio of the average assignment cost to holding cost, which we denote by $E[c/h]$. For a fixed value of h, we then have $E[c/h] = \frac{\tilde{c}}{h} + \frac{Hg(\alpha)}{2h}$. Let us consider a value of α such that the optimal single sourcing solution sets $x_{12} = x_{21} = 1$ and $x_{11} = x_{22} = 0$, which we can show occurs for an α value in the interval $[0, 0.2725]$. We therefore assume $\alpha = 0.25$. Note that for this case, the minimum cost solution gives an objective function value of $z^{opt} = c_{12} + c_{21} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2+(1-\alpha)^2}} = 2\tilde{c} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2+(1-\alpha)^2}}$. The difference in cost between the minimum cost single-sourcing solution and the minimum cost solution with demand splitting is $\Delta = 2H\left(1 - \frac{1-\alpha}{\sqrt{\alpha^2+(1-\alpha)^2}}\right)$. The percentage cost difference can then be written as $\frac{\Delta}{z^{opt}} = \frac{H(1-f(\alpha))}{\tilde{c}+Hf(\alpha)}$, where $f(\alpha) = \frac{1-\alpha}{\sqrt{\alpha^2+(1-\alpha)^2}}$. We then have

$$\frac{\Delta}{z^{opt}} = \frac{H(1-f(\alpha))}{\tilde{c}+Hf(\alpha)} = \frac{z^{\delta}\sigma(1-f(\alpha))}{E[c/h] + \frac{z^{\delta}\sigma\alpha}{\sqrt{\alpha^2+(1-\alpha)^2}}} = \frac{z^{\delta}\sigma(1-f(\alpha))}{E[c/h] + z^{\delta}\sigma\tilde{f}(\alpha)},$$

where $\tilde{f}(\alpha) = \frac{1}{2\sqrt{\alpha^2+(1-\alpha)^2}}$. If we assume $z^{\delta} = 2$ and $\sigma = 500$ (note that the magnitude of σ does not affect the percentage savings values, but scales the relevant values of $E[c/h]$ at which these magnitudes are achieved), then Figure 4-2 illustrates the behavior of $\frac{\Delta}{z^{opt}}$ as $E[c/h]$ varies (note that because we have an asymmetric cost case with $c_{12} = c_{21} = \tilde{c}$ and $c_{11} = c_{22} = \tilde{c} + Hg(\alpha)$, the minimum possible value of $E[c/h]$, which occurs at $\tilde{c} = 0$, equals $\frac{z^{\delta}\sigma g(\alpha)}{2}$). Figure 4-2 illustrates the following. For a fixed value of h, as we increase c_{ij} values, the assignment costs dominate, and the problem approaches the standard uncapacitated facility location problem (in this case, a single-sourcing solution

is optimal among all solutions). Similarly, as holding cost increases, and c_{ij} values remain asymmetric, the benefits of risk pooling (by mixing customer demands at each location) begin to dominate and lead to increased savings from mixing customer demands within a facility and splitting customer demands between facilities. Our computational results, discussed later, illustrate a similar behavior as $E[c/h]$ increases.

Note however, that when we permit $E[c/h]$ to go to zero (as we do in our computational tests), c_{ij} values become increasingly symmetric, and $\frac{\Delta}{z^{opt}}$ decreases to zero as $E[c/h]$ goes to zero (our computational test results later illustrate this phenomenon as well). We next discuss a generalized Benders decomposition approach that permits efficiently solving problems from the more general (and difficult) problem class (ILP).

4.4 A Generalized Benders Decomposition Approach for (ILP)

This section returns to the general (ILP) model and provides an effective solution approach for this problem class. Recall that for a fixed location vector $\tilde{\mathbf{y}}$ and a feasible binary assignment vector $\tilde{\mathbf{t}}$, from Lemma (1), we know that the remaining problem is a convex program. Let us temporarily fix the location vector at $\tilde{\mathbf{y}}$ and the binary assignment vector at $\tilde{\mathbf{t}}$, such that constraints (4–3), (4–4) and (4–5) admit a feasible solution in the x_{ij} variables. Then the associated restricted problem becomes

$$
\begin{aligned}
\textbf{(ILP}(\tilde{\mathbf{t}},\tilde{\mathbf{y}})\textbf{)} \qquad & \text{Minimize} && \sum_{j\in J} F_j\tilde{y}_j + \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} + \sum_{j\in J} h_j z_j^{\delta}\sqrt{\sum_{i\in I}\sigma_i^2 x_{ij}^2} \\
& \text{Subject to} && \sum_{j\in J} x_{ij} \geq 1, \qquad \forall i \in I, \\
& && 0 \leq x_{ij} \leq \tilde{t}_{ij}, \qquad \forall i \in I, j \in J. \qquad (4\text{–}13)
\end{aligned}
$$

Note that the fixed-charge component, $\sum_{j\in J} F_j\tilde{y}_j$, in the objective function is a constant for a given vector $\tilde{\mathbf{y}}$. Similarly, the right-hand-side value of each constraint in set (4–13) is either 0 or 1, depending on the value of $\tilde{t}_{ij}$. We also note that **(ILP($\tilde{\mathbf{t}}$, $\tilde{\mathbf{y}}$))** is feasible if and only if $\sum_{j\in J}\tilde{t}_{ij} \geq 1$ for all $i \in I$.

We can then write our original problem (ILP) in the space of the vector of t_{ij} and y_j variables as follows:

$$
\begin{array}{llll}
\textbf{(ILP}'\textbf{)} & \text{Minimize} & \displaystyle\sum_{j\in J} F_j y_j + v(t) & \\
 & \text{Subject to} & \displaystyle\sum_{j\in J} t_{ij} \le N_i, & \forall i \in I, \\
 & & \displaystyle\sum_{j\in J} t_{ij} \ge 1, & \forall i \in I, \\
 & & t_{ij} \le y_j, & \forall i \in I, j \in J, \\
 & & t_{ij}, y_j \in \{0,1\}, & \forall i \in I, j \in J,
\end{array}
$$

where, for any given vector $\mathbf{t}$, the value $v(\mathbf{t})$ is determined by the subproblem (ILSP) as follows:

$$
\begin{array}{llll}
(\textbf{ILSP}) & v(\mathbf{t}) = \text{Min} & \displaystyle\sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} + \sum_{j\in J} h_j z_j^{\delta} \sqrt{\sum_{i\in I} \sigma_i^2 x_{ij}^2} & \\
 & \text{Subject to} & \displaystyle\sum_{j\in J} x_{ij} \ge 1, & \forall i \in I, \\
 & & 0 \le x_{ij} \le t_{ij}, & \forall i \in I, j \in J.
\end{array}
$$

(Note that the second constraint set in ILP$'$ ensures feasibility of the subproblem ILSP.) Since (ILSP) is a convex programming problem with linear constraints when the $\mathbf{t}$ vector is fixed, its KKT conditions are necessary and sufficient for optimality (note that since the square root function is not differentiable at zero, the KKT conditions do not apply at this single point; if, however, we consider the approximate problem with each square root term replaced by $\sqrt{\epsilon + \sum_{i\in I} \sigma_i^2 x_{ij}^2}$, for arbitrarily small $\epsilon > 0$, then the KKT conditions are necessary and sufficient for this approximate problem). Problem (ILSP) is therefore amenable to dualization techniques, and its optimal dual solution value equals the optimal primal solution value. Define the vectors of dual variables $\mu = (\mu_1, \ldots, \mu_m) \ge 0$ and $\lambda = (\lambda_{11}, \ldots, \lambda_{mn}) \ge 0$ corresponding to the two constraint sets in (ILSP). Then we can

write the Lagrangian dual problem as

$$v(\mathbf{t}) = \max_{\mu\geq 0,\lambda\geq 0}\left[\min_{x\geq 0}\left[\phi(x) + \sum_{i\in I}\mu_i(1-\sum_{j\in J}x_{ij}) + \sum_{i\in I}\sum_{j\in J}\lambda_{ij}(x_{ij}-t_{ij})\right]\right]$$

$$\text{for every } \mathbf{t}\in\{0,1\}^{M\times N}, \tag{4–14}$$

where $\phi(x) = \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} + \sum_{j\in J} h_j z_j^{\delta}\sqrt{\sum_{i\in I}\sigma_i^2 x_{ij}^2}$.

Problem (ILP) is therefore equivalent to the following Master Problem (MP):

$$\begin{aligned}
(\mathbf{MP}) \qquad & \text{Minimize} && \sum_{j\in J} F_j y_j + \theta \\
& \text{Subject to} && \theta \geq \min_{x\geq 0}[\phi(x) + \sum_{i\in I}\mu_i(1-\sum_{j\in J}x_{ij}) + \sum_{i\in I}\sum_{j\in J}\lambda_{ij}(x_{ij}-t_{ij})], \\
& && \qquad\qquad \forall \mu\geq 0, \lambda\geq 0, && (4\text{–}15) \\
& && \sum_{j\in J} t_{ij} \leq N_i, \qquad \forall i\in I, && (4\text{–}16) \\
& && \sum_{j\in J} t_{ij} \geq 1, \qquad \forall i\in I, \\
& && t_{ij}\leq y_j, \qquad \forall i\in I, j\in J, \\
& && t_{ij}, y_j \in \{0,1\}, \qquad \forall i\in I, j\in J, \\
& && \theta\geq 0.
\end{aligned}$$

Clearly we cannot write the above formulation with a constraint of the form (4–15) for all possible values of μ and λ. We therefore generate valid cuts successively that correspond to specific values of the vectors μ and λ and add them to the formulation in an iterative fashion (such cuts are generally referred to as Benders cuts). Given a particular binary vector $\mathbf{t}^k$ we can solve the convex programming problem (ILSP) and recover corresponding optimal dual multiplier vectors μ^k and λ^k. We can then write

$$v(\mathbf{t}^k) = \min_{x\geq 0}\left[\sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} + \sum_{j\in J} h_j z_j^{\delta}\sqrt{\sum_{i\in I}\sigma_i^2 x_{ij}^2} + \sum_{i\in I}\mu_i^k(1-\sum_{j\in J}x_{ij}) + \sum_{i\in I}\sum_{j\in J}\lambda_{ij}^k(x_{ij}-t_{ij}^k)\right]$$

$$= \sum_{i \in I} \mu_i^k - \sum_{i \in I} \sum_{j \in J} \lambda_{ij}^k t_{ij}^k$$

$$+ \min_{x \geq 0} \left[\sum_{i \in I} \sum_{j \in J} (c_{ij} + \lambda_{ij}^k - \mu_i^k) x_{ij} + \sum_{j \in J} h_j z_j^\delta \sqrt{\sum_{i \in I} \sigma_i^2 x_{ij}^2} \right]. \tag{4–17}$$

We therefore have that $\min_{x \geq 0} \left[\sum_{i \in I} \sum_{j \in J} (c_{ij} + \lambda_{ij}^k - \mu_i^k) x_{ij} + \sum_{j \in J} h_j z_j^\delta \sqrt{\sum_{i \in I} \sigma_i^2 x_{ij}^2} \right] = v(\mathbf{t}^k) - \sum_{i \in I} \mu_i^k + \sum_{i \in I} \sum_{j \in J} \lambda_{ij}^k t_{ij}^k$. Substituting this in (4–15) provides the following Benders cut for (MP) corresponding to the dual multipliers μ^k and λ^k

$$\theta \geq v(\mathbf{t}^k) - \sum_{i \in I} \sum_{j \in J} \lambda_{ij}^k (t_{ij} - t_{ij}^k). \tag{4–18}$$

Our Relaxed Master Problem (RMP) then becomes

$$\begin{array}{llll}
(\mathbf{RMP}) & \text{Minimize} & \displaystyle\sum_{j \in J} F_j y_j + \theta & \\
 & \text{Subject to} & \theta \geq v(\mathbf{t}^k) - \displaystyle\sum_{i \in I} \sum_{j \in J} \lambda_{ij}^k (t_{ij} - t_{ij}^k), & \forall k = 1, \ldots, K, \\
 & & \displaystyle\sum_{j \in J} t_{ij} \leq N_i, & \forall i \in I, \\
 & & \displaystyle\sum_{j \in J} t_{ij} \geq 1, & \forall i \in I, \\
 & & t_{ij} \leq y_j, & \forall i \in I, j \in J, \\
 & & t_{ij}, y_j \in \{0, 1\}, & \forall i \in I, j \in J, \\
 & & \theta \geq 0, &
\end{array}$$

where K denotes the number of Benders cuts we have generated. For a given $\mathbf{t}^k$ vector, the above Benders cut implicitly accounts for all constraints of the form of (4–15) (for all possible μ and λ), because λ^k and μ^k maximize $v(\mathbf{t}^k)$ over all μ and λ. Note that the (RMP) formulation is a 0-1 integer program. At each iteration, we solve the RMP to obtain a (possibly) new $\mathbf{t}^k$ vector. Given this $\mathbf{t}^k$ vector, we then solve the subproblem (ILSP) to determine the corresponding optimal dual (KKT) multiplier values. We then

add the new constraint (4–18) to the RMP formulation. If the value of θ at the previous iteration does not violate this new cut at the previous $\mathbf{t}^k$, then the current solution is optimal. Otherwise we re-solve RMP and repeat this procedure until the same $\mathbf{t}^k$ vector is optimal in successive iterations. In the worst case, if we were to generate a constraint of the form of (4–18) for all possible $\mathbf{t}$ vectors, the resulting RMP formulation would be equivalent to MP. In practice, however, a relatively small number of such cuts are needed to find an optimal solution. We next formalize the algorithm as follows.

Step 1: Choose an initial pair of vectors $\mathbf{y}^0$ and $\mathbf{t}^0$ that ensure a feasible solution for ILSP and select an optimality tolerance ϵ. Solve ILSP at $\mathbf{t} = \mathbf{t}^0$, obtaining $\mathbf{x}^0$ and corresponding optimal μ^0 and λ^0 vectors. Set $UB = \sum_{j \in J} F_j y_j^0 + v(\mathbf{t}^0)$ and let $(\bar{\mathbf{x}}, \bar{\mathbf{t}}, \bar{\mathbf{y}}) = (\mathbf{x}^0, \mathbf{t}^0, \mathbf{y}^0)$ denote the initial incumbent solution.

Step 2: Solve the RMP with all previously generated cuts. Let $(\theta^*, \mathbf{t}^*)$ denote an optimal solution to RMP, and let $LB = \theta^* + \sum_{j \in J} F_j y_j^*$. If $UB - LB < \epsilon$, stop.

Step 3: Solve ILSP at $\mathbf{t} = \mathbf{t}^*$, denoting $\mathbf{x}^*$ as the optimal solution vector and $v(\mathbf{t}^*)$ as the optimal solution value. If $\sum_{j \in J} F_j y_j^* + v(\mathbf{t}^*) < UB$, set $UB = \sum_{j \in J} F_j y_j^* + v(\mathbf{t}^*)$ and update the incumbent solution, i.e., let $(\bar{\mathbf{x}}, \bar{\mathbf{t}}, \bar{\mathbf{y}}) = (\mathbf{x}^*, \mathbf{t}^*, \mathbf{y}^*)$. If $UB - LB < \epsilon$, stop; $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is a ϵ-optimal solution. Otherwise, recover the optimal dual multiplier vectors μ^* and λ^*, add the corresponding cut (4–18) to the RMP formulation and return to Step 2.

4.5 Computational Results

This section provides the results of a computational study of problem (ILP) intended to provide a characterization of the incremental costs that result from a single-sourcing strategy. In particular, we would like to characterize the percentage difference in the costs of problem instances when single sourcing is enforced relative to the case in which demand splitting is allowed. With this goal in mind, we conducted a broad set of computational tests using a range of parameter settings, and then compared the results that we obtained for both problems.

We first attempted to solve the problem (ILP) using GAMS/BARON, a commercial mixed-integer nonlinear solver. Using a set of ten randomly generated problem instances containing five supply facilities and five customers, we were not able to obtain an optimal solution within 1800 seconds for any of the problem instances. The optimality gaps in most of these problem instances after running for 1800 seconds were approximately 25% (with the smallest such gap being 18%). Therefore, we developed code for our generalized Benders decomposition algorithm described in the previous section for application to problem (ILP) in order to solve problems of reasonable size.

We implemented our Benders decomposition algorithm using GAMS 22.6 running on a Unix machine with two Pentium 4, 3.2 Ghz processors (with 1M cache) and 6 GB of RAM. We used CPLEX 11 for solving the 0-1 integer programming master problem (RMP) and CONOPT2 version 2.071K-010-061 for solving the convex programming subproblems (ILSP). All of our test problems used $M = 10$ customers and $N = 5$ supply facilities, which is the maximum problem size that we could consistently solve within 1200 seconds in GAMS.

The limit on the number of supply facilities that can serve each customer, i.e., N_i for customer $i \in I$, is an important parameter for our model. Since the maximum number of supply facilities for all instances was 5, we parametrically varied N_i between 1 and 5 for each problem instance (and used the same value of N_i for each customer). Obviously, when we set N_i to 1 for each customer $i \in I$, we obtain an optimal solution for the problem with single-sourcing requirements. Let Z_k be the optimal objective function value when $N_i = k$. Our main goal is to analyze the effect of different parameters on the percentage difference between the minimum cost when demand splitting is allowed and when single sourcing is imposed. We therefore calculated the percentage difference, ΔZ_k, as $\Delta Z_k = (Z_1 - Z_k)/Z_k$ for $k = 1, \ldots, 5$ and for each set of parameter values. Note that ΔZ_5 characterizes the percentage cost difference between the single-sourcing case and the case in which demand splitting is unrestricted.

As we noted in Section 3, the relative values of average assignment cost and holding cost play important roles in our model. We would therefore like to analyze the effect of these parameters simultaneously. Since these parameters tend to have opposite effects on the relative cost difference ΔZ_5, we analyze the effect of the expected value of the ratio of the (per unit) assignment cost to the holding cost, i.e. $E[\hat{c}/h]$. We used 5 different parameter settings for $E[\hat{c}]$, i.e., 0.3, 0.4, 0.5, 0.6, and 0.7 and we set the holding cost equal to 1 for all facilities (therefore $E[\hat{c}/h] = E[\hat{c}]$). The individual c_{ij} values were randomly generated from a uniform distribution that ensures the prescribed value of $E[\hat{c}/h]$. Table 4-1 provides the uniform distribution parameters for each setting of $E[\hat{c}/h]$. Our choice of values of $E[\hat{c}]$ was based on the fact that in practice, the holding cost is often a percentage of the total value of an item. That is, suppose $h = ic'$, where i is a percentage holding cost rate (often between 15% and 25%) and c' is the item's value. Next, suppose $\hat{c} = \hat{\imath}c'$, i.e., where $\hat{\imath}$ reflects the percentage of total value that constitutes transportation cost. Then, for example, if $\hat{\imath} = 10\%$, and $i = 20\%$, we have $\hat{c} = 0.5$.

The ratio of the standard deviation to the mean demand is another important parameter that affects the cost performance of single-sourcing relative to demand splitting. Since the standard deviation affects the magnitude of safety stock holding cost and the mean affects the magnitude of assignment costs, instead of analyzing the effects of these two parameters separately, we analyzed their ratio, i.e., the coefficient of variation (CoV= σ/μ) of demand. We randomly generated mean demands between 4000 and 6000 and used 3 different values for CoV, 0.35, 0.40, and 0.45, to determine the associated standard deviation values.

The other important parameter affecting cost performance is the fixed cost of a supply facility. A high fixed cost decreases the number of open supply facilities, which in turn affects the assignment of customers to supply facilities. We randomly generated four different data sets for F_j values from the uniform distributions shown in Table 4-1.

While these values of fixed costs may appear relatively small, these values might reflect the portion of fixed cost that is allocated to the single product in question. Clearly, as our results later show, higher fixed costs lead to a choice of fewer facilities. In such cases, the difference in cost between an optimal single-sourcing strategy and an optimal demand-splitting strategy will naturally decrease.

By using the cross combinations of these three parameter settings, i.e., $E[\hat{c}/h]$, CoV, and F_j, we generated 600 ($5 \times 4 \times 3$) different data sets. For each data set we generated 10 random test instances, resulting 6000 test instances in total. We set the service level to 97.5% ($z^{\delta} = 1.96$) for all test instances.

First, we analyzed the effect of $E[\hat{c}/h]$. Table 4-2 summarizes the results for different values of $E[\hat{c}/h]$. We provide the maximum and minimum values of ΔZ_5 from among the 6000 instances in the columns labeled **max** and **min**, respectively, with the average value in the column labeled **average**.

The highest percentage difference obtained among 6000 instances equals 6.82%. The minimum percentage difference is 0%, which means that in some of the cases a single-sourcing solution is optimal even though single sourcing is not enforced. The most remarkable row in Table 4-2 is the one corresponding to $E[\hat{c}/h] = 0.5$. The minimum percentage difference among the 1200 test instances with $E[\hat{c}/h] = 0.5$ is 2.39%. This means that in none of these 1200 instances was single-sourcing optimal. The effect of $E[\hat{c}/h]$ on the percentage gap is interesting. As seen in Figure 4-3, both low and high levels of $E[\hat{c}/h]$ lead to the optimality of single-sourcing solutions.

At higher levels of $E[\hat{c}/h]$, the problem becomes similar to an uncapacitated facility location problem, where single sourcing is optimal. Also, at lower levels of $E[c/h]$, the facility and safety stock costs dominate the objective function. In the presence of fixed facility location costs, the model reduces the number of facilities and uses aggregation to obtain risk pooling benefits. However, at intermediate values of the ratio of the transportation cost to the holding cost, the model seeks to reduce transportation costs

by utilizing more locations, and simultaneously benefits from risk pooling by mixing the demands of multiple customers at the open locations. This illustrates the fact that even in the presence of fixed facility costs, the benefits of deviating from a single-sourcing policy are non-negligible. However, when the facility and/or transportation costs dominate, the best single-sourcing solution value approaches the optimal solution value.

We illustrate the average value of ΔZ_k for different values of k (where $k = N_i$ for each $i \in I$) in Figure 4-4. As can be seen from Figure 4-4, when there is no limit on the number of facilities that can supply any customer, i.e., when $N_i = 5$, an optimal solution assigns customers to at most 3 different supply facilities. In the majority of cases, assigning each customer to at most 2 supply facilities is optimal. The gap between the performance of the single sourcing and multiple sourcing solutions is significant. However, the difference when we increase N_i from 2 to 3 is not significant.

Next, we analyze the effect of CoV. Table 4-3 summarizes the results. As we can see in both Table 4-3 and Figure 4-5, as the coefficient of variation increases from 0.35 to 0.45, the percentage cost difference between optimal single sourcing and demand splitting solutions decreases. The main reason for this is that as the CoV increases, the standard deviation of demand increases. In turn, this leads to higher safety stock holding costs. The model tends to open fewer supply facilities and benefits from risk pooling by assigning more customers to fewer supply facilities. Similarly, we expect a decrease in the percentage cost difference as the CoV approaches the origin because, in this case, the safety stock holding cost becomes so small that the problem becomes similar to an uncapacitated facility location problem.

We next analyze the effect of the fixed facility opening cost. This effect is shown in Table 4-4. As we would expect, as the fixed cost increases, fewer locations are opened, and customers are therefore assigned to fewer locations. Thus, the benefits of demand splitting decrease as the fixed facility costs increase.

Finally, we consider the CPU times for different values of N_i. Figure 4-7 illustrates these results. As Figure 4-7 shows, the greatest CPU time is needed when $N_i = 2$. In most of the instances when there is no limit on the number of facilities that can supply a customer, the optimal solution assigns a customer to at most 3 supply facilities. When we limit the number of supply facilities to $N_i = 2$, the corresponding constraint becomes tight and the required CPU time increases. This increase in CPU time comes as a result of the increased time CPLEX must spend solving the 0-1 integer master problem (RMP). However, when $N_i = 5$, the constraint is loose in almost all instances, and the required CPU time is significantly lower.

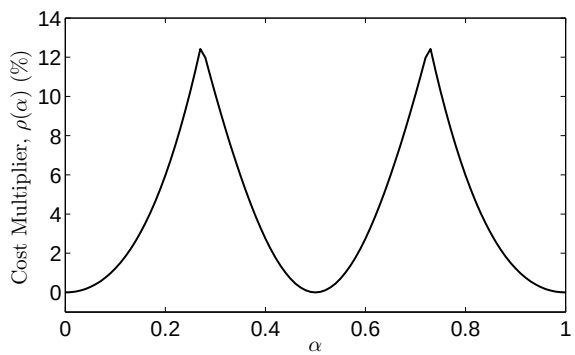

Figure 4-1. Cost increase multiplier for single-sourcing as a function of α

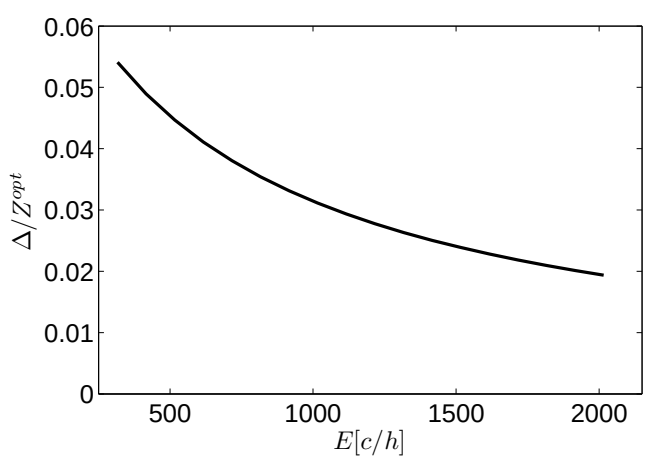

Figure 4-2. Ratio of cost savings from splitting to minimum cost as a function of $E[c/h]$

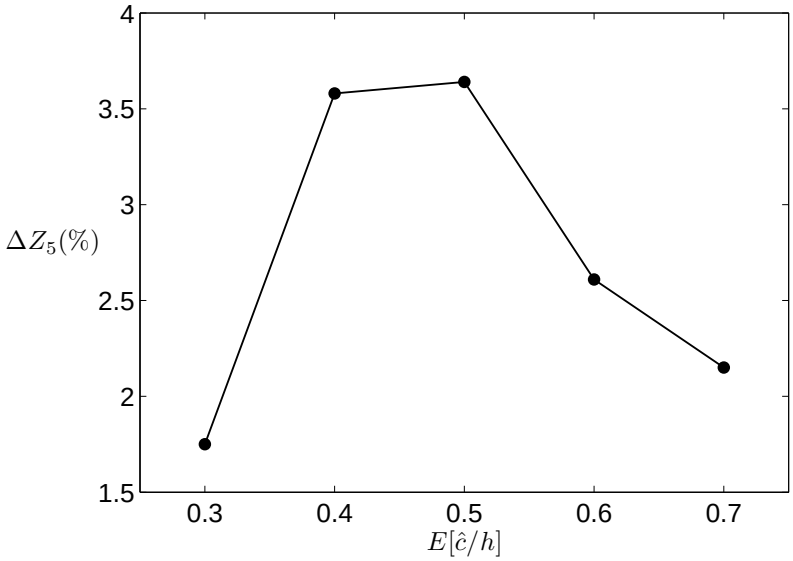

Figure 4-3. The effect of $E[\hat{c}/h]$ on ΔZ_5

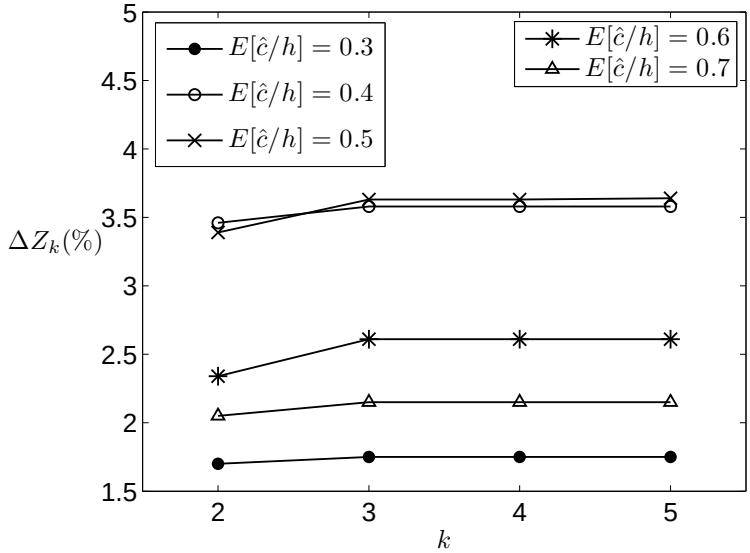

Figure 4-4. The effect of N_i on ΔZ_k for different values of $E[\hat{c}/h]$

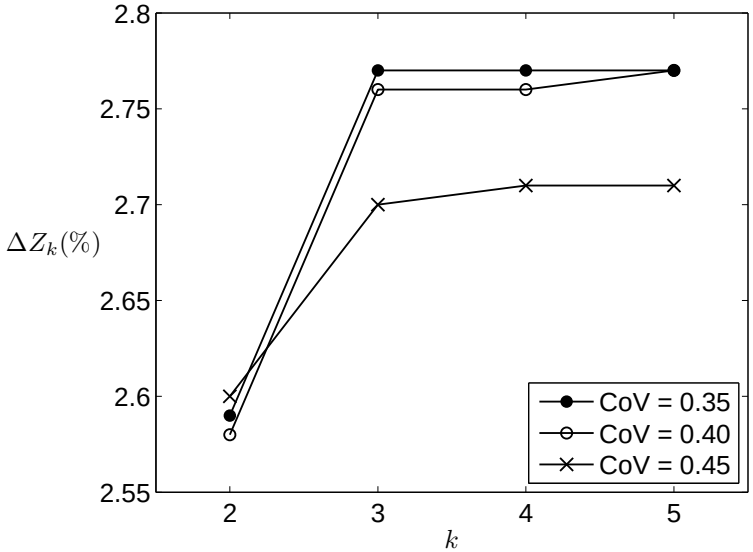

Figure 4-5. The effect of N_i on ΔZ_k for different values of CoV

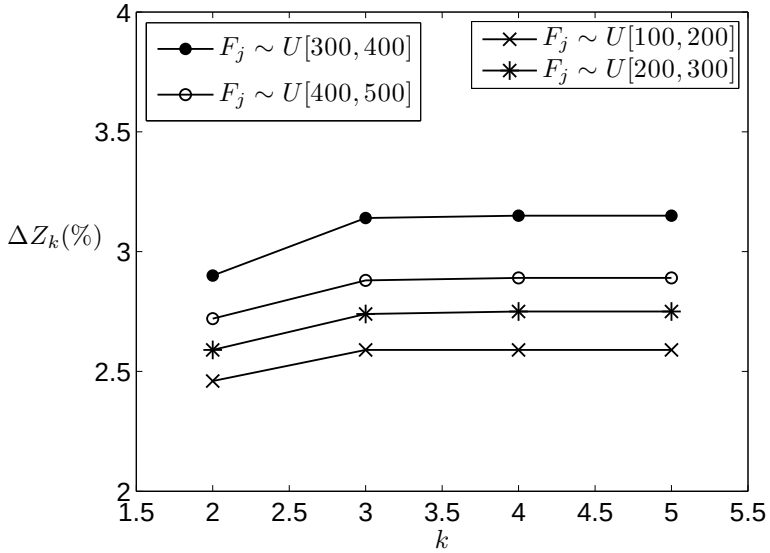

Figure 4-6. The effect of N_i on ΔZ_k for different values of fixed cost

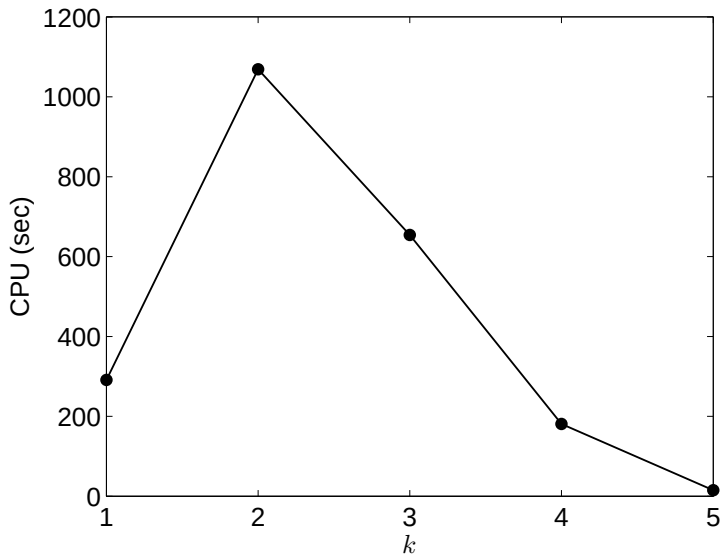

Figure 4-7. Cpu times for different values of N_i

Table 4-1. Data parameter settings

$\mathbf{E}[\hat{c}/h]$	c_{ij}	μ_i	**CoV** (σ/μ)	**Fixed Cost** (F_j)
0.3	$U[0.05, 0.55]$	$U[4000, 6000]$	0.35	$U[100, 200]$
0.4	$U[0.05, 0.75]$		0.40	$U[200, 300]$
0.5	$U[0.05, 0.95]$		0.45	$U[300, 400]$
0.6	$U[0.05, 1.15]$			$U[400, 500]$
0.7	$U[0.05, 1.35]$			

Table 4-2. The max, min, and average values of ΔZ_5 for different values of $E[\hat{c}/h]$

	ΔZ_5		
$E(\hat{c}/h)$	**max**	**min**	**average**
0.3	4.40%	0.00%	1.75%
0.4	**6.82%**	0.09%	3.58%
0.5	5.30%	**2.39%**	3.64%
0.6	3.92%	0.62%	2.61%
0.7	5.25%	0.49%	2.15%

Table 4-3. The max, min, and average values of ΔZ_5 for different values of CoV

	ΔZ_5		
CoV	**max**	**min**	**average**
0.35	6.82%	0.00%	2.87%
0.4	6.75%	0.00%	2.85%
0.45	5.77%	0.00%	2.76%

Table 4-4. The max, min, and average values of ΔZ_5 for different values of fixed cost

	ΔZ_5		
fixed cost	**max**	**min**	**average**
U(100,200)	6.82%	0.00%	3.15%
U(200,300)	6.49%	0.00%	2.89%
U(300,400)	6.23%	0.00%	2.59%
U(400,500)	5.77%	0.00%	2.36%

CHAPTER 5
COMPONENT PROCUREMENT PLANNING AND PRODUCT PORTFOLIO DESIGN PROBLEM

5.1 Introduction and Motivation

Determining an appropriate degree of product variety is a critical decision that significantly affects a firm's cost and revenue structures. On the one hand, offering a small number of products is desirable from a cost point of view, since each product has associated design, production and inventory holding costs. On the other hand, offering a large number of products is desirable from a demand satisfaction point of view since each customer may find an "ideal product" on which she places a high premium. If a customer's ideal product is not offered by the company, then the customer may be willing to accept a substitute if offered an appropriate incentive, or may leave without buying anything. In the personal computer industry, for example, if a customer's ideal product contains a certain component configuration, then a different component configuration may act as a substitute or the customer may find her preferred configuration offered by another supplier. Thus, a critical problem from a supplier's perspective is determining which products to offer and in what quantities, in order to maximize profit.

Depending on the customer's profile and her ideal product, an alternative product set can often be identified (which includes all substitute products that this customer may be willing to buy instead of her ideal product). For example, if a customer's ideal product is a computer with a 2GB RAM, she may be willing to buy a computer with the same configuration with 3GB RAM instead, if an incentive is offered. Then, this new product will serve as one of the substitutes for her ideal product and will be a member of the alternative product set for this customer. Another example can be found in problem settings where one-way substitution is possible. In these contexts, a product can be substituted for higher/lower level quality or performance products. In such cases, the alternative set for an ideal product will include all products whose quality/performance levels are higher/lower than that of the ideal product. A customer may leave the system

at any point without buying a product (which results in a lost sale), or may stay in the system and consider the next available substitute product. From this perspective, deciding which products to offer is a critical decision.

Several issues complicate the optimization of the product portfolio in this environment. One of these issues is the product architecture. If a product is assembled from a number of components that may be ordered from an outside supplier or manufactured in house, then the problem contains an embedded component procurement/production planning problem. In this type of environment, several different products can often draw from the same set of components. The possibility of substitution across products impacts component inventory levels. Therefore, while optimizing the assortment of products, we also need to consider the component procurement/manufacturing plan that minimizes the component order/manufacturing and inventory holding costs.

In this chapter we consider a multi-period component procurement planning and product portfolio design problem with product substitutions. We assume that there is a set of potential or "ideal" products for which demands occur from a set of customer segments. Each customer segment-ideal product pair has an alternative product set that includes all potentially acceptable substitutes for the ideal product for that customer segment. If the ideal product of a customer is not available, the substitute products are offered in a predetermined or ranked order at some discount. For example, in the previous example, where one-way substitution is possible, higher/lower quality/performance products may be ranked and offered to the customer in ascending/descending quality/ performance order. The customer has three choices at any point: (i) accept a substitute product, (ii) decline the current offered substitute product and consider the next available one, or (iii) leave the system without buying any product. Products are assembled from a set of components, which are procured from an outside supplier at a fixed plus variable cost in each procurement period. We model this problem as a large scale mixed-integer linear programming problem. We first show that this problem is $\mathcal{NP}$–Hard, even for

the case where there is an unlimited supply of components at no cost. We then consider solution methods for the problem and develop a Benders decomposition-based exact algorithm.

The remainder of this chapter is organized as follows. We give a brief overwiev of the literature related to our work in Section 5.2. In Section 5.3, we define the problem and formulate the mathematical model, and in Section 5.4 we provide the Benders decomposition-based algorithm that we developed to solve the model.

5.2 Literature Review

Both assemble-to-order systems and demand substitution models (and also their joint applications) are related to our work. Therefore, we analyze the literature on both of these areas and explain our contribution to the existing literature in this section.

The most closely related branch of the assemble-to-order literature studies systems where products require common components and analyzes the effects of commonality on component inventory levels (see Baker et al. (1986), Gerchak et al. (1988), Gerchak and Henig (1989), Hillier (2000), Lu et al. (2003)). Akcay and Xu (2004) study an assemble-to-order system with multiple components and multiple products, where each product has a prespecified response time window. The system receives a reward if the demand is fulfilled within its response time window. They formulate this problem as a two-stage stochastic integer program to determine the optimal base stock levels of components subject to an investment budget. Afentakis et al. (1984) develop a branch and bound algorithm for optimal lot sizing in multistage assembly systems. Their method is suitable for products with an assembly structure only. Afentakis and Gavish (1986) relax this restriction and examine the lot sizing problem for general product structures by transforming the general product structure problem into an equivalent and larger assembly system. Rosling (1989) identifies the optimal policy for uncapacitated multistage general assembly systems under a restriction on the initial stock levels. With this condition, the assembly system can be interpreted as a series system, and hence, can be solved optimally.

In the substitution literature, one-way substitution, in which a demand for a lower (higher) quality item can be satisfied by using a higher (lower) quality item, is a well studied area (see Pentico (1976), Chand et al. (1994), Bassok et al. (1999), Rao et al. (2004), Hsu et al. (2005), Taşkın and Ünal (2009)). Bassok et al. (1999) study single period multi-product inventory model with stochastic demand and full downward substitution where the unsatisfied demand for a product can be filled with a product with higher utility. Smith and Agrawal (2000) develop a model that determines the effect of substitution on the demand distribution, inventory levels of items, and customer service levels. They assume that demand originates from a random number of customers, who select randomly with known frequencies from a "choice set" of items that contains all potential substitutes. Pentico (1974) studies an assortment problem in which a set of candidate sizes of some product is given, from which a subset of sizes will be selected to be stocked. Demand for an unstocked size is filled from a larger stocked size with an associated substitution cost. He provides an optimal stationary stocking policy under certain assumptions, and extends the problem by considering a nonlinear cost function in Pentico (1976).

Balakrishnan and Geunes (2000) study a dynamic, multi-period requirements-planning problem with flexible bills-of-materials with an option to substitute components. They model the problem as an integer program and provide a dynamic-programming solution algorithm that generalizes the single-item lot-sizing algorithm. Hale et al. (2000) study a model with two products, each composed of two components, one of which can be downward substituted. They formulate this problem as a two-stage stochastic program, the objective of which they proved to be jointly concave in the order quantities, allowing them to develop bounds on the optimal order quantities. Yunes et al. (2007) develop marketing and operational methodologies and tools for John Deere, one of the world's leading producers of machinery, reducing costs by concentrating product line configurations while maintaining high customer service and profits. Deere's products

contain various features, each of which can be selected from a number of possible options, which may result in tens of thousands of combinations of options. For every customer, a migration list is created that consists of a set of acceptable configurations and is sorted in decreasing order of preference. They build an integer programming model by using the migration lists, costs and profits for all feasible configurations, where the configurations are found via constraint programming.

Our work is most closely related to that of Ervolina et al. (2009). They propose a process that aims at finding marketable product alternatives that are assembled-to-order from a certain number of components, each having a limited supply. They provide a single-period mathematical model that determines a substitution plan for a system, where the demand is deterministic and a known percentage of customers accept a substitute product if its price and quality are within a certain range. In their model, they define a core product set that includes the products for which demands may occur, and a set of alternative products that includes the products that may be used to satisfy the demand occurred for core products. They simulate the system and provide computational results. No solution algorithm is provided. In contrast, we propose a multi-period production and substitution plan in which we decide which products to offer, how to satisfy demands and how to procure the components that are used to produce products. Our model have distinct features then that of Ervolina et al. (2009): (i) our problem is a multi-period problem; (ii) it includes a component procurement plan that accounts for economies of scale, which makes the problem much complex and realistic; (iii) we assume that demand may occur for any product that is decided to be designed (not only for a certain set of products); (iv) we define an alternative product set for every customer segment-ideal product pair (not just one set for all customer segments and core products); and (v) we propose an exact algorithm to solve the model.

Our contribution to the existing literature is two fold. First, we model a multi-period complex product-line design problem with product substitutions, in which products are

assembled from a set of components ordered from an outside supplier. Second, we prove that this problem is $\mathcal{NP}$–Hard and propose an exact algorithm to solve the proposed model. We provide some computational results that compare our algorithm with a commercial solver and show the limits of our algorithm.

5.3 Problem Description and Mathematical Formulation

We consider a set of products, M, each of which has a fixed design cost, f_m. A set of customer segments, C, exists, each with demands for "ideal" products, and a set of alternative products, N_m^c, that consists of the substitute products that she might be willing to buy if her ideal product is not made available. Our alternative product set is similar to the "choice set" defined in Smith and Agrawal (2000), where a random number of customers chooses a product from the choice set that contains all potential substitutes. In contrast, we assume that demand occurs for a specific product, and that product has an alternative product set, which corresponds to a "choice set" that is defined for every customer segment. We assume that the alternative product set is known both to the retailer and the customer. If a customer's ideal product is not made available, demand may be satisfied using an alternative product from this set at a substitution cost, if the customer accepts a substitute.

Yunes et al. (2007) develop an algorithm with a customer migration component, which quantitatively characterizes customer behavior by permitting a customer to migrate to an alternative configuration if her first choice is unavailable. For every customer, they create a migration list that consists of a set of acceptable configurations and is sorted in decreasing order of preference. In this study, we use alternative product sets that are defined for every product-customer segment pair, like the migration lists developed by Yunes et al. (2007). We also assume that these alternative product sets are ranked in order of customers preferences. Therefore, if a customer's ideal product is not made available, she can either leave the system or purchase the next substitute item from the alternative product list. When the customer leaves the system without a purchase, the

demand is lost, and we assume this occurs for $\alpha^c_{mn}\%$ of the customers in segment c whose ideal product is m when they are offered substitute product n.

Each product m has a profit margin, p_{mt}, that is period dependent. Moreover, each product is assembled-to-order from a set of components, I, and inventory is held at the component level. Component i procurement costs contain a nonnegative fixed cost, a_{it}, plus variable cost, b_{it}, in period t. Each product has an associated *usage* vector, $\mathbf{u}$, which determines the number of components included in the product (i.e., u_{im} is the number of required components of type i in product m). If a product does not contain a component, the corresponding row of the usage vector is zero.

The parameters and decision variables for the problem are as follows:

Parameters

C	set of customer segments, indexed by c
I	set of components, indexed by i
M	set of products, indexed by m
N^c_m	set of alternative products to ideal product m for customer segment c (items in N^c_m are ranked in a preference order such that $n_0 \succ n_1 \succ \ldots$, where n_0 is the ideal product m, and $\succ$ is used as a precedence order symbol)
T	set of time periods, indexed by t
u_{im}	number of units of component $i \in I$ used in assembly of product $m \in M$
D^c_{mt}	demand for ideal product m and customer segment c in period t
h_{it}	inventory holding cost of component i in period t
p_{mt}	per-unit marginal profit of product $m \in M$ in period t
f_m	fixed design or offering cost of product m
w^c_{mnt}	per-unit penalty cost for substituting one unit of product n for one unit of product m for customer segment c in period t (Note that $w^c_{mn_0t} = 0$ for all $c \in C, m \in M$ and $t \in T$)
α^c_{mn}	Percentage of customers in segment c that will not accept product n as a

substitution for product m and will leave the system after product n is offered as a substitute (Note that $\alpha^c_{mn_0}$ gives the percentage of customers in segment c that will leave the system if ideal product m is not available)

H_{it} inventory of component i at the beginning of period t

a_{it} fixed procurement cost of component i in period t

b_{it} variable procurement cost of component i in period t

Decision Variables

Y^c_{mnt} number of units of product n used to satisfy demand for ideal product m for customer segment c in period t

$K^c_{mn_i}$ number of customers from segment c remaining in the system after alternative product n_i is offered (Note that $K^c_{mn_i}$ is actually not an explicit decision variable and its value can be derived from Y^c_{mnt}.)

V_{it} procurement quantity of component i in period t

Z_m 1, if product m is offered; 0, otherwise

δ_{it} 1, if we procure component i in period t; 0, otherwise

The problem, in a multi-period setting, requires determining which products to offer, how many components to procure, and how many components to hold in inventory in each period in order to maximize overall profit. Hence, our aim is to determine a product portfolio, substitution plan, and procurement plan in order to maximize profit. We formulate this problem as a mixed-integer linear programming problem as follows:

(MILP-1)

$$\textbf{Maximize} \quad \sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{t\in T}(p_{nt}-w^c_{mnt})Y^c_{mnt} - \sum_{m\in M} f_m Z_m - \sum_{i\in I}\sum_{t\in T}(a_{it}\delta_{it}+b_{it}V_{it}+h_{it}H_{it}) \tag{5–1}$$

$$\textbf{Subject to} \quad H_{it} = H_{i,t-1} + V_{it} - \sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}(u_{in}Y^c_{mnt}), \quad \forall i\in I,\ t\in T \tag{5–2}$$

$$H_{i0} = 0, \quad \forall i \in I \tag{5–3}$$

$$K^c_{mn_0t} = \left(D^c_{mt} - Y^c_{mn_0t}\right)(1-\alpha^c_{mn_0}), \quad \forall c \in C, m \in M, t \in T \tag{5–4}$$

$$K^c_{mn_kt} = \left(K^c_{mn_{k-1}t} - Y^c_{mn_kt}\right)(1-\alpha^c_{mn_k}), \quad \forall c \in C, m \in M, n_{k:k>1} \in N^c_m,\ t \in T \tag{5–5}$$

$$Y^c_{mnt} - D^c_{mt}Z_n \le 0, \quad \forall c \in C,\ m \in M, n \in N^c_m,\ t \in T \tag{5–6}$$

$$V_{it} - \left(\sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{l=t}^{T} u_{in}D^c_{ml}\right)\delta_{it} \le 0, \quad \forall i \in I,\ t \in T \tag{5–7}$$

$$Y^c_{mnt}, V_{it}, H_{it} \ge 0, \quad \forall c \in C,\ m \in M, n \in N^c_m,\ t \in T,\ i \in I \tag{5–8}$$

$$Z_m, \delta_{it} \in \{0,1\}, \quad \forall m \in M,\ i \in I,\ t \in T. \tag{5–9}$$

The objective function (5–1) maximizes the sum of profit margins less the substitution cost, fixed product design cost, fixed and variable procurement cost and the inventory holding cost of components. Constraints (5–2) and (5–3) are inventory balance constraints. Constraints (5–4) and (5–5) track the number of customers remaining in the system after each substitute offer. Constraints (5–6) ensure that we produce and sell only the products that are offered. Constraints (5–7) ensure that we procure components only if we incur the associated procurement cost. The other constraints include nonnegativity and binary requirements.

Let $c_{it} = b_{it} + \sum_{l=t}^{T} h_{il}$ and $\mathcal{L}_{it} = \sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{l=t}^{T} u_{in}D^c_{ml}$. Since $H_{it} = \sum_{l=1}^{t} V_{il} - \sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{l=1}^{t} u_{in}Y^c_{mnl}$ for all $i \in I$ and $t \in T$, then we can rewrite **(MILP-1)** as follows:

(MILP-2)

$$\textbf{Maximize} \quad \sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{t\in T}(p_{nt} - w^c_{mnt})Y^c_{mnt} - \sum_{m\in M} f_m Z_m$$
$$- \sum_{i\in I}\sum_{t\in T}(a_{it}\delta_{it} + c_{it}V_{it}) + \sum_{i\in I}\sum_{t\in T}\sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{l=1}^{t} h_{it}u_{in}Y^c_{mnl} \tag{5–10}$$

$$\textbf{Subject to} \quad \sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{l=1}^{t} u_{in}Y^c_{mnl} - \sum_{l=1}^{t} V_{il} \le 0, \quad \forall i \in I,\ t \in T \tag{5–11}$$

$$K^c_{mn_0t} = (D^c_{mt} - Y^c_{mn_0t})(1 - \alpha^c_{mn_0}), \quad \forall c \in C, m \in M, t \in T \tag{5–12}$$

$$K^c_{mn_kt} = \left(K^c_{mn_{k-1}t} - Y^c_{mn_kt}\right)(1 - \alpha^c_{mn_k}), \quad \forall c \in C, m \in M, n_{k:k>1} \in N^c_m,\ t \in T \tag{5–13}$$

$$Y^c_{mnt} - D^c_{mt} Z_n \leq 0, \quad \forall c \in C,\ m \in M, n \in N^c_m,\ t \in T \tag{5–14}$$

$$V_{it} - \mathcal{L}_{it}\delta_{it} \leq 0, \quad \forall i \in I,\ t \in T \tag{5–15}$$

$$Y^c_{mnt}, K^c_{mnt}, V_{it} \geq 0, \quad \forall c \in C,\ m \in M, n \in N^c_m,\ t \in T,\ i \in I \tag{5–16}$$

$$Z_m, \delta_{it} \in \{0, 1\}, \quad \forall m \in M,\ i \in I,\ t \in T. \tag{5–17}$$

Observe that when the products that will be offered and the number of products that will be assembled in each period are known, the resulting problem is actually a component procurement planning problem, which is a well-known uncapacitated lot sizing problem. We will use this fact later to develop a decomposition-based solution algorithm. First, however, we show that (MILP-2) is $\mathcal{NP}$-Hard.

Theorem 6. *(MILP-2) is $\mathcal{NP}$-Hard.*

Proof. To prove that (MILP-2) is $\mathcal{NP}$-Hard we will show that the uncapacitated facility location problem (UFLP) can be reduced to (MILP-2). Specifically, we will show that (MILP-2) contains UFLP as a special case. The UFLP can be formulated as follows:

$$\begin{aligned}
\textbf{Minimize} \quad & \sum_{i \in M} f_i X_i + \sum_{i \in M} \sum_{j \in N} c_{ij} Y_{ij} \\
\textbf{Subject to} \quad & \sum_{i \in M} Y_{ij} = D_j, \quad \forall j \in N \\
& Y_{ij} \leq D_j X_i, \quad \forall i \in M, j \in N \\
& Y_{ij} \geq 0, X_i \in \{0, 1\}, \quad \forall i \in M, j \in N.
\end{aligned}$$

consider the following special case of (MILP-2): Assume that we have a single period problem ($|T| = 1$) with a single customer segment ($|C| = 1$) that has demand for M

products, each of which has an alternative product set N_m. Set $a_{it} = b_{it} = h_{it} = 0$, for all $i \in I$, $t \in T$, which means there is an ample supply of components with no procurement and holding costs. Then, we can neglect constraint (5–11), without loss of optimality. Additionally, assume $\alpha = 0$, which requires that customers stay in the system until the last substitute product is offered if their ideal product is not available. Then, we can write this special case of (MILP-2) as follows:

$$\textbf{Maximize} \quad \sum_{m \in M} \sum_{n \in N_m} (p_n - w_{mn}) Y_n - \sum_{m \in M} f_m Z_m \tag{5–18}$$

$$\textbf{Subject to} \quad \sum_{n \in N_m} Y_{mn} \leq D_m, \quad \forall m \in M \tag{5–19}$$

$$Y_{mn} \leq D_m Z_n, \quad \forall m \in M, n \in N_m \tag{5–20}$$

$$Y_{mn} \geq 0, Z_m \in \{0, 1\}, \quad \forall m \in M, n \in N_m. \tag{5–21}$$

For evey $m \in M$, let $k_m = \max_{n \in N_m}(p_n - w_{mn})$ and $c_{mn} = k_m - (p_n - w_{mn})$, i.e., $(p_n - w_{mn}) = k_m - c_{mn}$. We can write the above objective as

$$\textbf{Maximize} \quad \sum_{m \in M} \sum_{n \in N_m} (k_m - c_{mn}) Y_{mn} - \sum_{m \in M} f_m Z_m = \sum_{m \in M} k_m \sum_{n \in N_m} Y_{mn} - \sum_{m \in M} \sum_{n \in N_m} c_{mn} Y_{mn} - \sum_{m \in M} f_m Z_m.$$

Consider a special case with sufficiently large values of $(p_n - w_{mn})$ to ensure that, at optimality, $\sum_{n \in N_m} Y_{mn} = D_m$, for every $m \in M$. Then, we can rewrite our model equivalently as:

$$\textbf{Minimize} \quad \sum_{m \in M} f_m Z_m + \sum_{m \in M} \sum_{n \in N_m} c_{mn} Y_{mn} - \sum_{m \in M} \sum_{n \in N_m} k_m D_m \tag{5–22}$$

$$\textbf{Subject to} \quad \sum_{n \in N_m} Y_{mn} = D_m, \quad \forall m \in M \tag{5–23}$$

$$Y_{mn} \leq D_m Z_n, \quad \forall m \in M, n \in N_m \tag{5–24}$$

$$Y_{mn} \geq 0, Z_m \in \{0, 1\}, \quad \forall m \in M, n \in N_m. \tag{5–25}$$

The above special instance of (MILP-2) is nothing but a UFLP. Since the UFLP is known to be an $\mathcal{NP}$-Hard problem, this proves that (MILP-2) is also $\mathcal{NP}$-Hard. $\square$

(MILP-2) quickly grows in the number of variables as the number of potential products, customer segments, and periods grow. For example, for a problem with 10 customer segments, 30 products, 20 components, and 12 periods, if any product can substitute for any other product, the number of binary variables is 270 ($= 30 + 20 \times 12$), the number of continuous variables is $54,240$ ($= 10 \times 30 \times 15 \times 12 + 20 \times 12$) and the number of constraints is $108,480$ ($= 20 \times 12 + 10 \times 30 \times 15 \times 12 + 10 \times 30 \times 15 \times 12 + 20 \times 12$). Our preliminary tests show that CPLEX runs out of memory even for a medium-size problem with 10 customer segments, 30 products, 20 components and 12 periods. Therefore, we seek an algorithm that can handle medium and large problem sizes.

The next section provides such a solution algorithm, which is a Benders decomposition-based exact algorithm that we have developed for solving (MILP-2).

5.4 Solution Methodology

We note that **(MILP-2)** reduces to an uncapacitated lot sizing model for a fixed product portfolio, $\tilde{\mathbf{Z}}$, and a feasible assembly plan, $\tilde{\mathbf{Y}}$. We can use this observation to design a decomposition algorithm. The underlying lot sizing problem for a fixed assembly plan, $\tilde{\mathbf{Y}}$, can be written as follows:

$$\textbf{(SP-IP1)} \quad \textbf{Minimize} \quad \sum_{i \in I} \sum_{t \in T} (a_{it}\delta_{it} + c_{it}V_{it}) \tag{5–26}$$

$$\textbf{Subject to} \quad \sum_{l=1}^{t} V_{il} - \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} \sum_{l=1}^{t} u_{in}\tilde{Y}_{mnl}^{c} \geq 0, \quad \forall i \in I,\ t \in T \tag{5–27}$$

$$\mathcal{L}_{it}\delta_{it} - V_{it} \geq 0, \quad \forall i \in I,\ t \in T \tag{5–28}$$

$$V_{it} \geq 0, \quad \forall i \in I,\ t \in T \tag{5–29}$$

$$\delta_{it} \in \{0, 1\}, \quad \forall i \in I,\ t \in T. \tag{5–30}$$

We can design a Benders decomposition algorithm that uses **(SP-IP1)** as a subproblem. In order to use a Benders decomposition algorithm, we need two types of information: (1)

the dual variables associated with the subproblem's constraints, and (2) a lower bound on the value of the subproblem. Since **(SP-IP1)** is an integer programming problem, it is not directly amenable to dualization techniques in order to guarantee finding an optimal solution. Therefore, we cannot directly use the above lot sizing model in a decomposition algorithm to obtain an exact solution. However, since the subproblem is a minimization problem, the dual of its relaxation's objective function value will always give us a lower bound. Hence, we can always use its linear relaxation to get a lower bound and obtain the dual variables. Note that **(SP-IP1)** is separable among components. Then, the linear relaxation of model **(SP-IP1)** for each component $i \in I$ can be given as

$$\textbf{(SP-LP}(i)\textbf{)} \quad \textbf{Minimize} \quad \sum_{t \in T} (a_{it}\delta_{it} + c_{it}V_{it}) \tag{5–31}$$

$$\textbf{Subject to} \quad \sum_{l=1}^{t} V_{il} - \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} \sum_{l=1}^{t} u_{in}\tilde{Y}_{mnl}^{c} \geq 0, \quad \forall\, t \in T \tag{5–32}$$

$$\mathcal{L}_{it}\delta_{it} - V_{it} \geq 0, \quad \forall\, t \in T \tag{5–33}$$

$$V_{it} \geq 0, \quad \forall t \in T \tag{5–34}$$

$$0 \leq \delta_{it} \leq 1, \quad \forall\, t \in T. \tag{5–35}$$

Note that since this is a minimization problem, the restriction $\delta_{it} \leq 1$ in the LP-relaxation given above will always hold and therefore can be omitted from the formulation. Let ξ_{it} and ζ_{it} be the dual variables associated with constraints (5–32) and (5–33), respectively. Then the dual of model **(SP-LP(i))** for component $i \in I$ can be written as

$$\textbf{(D-SP-LP}(i)\textbf{)} \quad \textbf{Maximize} \quad \sum_{t \in T} \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} \sum_{l=1}^{t} u_{in}\tilde{Y}_{mnl}^{c}\xi_{it}$$

$$\textbf{Subject to} \quad \mathcal{L}_{it}\zeta_{it} \leq a_{it}, \quad \forall t \in T,$$

$$\xi_{it} - \zeta_{it} \leq c_{it}, \quad \forall t \in T,$$

$$\zeta_{it}, \xi_{it} \geq 0, \quad \forall t \in T.$$

Let F be the feasible space of the above model **(D-SP-LP(i))**. Since $\xi_{it} = c_{it}$ and $\zeta_{it} = 0$ for all $i \in I$ and $t \in T$ is a feasible solution for this model, F is not empty. F is, therefore, composed of the set of extreme points (ξ^p, ζ^p) (for $p = 1, \ldots, P$, where P is the number of extreme points of F). Moreover, since **(SP-LP(i))** is an uncapacitated lot sizing problem, the primal problem has always a feasible solution which directly requires the dual to be bounded.

The solution to **(D-SP-LP(i))** is one of the extreme points (ξ^p, ζ^p). Therefore, the maximum value of the subproblem is its value at one of the extreme points of F. We add the constraint

$$\theta \geq \sum_{i \in I} \sum_{t \in T} \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} \sum_{l=1}^{t} u_{in} \tilde{Y}_{mnl}^c \xi_{it} \tag{5–36}$$

for all extreme points of the dual feasible region as Benders cuts to create a Benders master problem.

Since the feasible space of the subproblem is independent of the choice made for the Y variables, we can use the dual variables obtained by solving the above formulation in our Benders cut. Then we can write the master problem as follows:

(MP)

$$\textbf{Maximize} \quad \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} \sum_{t \in T} (p_{nt} - w_{mnt}^c) Y_{mnt}^c - \sum_{m \in M} f_m Z_m$$

$$+ \sum_{i \in I} \sum_{t \in T} \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} \sum_{l=1}^{t} h_{it} u_{in} Y_{mnl}^c - \theta$$

$$\textbf{Subject to} \quad \theta \geq \sum_{i \in I} \sum_{t \in T} \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} \sum_{l=1}^{t} u_{in} \tilde{Y}_{mnl}^c \xi_{it}, \quad \forall p = 1, \ldots, P \tag{5–37}$$

$$K_{mn_0t}^c = (D_{mt}^c - Y_{mn_0t}^c)(1 - \alpha_{mn_0}^c), \quad \forall c \in C, m \in M, t \in T \tag{5–38}$$

$$K_{mn_kt}^c = \left(K_{mn_{k-1}t}^c - Y_{mn_kt}^c\right)(1 - \alpha_{mn_k}^c), \quad \forall c \in C, m \in M, n_{k:k>1} \in N_m^c, \ t \in T \tag{5–39}$$

$$Y_{mnt}^c - D_{mt}^c Z_n \leq 0, \quad \forall c \in C, \ m \in M, \ n \in N_m^c, \ t \in T \tag{5–40}$$

$$Y^c_{mnt}, K^c_{mnt}, \theta \geq 0, \quad \forall c \in C,\ m \in M,\ n \in N^c_m,\ t \in T \tag{5–41}$$

$$Z_m \in \{0,1\}, \quad \forall m \in M, \tag{5–42}$$

where P is the number of extreme points associated with the subproblems **(SP-LP(i))**. There is an exponential number of such extreme points, each of which corresponds to a constraint of the form of (5–36). However, we can generate valid Benders cuts and successively add these to the formulation. Then, the relaxed master problem can be written as

(RMP)

$$\textbf{Maximize} \quad RMP^z = \sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{t\in T}(p_{nt} - w^c_{mnt})Y^c_{mnt} - \sum_{m\in M} f_m Z_m \tag{5–43}$$

$$+ \sum_{i\in I}\sum_{t\in T}\sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{l=1}^{t} h_{it}u_{in}Y_{mnl} - \theta$$

$$\textbf{Subject to} \quad \theta \geq \sum_{i\in I}\sum_{t\in T}\sum_{c\in C}\sum_{m\in M}\sum_{n\in N^c_m}\sum_{l=1}^{t} u_{in}\tilde{Y}^c_{mnl}\xi_{it}, \quad \forall p = 1,\ldots,K \tag{5–44}$$

$$K^c_{mn_0t} = (D^c_{mt} - Y^c_{mn_0t})(1 - \alpha^c_{mn_0}), \quad \forall c \in C, m \in M, t \in T \tag{5–45}$$

$$K^c_{mn_kt} = \left(K^c_{mn_{k-1}t} - Y^c_{mn_kt}\right)(1 - \alpha^c_{mn_k}), \quad \forall c \in C, m \in M, n_{k:k>1} \in N^c_m,\ t \in T \tag{5–46}$$

$$Y^c_{mnt} - D^c_{mt}Z_n \leq 0, \quad \forall c \in C, m \in M, n \in N^c_m, t \in T \tag{5–47}$$

$$Y^c_{mnt}, K^c_{mnt}, \theta \geq 0, \quad \forall c \in C, m \in M, n \in N^c_m, t \in T \tag{5–48}$$

$$Z_m \in \{0,1\}, \quad \forall m \in M, \tag{5–49}$$

where K denotes the number of Benders cuts we have generated.

Note that the (RMP) formulation is a mixed integer linear program. At each iteration, we solve (RMP) to obtain a (possibly) new $\mathbf{Y^k}$ vector. Given this $\mathbf{Y^k}$ vector, we then solve the subproblem **(SP-LP(i))** for all $i \in I$ to determine the corresponding optimal dual values, ξ. The objective function value of (RMP) gives us an upper bound at each iteration. The objective function value of the subproblems will combine to form a

lower bound on the variable θ. Therefore, we can replace the value of θ with the sum of the objective function values of the subproblems in (RMP)'s objective function to obtain a lower bound. If the ratio of the difference of the upper and lower bounds to the lower bound is less than an ϵ, which is a small predetermined number, then we stop. Otherwise we add a new constraint in the form of inequality (5–36) to (RMP) formulation, re-solve (RMP) and repeat this procedure.

The objective function value of a relaxation of the actual subproblem may lead to a weak bound for our algorithm. If we can rewrite the lot sizing model so that the LP-relaxation gives a solution in which the δ-variables take integral values, we can use the new formulation's optimal objective function value to calculate the lower bound.

Define V_{its} as the number of units of component i procured in period t to satisfy assembly requirements in period $s \geq t \in T$. By disaggregating the V_{it} variables into V_{its} variables, we can obtain a model in the form of the simple plant location formulation given in Krarup and Bilde (1977), which is known to have an optimal solution in which the binary variables are integer in the LP-relaxation solution.

Note that **(SP-IP1)** is separable among components. Therefore, we can solve **(SP-IP1)** for every component separately and then merge the results. Since we assume the Y variables are known in **(SP-IP1)**, we can use these known values to make constraint (5–28) as tight as possible. Let $\Delta_{is} = \sum_{c \in C} \sum_{m \in M} \sum_{n \in N_m^c} u_{in} \tilde{Y}_{mns}^c$. Then the subproblem formulation for component $i \in I$ can be written as follows:

$$\textbf{(SP-DIP(i))} \qquad \textbf{Minimize} \quad \sum_{t \in T} \left(a_{it}\delta_{it} + c_{it} \sum_{s=t}^{T} V_{its} \right) \tag{5–50}$$

$$\textbf{Subject to} \quad \sum_{t=1}^{s} V_{its} \geq \Delta_{is}, \quad \forall s \in T \tag{5–51}$$

$$\Delta_{is}\delta_{it} - V_{its} \geq 0, \quad \forall t, s \in T, \ s \geq t \tag{5–52}$$

$$V_{its} \geq 0, \quad \forall t, s \in T, \ s \geq t \tag{5–53}$$

$$\delta_{it} \in \{0, 1\}, \quad \forall t \in T. \tag{5–54}$$

Note that **(SP-DIP(i))** has a simple plant location formulation whose LP-relaxation solution is known to have a binary optimal solution. The LP-relaxation of model **(SP-DIP(i))** can be written as follows:

$$\textbf{(SP-DLP(i))} \qquad \textbf{Minimize} \quad \sum_{t\in T}\left(a_{it}\delta_{it} + c_{it}\sum_{s=t}^{T} V_{its}\right) \tag{5–55}$$

$$\textbf{Subject to} \quad \sum_{t=1}^{s} V_{its} \geq \Delta_{is}, \quad \forall s \in T \tag{5–56}$$

$$\Delta_{is}\delta_{it} - V_{its} \geq 0, \quad \forall t, s \in T,\ s \geq t \tag{5–57}$$

$$\delta_{it} \leq 1, \quad \forall t \in T \tag{5–58}$$

$$V_{its}, \delta_{it} \geq 0, \quad \forall t, s \in T,\ s \geq t. \tag{5–59}$$

We can use this model to provide a lower bound for the Benders decomposition algorithm given above. Let us rewrite the model **(MILP-2)** with the new subproblem as follows:

(MILP-3)

$$\textbf{Maximize} \quad \sum_{c\in C}\sum_{m\in M}\sum_{n\in N_m^c}\sum_{t\in T}(p_{nt} - w_{mnt}^c)Y_{mnt}^c - \sum_{m\in M} f_m Z_m$$
$$+\sum_{i\in I}\sum_{t\in T}\sum_{c\in C}\sum_{m\in M}\sum_{n\in N_m^c}\sum_{l=1}^{t} h_{it}u_{in}Y_{mnl}^c - \sum_{i\in I} v_i(Y)$$

$$\textbf{Subject to} \quad K_{mn_0t}^c = (D_{mt}^c - Y_{mn_0t}^c)(1-\alpha_{mn_0}^c), \quad \forall c \in C, m \in M, t \in T \tag{5–60}$$

$$K_{mn_kt}^c = \left(K_{mn_{k-1}t}^c - Y_{mn_kt}^c\right)(1-\alpha_{mn_k}^c), \quad \forall c \in C, m \in M, n_{k:k>1} \in N_m^c,\ t \in T \tag{5–61}$$

$$Y_{mnt}^c - D_{mt}^c Z_n \leq 0, \quad \forall c \in C,\ m \in M,\ n \in N_m^c,\ t \in T \tag{5–62}$$

$$Y_{mnt}^c, K_{mnt}^c \geq 0, \quad \forall c \in C,\ m \in M,\ n \in N_m^c,\ t \in T \tag{5–63}$$

$$Z_m \in \{0,1\}, \quad \forall m \in M, \tag{5–64}$$

where for any given vector $\tilde{\mathbf{Y}}$, the value $v_i(\tilde{\mathbf{Y}})$ is determined by solving the subproblem **(SP(i))** as follows:

$$\textbf{(SP(i))} \qquad v_i(\tilde{\mathbf{Y}}) = \textbf{Minimize} \quad \sum_{t \in T} \left(a_{it}\delta_{it} + c_{it} \sum_{s=t}^{T} V_{its} \right)$$

$$\textbf{Subject to} \quad \sum_{t=1}^{s} V_{its} \geq \Delta_{is}, \quad \forall s \in T \tag{5–65}$$

$$\Delta_{is}\delta_{it} - V_{its} \geq 0, \quad \forall t, s \in T, \ s \geq t \tag{5–66}$$

$$V_{its}, \delta_{it} \geq 0, \quad \forall t, s \in T, \ s \geq t. \tag{5–67}$$

Note that **(SP(i))** is a linear program, and we can solve it using the algorithm provided in Wagelmans et al. (1992), and obtain the objective function's value. Let μ_{is} and β_{its} be the dual variables associated with constraints (5–65), and (5–66), respectively. Then we can write the dual of **(SP(i))** as

$$\textbf{(D-SP}(i)\textbf{)} \qquad \textbf{Maximize} \quad \sum_{s \in T} \mu_{is}\Delta_{is}$$

$$\textbf{Subject to} \quad \sum_{s \in T, s \geq t} \beta_{its}\Delta_{is} \leq a_{it}, \quad \forall t \in T$$

$$\mu_{is} - \beta_{its} \leq c_{it}, \quad \forall t, s \in T, \ s \geq t$$

$$\beta_{its}, \mu_{is} \geq 0, \quad \forall t, s \in T, \ s \geq t.$$

Using duality theory, the primal and dual formulations can be interchanged. Therefore, we can use the above dual model **(D-SP(i))** instead of $v_i(\mathbf{Y})$. We can use this dual problem to obtain a feasible integer subproblem that we can use to obtain a lower bound. The formal algorithm can be given as follows:

Step 1: Choose an initial pair of vectors $\mathbf{Z}^0$ and $\mathbf{Y}^0$, and select an optimality tolerance ϵ. Solve (D-SP(i)) at $\mathbf{Y} = \mathbf{Y}^0$ for every component $i \in I$ by using the Dual Algorithm in Wagelmans et al. (1992) to obtain a feasible solution with $\mathbf{V}^0$ and δ^0 with a dual objective function value of SP_z^0. Set $LB = RMP_z^0 + \theta^0 - SP_z^0$. and let $(\bar{\mathbf{Y}}, \bar{\mathbf{Z}}, \bar{\mathbf{V}}, \bar{\delta}) = (\mathbf{Y}^0, \mathbf{Z}^0, \mathbf{V}^0, \delta^0)$ denote the initial incumbent solution.

Step 2: Solve the RMP with all previously generated cuts. Let $(\mathbf{Y}^*, \theta^*)$ denote an optimal solution to RMP, and let $UB = RMP_z^*$. If $UB - LB < \epsilon$, stop.

Step 3: Solve (D-SP(i)) at $\mathbf{Y} = \mathbf{Y}^*$ for every component $i \in I$ by using the Dual Algorithm to obtain a feasible solution with $\mathbf{V}^*$ and δ^* with a dual objective function value of SP_z^*. If $RMP_z^* + \theta^* - SP_z^* > LB$, set $LB = RMP_z^* + \theta^* - SP_z^*$ and update the incumbent solution. If $UB - LB < \epsilon$, stop; $(\bar{\mathbf{Y}}, \bar{\mathbf{Z}}, \bar{\mathbf{V}}, \bar{\delta})$ is an ϵ-optimal solution. Otherwise, solve (D-SP-LP(i)) at $\mathbf{Y} = \mathbf{Y}^*$ for every component $i \in I$, obtain the optimal dual, ξ^* and ζ^* vectors and add the corresponding cut (5–36) to the (RMP) formulation and return to Step 2.

We have performed a set of preliminary computational tests that show that this algorithm is able to solve problem instances that cannot be solved in CPLEX. Future research will include a broad set of computational tests to characterize the performance of the Benders decomposition approach across a wide range of parameter settings.

CHAPTER 6
CONCLUSION AND FUTURE RESEARCH DIRECTIONS

In this chapter, we conclude the dissertation by discussing the first four chapters, providing concluding remarks, and summarizing our contributions to the existing literature. We also briefly discuss future research directions based on the results of chapters.

6.1 Algorithms for Solving a Knapsack Problem with S-Curve Return Function

We discussed a single-resource allocation problem with nonlinear S-curve returns as a function of resource allocation. The resulting model is a specially structured class of nonlinear knapsack problems in which the objective function is neither concave nor convex. The structure of the S-curve, however, leads to a characterization of optimal solution properties that permits the development of practically useful solution algorithms. Because no previous work exists on this class of relevant problems, we provided a pseudo-polynomial time solution algorithm, as well as a polynomial-time approach for a special case in which the return functions differ by a constant at all investment levels. We also provide a fully polynomial time approximation scheme, which, given a solution tolerance ϵ, permits obtaining a solution within $\epsilon\%$ of an optimal solution value using an algorithm that is polynomial in the number of investment instruments. Future avenues for research include explicit consideration of the logistic functional forms found in much of the marketing literature on S-curves (see, e.g., Johansson (1979)). A further exploration of the system of equations defined by the generalized KKT conditions might also provide value in the development of algorithms for the general form of the problem where investment levels may take any real-valued number.

In Chapter 3, we analyzed a stochastic knapsack problem where the weights of items are Poisson distributed random variables and a penalty is assessed when the knapsack capacity is exceeded. We provided a polynomial-time solution for the continuous relaxation of this problem and a customized branch-and-bound algorithm to solve the

binary version of the problem. Computational tests on a set of randomly generated problem instances showed that our algorithm performs very favorably when compared with a commercial mixed-integer nonlinear solver. As a future research direction, other distributions of item weights can be considered, which will change the structure of the objective function and the solution algorithm.

6.2 A Facility Location Model with Safety Stock Costs

We discussed a supply chain setting where customers with stochastic demand are assigned to uncapacitated supply facilities. In Chapter 4, our model determines the location of facilities and the assignment of customers to supply facilities in order to minimize the total supply facility opening cost, customer-supply facility assignment cost and the safety stock costs at supply facilities. In the literature, similar problems have been investigated with a single-sourcing requirement for each customer. We relax this constraint and apply an upper bound on the number of facilities to which a customer can be assigned. Clearly, when this number equals one, we obtain a special case of the problem that enforces single-sourcing. Our goal was to characterize the difference between the costs of problems where demand-splitting is allowed and those that enforce single-sourcing.

The resulting location-inventory problem falls into a class of difficult mixed-integer nonlinear programming problems. The structure of the objective function, however, leads us to characterize the solution properties for some special cases. For the general problem, we proposed a generalized Benders decomposition algorithm. We implemented our algorithm and conducted a broad set of computational tests to analyze the effects of key parameters on the percentage difference in costs when demand-splitting is allowed and when single sourcing is required. According to our computational study, with the parameter settings we tested, this percentage difference can be as high as 6.82%.

The relative values of assignment and holding costs, the coefficient of variation of customer demands, and the fixed opening costs of facilities are the most important parameters affecting the optimal assignment of customers. Therefore, we analyzed the

effects of these parameters by using a range of settings. According to our computational study, both low and high levels of the ratio of assignment cost to holding cost lead to solutions where single sourcing is optimal (or near optimal). However, at intermediate values, the model benefits from risk pooling by mixing the demands of multiple customers at the open locations. Similarly, low and high levels of the coefficient of variation lead to solutions where single sourcing is optimal because either the assignment cost or the safety stock cost tends to dominate. At intermediate values, where there is a balance in the costs, the model benefits from multiple-sourcing. Furthermore, high values of fixed cost naturally lead to opening fewer facilities, which in turn leads to the assignment of customers to fewer locations. Therefore, as the fixed cost increases, the benefits of demand splitting decrease.

This research can be extended in a number of different ways. One possible extension would consider the addition of finite capacities to supply facilities. Another extension might consider adding a penalty cost for assigning a customer to more than one facility, instead of using a restriction on the number of facilities to which a customer can be assigned. An additional interesting extension considers service-level-dependent assignment costs, which reflect cases in which some facilities may require a higher service level and an increase in associated assignment cost. In this setting, customers may accept reduced service levels instead of paying higher costs. Thus, instead of defining pre-specified service levels at the supply facilities, we may treat facility service levels as decision variables.

6.3 Procurement Planning and Product Portfolio Design Problem

We discussed a product portfolio design problem with product substitutions. In Chapter 5, we assume that products are assembled from a number of components that are procured from an outside supplier. Demands for ideal products occur from a set of customer segments, and each ideal product can be substituted with a set of alternative products at a substitution cost, subject to the availability of the products in the substitution set. Each customer may leave the system without buying any product

at any time if her ideal product is not available and she does not wish to buy the offered substitute product. If a customer leaves the system without buying any product, a lost sale results. Our objective in this setting is to decide on which products to produce, how to satisfy the demand and how to procure components.

The problem includes an embedded component procurement plan, which is actually a lot sizing problem. We propose a Benders decomposition-based exact algorithm that uses this lot sizing problem as a subproblem. Since the lot sizing problem is a mixed-integer linear programming problem, we cannot use it directly as a subproblem for the decomposition algorithm. We, therefore, use its linear relaxation to obtain the dulas and rewrite the model by disaggregating the amount of components procured in each period, producing a tight formulation of the lot sizing problem, and use the new disaggregated model's objective function value to obtain better lower bounds.

We have performed a set of preliminary computational tests that show that this algorithm is able to solve medium to large problem instances. Future research will include a broad set of computational tests to characterize the performance of the Benders decomposition approach across a wide range of parameter settings.

This research can be extended in several different ways. We assume that demand is known. This assumption might be relaxed, and the dependency of demands on the set of substitutable products can be accounted for within the model. Another extension might account for customer service levels within each group of customer segments. For example, we might require that a certain percentage of all demands receive their ideal product.

APPENDIX A
GENERALIZED KARUSH KUHN-TUCKER OPTIMALITY CONDITIONS

When each $\mu_i(a_i)$ function is locally Lipschitz continuous, the generalized KKT conditions for [KPS] are *necessary* but are not *sufficient* for optimality (see Hiriart-Urruty Hiriart-Urruty (1978)). We will refer to a point that satisfies the generalized KKT conditions as a "KKT point", and let w be a KKT multiplier for the budget constraint and let λ_i (for all $i \in I$) be multipliers associated with the lower-bound (nonnegativity) constraints. Define $\partial\mu_i(a_i)$ as the set of subgradients of the function $\mu_i(\cdot)$ at a_i, with $\partial^+\mu_i(a_i)$ denoting the right directional derivative at a_i and $\partial^-\mu_i(a_i)$ denoting the corresponding left directional derivative. For our S-curves, the set of subgradients at a_i is equal to the interval $[\partial^-\mu_i(a_i), \partial^+\mu_i(a_i)]$ if a_i lies in the convex portion of the function, while the set of subgradients at a_i equals the interval $[\partial^+\mu_i(a_i), \partial^-\mu_i(a_i)]$ if a_i lies in the concave portion of the function. The generalized KKT conditions can be written as:

$$
\begin{aligned}
-\partial\mu_i(a_i) + w - \lambda_i \ni 0, \quad & \forall i = 1, \ldots, N, \\
w\left(\sum\nolimits_{i=1}^{n} a_i - A\right) = 0, \quad & \\
\lambda_i a_i = 0, \quad & \forall i = 1, \ldots, N, \\
\sum_{i=1}^{n} a_i \leq A, \quad & \\
w \geq 0, \quad & \\
\lambda_i \geq 0, \quad & \forall i = 1, \ldots, N.
\end{aligned}
$$

APPENDIX B
PROOF OF PROPOSITION 3

From Lemma (1) we know that this two-supplier, two-customer problem is a convex programming problem. Therefore the generalized KKT Conditions are necessary and sufficient for optimality. The KKT conditions for this problem can be written as follows:

$$c_{ij} + H\frac{x_{ij}}{\sqrt{\sum_{i\in I} x_{ij}^2}} - \mu_i - \beta_{ij} = 0 \qquad \text{for } i = 1,2 \text{ and } j = 1,2 \tag{A-1}$$

$$\mu_i(1 - \sum_{j\in J} x_{ij}) = 0 \qquad \text{for } i = 1,2 \tag{A-2}$$

$$\beta_{ij}x_{ij} = 0 \qquad \text{for } i = 1,2 \text{ and } j = 1,2 \tag{A-3}$$

$$1 - \sum_{j\in J} x_{ij} \leq 0 \qquad \text{for } i = 1,2 \tag{A-4}$$

$$x_{ij} \geq 0 \qquad \text{for } i = 1,2 \text{ and } j = 1,2 \tag{A-5}$$

$$\mu_i \geq 0 \qquad \text{for } i = 1,2 \tag{A-6}$$

$$\beta_{ij} \geq 0 \qquad \text{for } i = 1,2 \text{ and } j = 1,2 \tag{A-7}$$

For the given solution, $x_{11} = x_{22} = \alpha$ and $x_{12} = x_{21} = 1 - \alpha$ where $0 < \alpha < 1$, from condition (A-3) we set $\beta_{ij} = 0$ for $i = 1,2$ and $j = 1,2$. Since $x_{11} + x_{12} = x_{21} + x_{22} = 1$, condition (A-2) is already satisfied. From condition (A-1), we require $\mu_1 = c_{12} + H\frac{(1-\alpha)}{\sqrt{\alpha^2+(1-\alpha)^2}}$ and $\mu_2 = c_{21} + H\frac{(1-\alpha)}{\sqrt{\alpha^2+(1-\alpha)^2}}$. Thus we have $\mu_1 \geq 0$ and $\mu_2 \geq 0$. Hence, the given solution satisfies all KKT conditions from (A-1) to (A-7) and is therefore optimal. The value of the objective function, Z^{opt}, equals $c_{12} + c_{21} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2+(1-\alpha)^2}}$ which also equals $c_{11} + c_{22} + 2H\frac{\alpha}{\sqrt{\alpha^2+(1-\alpha)^2}}$.

Proof of Corollary 1: Define Z^{opt} be the objective function value for the given solution, $x_{11} = x_{22} = \alpha$ and $x_{12} = x_{21} = 1 - \alpha$, let Z^1 be the objective function value when $x_{11} = x_{22} = 1$; $x_{12} = x_{21} = 0$, let Z^2 be the objective function value when $x_{12} = x_{21} = 1$; $x_{11} = x_{22} = 0$, and finally let Z^3 be the objective function value when $x_{11} = x_{21} = 1$;

$x_{12} = x_{22} = 0$ (from symmetry Z^3 also gives the objective value when $x_{12} = x_{22} = 1$; $x_{11} = x_{21} = 0$). Then these objective function values can be calculated as follows:

$$
\begin{aligned}
Z^{opt} &= c_{12} + c_{21} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2 + (1-\alpha)^2}} \\
Z^1 &= c_{11} + c_{22} + 2H \\
&= c_{12} + c_{21} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2 + (1-\alpha)^2}} + 2H\left(1 - \frac{\alpha}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right) \\
&= Z^{opt} + 2H\left(1 - \frac{\alpha}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right) \\
Z^2 &= c_{12} + c_{21} + 2H \\
&= c_{12} + c_{21} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2 + (1-\alpha)^2}} + 2H\left(1 - \frac{1-\alpha}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right) \\
&= z^{opt} + 2H\left(1 - \frac{1-\alpha}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right) \\
Z^3 &= c_{11} + c_{21} + \sqrt{2}H (\text{or } c_{12} + c_{22} + \sqrt{2}H) \\
&= c_{12} + c_{21} + 2H\frac{(1-\alpha)}{\sqrt{\alpha^2 + (1-\alpha)^2}} + H\left(\sqrt{2} - \frac{1}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right) \\
&= z^{opt} + H\left(\sqrt{2} - \frac{1}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right)
\end{aligned}
$$

Let $\Delta Z^1 = Z^1 - Z^{opt}$, $\Delta Z^2 = Z^2 - Z^{opt}$ and $\Delta Z^3 = Z^3 - Z^{opt}$. Clearly, the objective function value of the minimum-cost single-sourcing solution minus that of the minimum-cost solution with customer demand splitting, $\Delta\overline{Z}_{min}$, equals the minimum of ΔZ^1, ΔZ^2 and ΔZ^3.

$$
\begin{aligned}
\Delta\overline{Z}_{min} &= \min\left\{\Delta Z^1; \Delta Z^2; \Delta Z^3\right\} \\
&= \min\left\{2H\left(1 - \frac{\alpha}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right); 2H\left(1 - \frac{1-\alpha}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right); H\left(\sqrt{2} - \frac{1}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right)\right\} \\
&= H\left[\min\left\{2\left(1 - \frac{\max\{\alpha, 1-\alpha\}}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right); \sqrt{2} - \frac{1}{\sqrt{\alpha^2 + (1-\alpha)^2}}\right\}\right]
\end{aligned}
$$

$$= H \times \rho(\alpha)$$

where $\rho(\alpha) = \left[\min\left\{2\left(1 - \frac{\max\{\alpha, 1-\alpha\}}{\sqrt{\alpha^2+(1-\alpha)^2}}\right); \sqrt{2} - \frac{1}{\sqrt{\alpha^2+(1-\alpha)^2}}\right\}\right]$.

REFERENCES

Afentakis, P., B. Gavish. 1986. Optimal lot-sizing algorithms for complex product structures. *Operations Research* **34(2)** 237–249.

Afentakis, P., B. Gavish, U. Karmarkar. 1984. Computationally efficient optimal solutions to the lot-sizing problem in multistage assembly systems. *Management Science* **30(2)** 222–239.

Akcay, Y., S.H. Xu. 2004. Joint inventory replenishment and component allocation optimization in an assemble-to-order system. *Management Science* **50(1)** 99–116.

Baker, K.R., M.J. Magazine, H.L.W. Nuttle. 1986. The effect of commonality on safety stock in a simple inventory model. *Management Science* **32(8)** 982–988.

Balakrishnan, A., J. Geunes. 2000. Requirements planning with substitutions: Exploiting bill-of-materials flexibility in production planning. *Manufacturing & Service Operations Management* **2(2)** 166–185.

Balakrishnan, A., J. Geunes. 2003. Production planning with flexible product specifications: An application to speciality steel manufacturing. *Operations Research* **51(1)** 94–112.

Barnhart, C., A.M. Cohn. 1998. The stochastic knapsack problem with random weights: a heuristic approach to robust transportation planning. *Proceedings of Tristan III, Puerto Rico*.

Bassok, Y., R. Anupindi, R. Akella. 1999. Single-period multiproduct inventory models with substitution. *Operations Research* **47(4)** 632–642.

Bazaraa, M.S., H. Sherali, C.M. Shetty. 2006. *Nonlinear Programming: Theory and Algorithms*. John Wiley & Sons, New York, NY.

Bilde, O., J. Krarup. 1977. Sharp lower bounds for the simple location problem. *Annals of Discrete Mathematics* **1** 79–97.

Bitran, G.R., A.C. Hax. 1981. Disaggregation and resource allocation using convex knapsack problems with bounded variables. *Management Science* **27(4)** 431–441.

Bretthauer, K.M., B. Shetty. 1995. The nonlinear resource allocation problem. *Operations Research* **43(4)** 670–683.

Bretthauer, K.M., B. Shetty. 2002a. The nonlinear knapsack problem - algorithms and applications. *European Journal of Operational Research* **138** 459–472.

Bretthauer, K.M., B. Shetty. 2002b. A pegging algorithm for the nonlinear resource allocation problem. *Computers & Operations Research* **29** 505–527.

Burke, G.J., J. Geunes, H.E. Romeijn, A. Vakharia. 2008. Allocating procurement to capacitated suppliers with concave quantity discounts. *Operations Research Letters* **36** 103–109.

Carraway, R.L., R.L. Schmidt, L.R. Weatherford. 1993. An algorithm for maximizing target achievement in the stochastic knapsack problem with normal returns. *Naval Research Logistics* **40** 161–173.

Chand, S., J.E. Ward, Z.K. Weng. 1994. A parts selection model with one-way substitution. *European Journal of Operational Research* **73** 65–69.

Chopra, S., P. Meindl. 2007. *Supply Chain Management: Strategy, Planning , and Operations*. 3rd ed. Prentice-Hall, New Jersey.

Daskin, Mark S., L.V. Snyder, R.T. Berger. 2005. *Facility location in supply chain design*, chap. 2. Springer, New York, 39–66.

Daskin, M.S., C.R. Coullard, Z-J.M. Shen. 2002. An inventory-location model: Formulation, solution algorithm and computational results. *Annals of Operations Research* **110** 83–106.

Daskin, M.S., S.H. Owen. 199. *Location Models in Transportation in Handbook of Transportation Science*. Kluwer Academic Publishers, Boston.

Dean, B.C., M.X. Goemans, J. Vondrak. 2004. Approximating the stochastic knapsack problem: the benefitt of adaptivity. *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*.

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

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

Ervolina, T.R., M. Ettl, Y.M. Lee, D.J. Peters. 2009. Managing product availability in an assemble-to-order supply chain with multiple customer segments. *OR Spectrum* **31** 257–280.

França, P.M., H.P.L. Luna. 1982. Solving stochastic transportation-location problems by generalized benders decomposition. *Transportation Science* **16(2)** 113–126.

Freeland, J.R., C.B. Weinberg. 1980. S-shaped response functions: Implications for decision models. *Journal of the Operational Research Society* **31(11)** 1001–1007.

Geoffrion, A.M. 1972. Generalized benders decomposition. *Journal of Optimization Theory and Applications* **10(4)** 237–260.

Geoffrion, A.M. 1974. Lagrangean relaxation for integer programming. *Mathematical Programming Study* **2** 82–114.

Gerchak, Y., M. Henig. 1989. Component commonality in assemble- to-order systems: Models and properties. *Naval Research Logistics* **36** 61–68.

Gerchak, Y., M.J. Magazine, A.B. Gamble. 1988. Component commonality with service level requirements. *Management Science* **34(6)** 753–760.

Ginsberg, W. 1974. The multiplant firm with increasing returns to scale. *Journal of Economic Theory* **9** 283–292.

Goel, A., P. Indyk. 1999. Stochastic load balancing and related problems. *Proceedings of the 40th Annual Symposium on Foundations of Computer Science*.

Hale, W.W., D.F. Pyke, N. Rudi. 2000. An assemble-to-order system with component substitution. *Proceedings of the 4th MSOM Conference*.

Henig, M. 1990. Risk criteria in a stochastic knapsack problem. *Operations Research* **38** 820–825.

Hillier, M.S. 2000. Component commonality in multi-period assemble-to-order systems. *IIE Transactions* **32** 755–766.

Hiriart-Urruty, J.B. 1978. On optimality conditions in nondifferentiable programming. *Mathematical Programming* **14** 73–86.

Hoc, H.H. 1982. Topological optimization of networks: A nonlinear mixed integer model employing generalized benders decomposition. *IEEE Transactions on Automatic Control* **27(1)** 164–169.

Hochbaum, D.S. 1995. A nonlinear knapsack problem. *Operations Research Letters* **17** 103–110.

Holthausen, D.M., Jr., G. Assmus. 1982. Advertising budget allocation under uncertainty. *Management Science* **28(5)** 487–499.

Horst, R., P.M. Pardalos, N.V. Thoai. 1995. *Introduction to Global Optimization*. Kluwer Academic Publishers, London, UK.

Hsu, V.N., C-L. Li, W-Q. Xiao. 2005. Dynamic lot size problems with one-way product substitution. *IIE Transactions* **37** 201–215.

Johansson, J. 1979. Advertising and the s-curve: A new approach. *Journal of Marketing Research* **16** 346–354.

Kleywegt, A., J.D. Papastavrou. 1999. The dynamic and stochastic knapsack problem. *Operations Research* **46** 17–35.

Kleywegt, A., J.D. Papastavrou. 2001. The dynamic and stochastic knapsack problem with random sized items. *Operations Research* **49** 26–41.

Kleywegt, A., A. Shapiro, T. Homem de Mello. 2001. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization* **12(2)** 479–502.

Kodialam, M.S., H. Luss. 1998. Algorithms for separable nonlinear resource allocation problems. *Operations Research* **46(2)** 272–284.

Krarup, J., O. Bilde. 1977. *Plant Location, Set Covering and Economic Lot Size: An $O(nm)$-Algorithm for Structured Problems*, chap. Numerische Methoden bei Optimierungsaufgaben, Band 3: Optimierung bei Graph-Theoretischen und Ganzzahligen Problemen. Birkhauser Verlag, Basel, Switzerland, 155–180.

Lodish, L.M. 1971. Callplan: An interactive salesman's call planning system. *Management Science* **18(4)** 25–40.

Lu, Y., J-S. Song, D.D. Yao. 2003. Order fill rate, leadtime variability, and advance demand information in an assemble-to-order system. *Operations Research* **51(2)** 292–308.

Luss, H., S.K. Gupta. 1975. Allocation of effort resources among competing activities. *Operations Research* **23(2)** 360–366.

Martello, S., P. Toth. 1990. *Knapsack Problems - Algorithms and Computer Implementations*. Wiley, New York, NY.

Mathur, K., H.M. Salkin, S. Morito. 1983. A branch and search algorithm for a class of nonlinear knapsack problems. *Operations Research Letters* **2(4)** 155–160.

Melo, T., S. Nickel, F. Saldanha da Gama. 2007. *Facility Location and Supply Chain Management - A Comprehensive Review*, chap. Berichte des Fraunhofer ITWM, Nr. 130.

Merzifonluoğlu, Y., J. Geunes, H.E. Romeijn. 2009. The static stochastic knapsack problem with normally distributed item sizes. Working paper, University of Florida, Department of Industrial and Systems Engineering, Gainesville, FL, 2008.

Miranda, P.A., R.A. Garrido. 2006. A simultaneous inventory control and facility location model with stochastic capacity constraint. *Networks and Spatial Economics* **6** 39–53.

Moré, J.J., S.A. Vavasis. 1990. On the solution of concave knapsack problems. *Mathematical Programming* **49(3)** 397–411.

Morin, T.L., R.E. Marsten. 1976. An algorithm for nonlinear knapsack problems. *Management Science* **22(10)** 1147–1158.

Norkin, V.I., Y.M. Ermoliev, A. Ruszczynski. 1998. On optimal allocation of indivisibles under uncertainty. *Operations Research* **46(3)** 381–395.

Nozick, L.K., M.A. Turnquist. 1998. Integrating inventory impacts into a fixed-charge model for locating distribution centers. *Transportation Research Part E* **34(3)** 173–186.

Nozick, L.K., M.A. Turnquist. 2001a. Inventory, transportation, service quality and the location of distribution centers. *European Journal of Operational Research* **129** 362–371.

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

Owen, S.H., M.S. Daskin. 1998. Strategic facility location: A review. *European Journal of Operational Research* **111(3)** 423–447.

Ozsen, L., C.R. Coullard, M.S. Daskin. 2008a. Capacitated warehouse location model with risk pooling. *Naval Research Logistics* **55** 295–312.

Ozsen, L., C.R. Coullard, M.S. Daskin. 2008b. Facility location modeling and inventory management with multi-sourcing. *forthcoming in Transportation Science* .

Papastavrou, J.D., S. Rajagopalan, A. Kleywegt. 1996. The dynamic and stochastic knapsack problem with deadlines. *Management Science* **42** 1706–1718.

Pentico, D. 1974. The assortment problem with probabilistic demands. *Management Science* **21(3)** 286–290.

Pentico, D. 1976. The assortment problem with nonlinear cost functions. *Operations Research* **24(6)** 1129–1142.

Rao, U.S., J.M. Swaminathan, J. Zhang. 2004. Multi-period invenotory planning with downward substitution, stochastic demand and setup costs. *IIE Transactions* **36** 59–71.

Romeijn, H.E., J. Geunes, K. Taaffe. 2007. On a nonseparable convex maximization problem with continuous knapsack constraints. *Operations Research Letters* **35(2)** 172–180.

Romeijn, H.E., F.Z. Sargut. 2009. The stochastic transportation problem with single sourcing. Working Paper, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL, 2007.

Rosenthal, R.E. 2007. Gams - a user's guide.

Rosling, K.i̇1989. Optimal inventory policies for assembly systems under random demands. *Operations Research* **37(4)** 565–579.

Ross, K.W., D.H.K. Tsang. 1989. The stochastic knapsack problems. *IEEE Transactions on Communications* **37** 740–747.

Shen, Z-J. M. 2005. A multi-commodity supply chain design problem. *IIE Transactions* **37** 753–762.

Shen, Z-J. M., D. Coullard, M.S. Daskin. 2003. A joint location-inventory model. *Transportation Science* **37(1)** 40–55.

Shen, Z-J. M., M.S. Daskin. 2005. Trade-offs between customer service and cost in integrated supply chain design. *Manufacturing & Service Operations Management* **7(3)** 188–207.

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

Simon, J.L., A.J. Arndt. 1980. The shape of the advertising response function. *Journal of Advertising Research* **20(4)** 11–27.

Smith, S.A., N. Agrawal. 2000. Management of multi-item retail inventory system with demand substitution. *Operations Research* **48(1)** 50–64.

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

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

Sourirajan, K., L. Ozsen, R. Uzsoy. 2007. A single-product network design model with lead time and safety stock considerations. *IIE Transactions* **39** 411–424.

Steinberg, E., M.S. Parks. 1979. A preference order dynamic program for a knapsack problem with stochastic rewards. *Journal of the Operational Research Society* **30** 141–147.

Sun, X.L., F.L. Wang, D. Li. 2005. Exact algorithm for concave knapsack problems: Linear underestimation and partition method. *Journal of Global Optimization* **33** 15–30.

Taşkın, Z.C., A.T. Ünal. 2009. Tactical level planning in float glass manufacturing with co-production, random yields and substitutable products. *European Journal of Operational Research* **199** 252–261.

Tuy, H. 2000. Monotonic optimization: Problems and solution approaches. *SIAM Journal on Optimization* **11(2)** 464–494.

Van Hoesel, C.P.M., A.P.M. Wagelmans. 2001. Fully polynomial approximation schemes for single-item capacitated economic lot-sizing problems. *Mathematics of Operations Research* **26(2)** 339–357.

Vidyarthi, N., E. Celebi, S. Elhedhli, E. Jewkes. 2007. Integrated production-inventory-distribution system design with risk pooling: Model formulation and heuristic solution. *Transportation Science* **41(3)** 392–408.

Wagelmans, A., S.V. Hoesel, A. Kolen. 1992. Economic lot sizing: An $o(n \log n)$ algorithm that runs in linear time in the wagner-whitin case. *Operations Research* **40(1)** 145–156.

Yunes, T.H., D. Napolitano, A. Scheller-Wolf, S. Tayur. 2007. Building efficient product portfolios at john deere and company. *Operations Research* **55(4)** 615–629.

Zipkin, P.H. 1980. Simple ranking methods for allocation of one resource. *Management Science* **26(1)** 34–43.

Zoltners, A.A., P. Sinha. 1980. Integer programming models for sales resource allocation. *Management Science* **26(3)** 242–260.

Zoltners, A.A., P. Sinha, P.S.C. Chong. 1979. An optimal algorithm for sales representative time management. *Management Science* **25(12)** 1197–1207.

BIOGRAPHICAL SKETCH

Semra Ağralı was born in Malatya, Turkey, in 1980. She graduated from high school, Malatya Anadolu Lisesi, in 1998. She received her B.S. degree in industrial engineering from İstanbul Technical University in 2003. Upon graduation, she attended Koç University, where she received her master's degree in industrial engineering in 2005. In August 2009, she received her Ph.D. degree in industrial engineering from the Department of Industrial and Systems Engineering at the University of Florida. Following graduation, she will join the faculty of the Department of Industrial Engineering at Bahçeşehir University.