North Carolina Agricultural and Technical State University
Aggie Digital Collections and Scholarship

Dissertations | Electronic Theses and Dissertations

2014

Designing Medical Treatment Protocols To Improve Healthcare Supply Chain Management

LaKausha Tanette Simpson
North Carolina Agricultural and Technical State University

Follow this and additional works at: https://digital.library.ncat.edu/dissertations

Recommended Citation

Simpson, LaKausha Tanette, "Designing Medical Treatment Protocols To Improve Healthcare Supply Chain Management" (2014). *Dissertations*. 73.
https://digital.library.ncat.edu/dissertations/73

This Dissertation is brought to you for free and open access by the Electronic Theses and Dissertations at Aggie Digital Collections and Scholarship. It has been accepted for inclusion in Dissertations by an authorized administrator of Aggie Digital Collections and Scholarship. For more information, please contact iyanna@ncat.edu.

DESIGNING MEDICAL TREATMENT PROTOCOLS TO IMPROVE HEALTHCARE SUPPLY CHAIN MANAGEMENT

LaKausha Tanette Simpson

North Carolina A&T State University

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Department: Industrial & Systems Engineering

Major: Industrial & Systems Engineering

Major Professor: Dr. Paul Stanfield

Greensboro, North Carolina

2014

The Graduate School
North Carolina Agricultural and Technical State University
This is to certify that the Doctoral Dissertation of

LaKausha Tanette Simpson

has met the dissertation requirements of
North Carolina Agricultural and Technical State University

Greensboro, North Carolina
2014

Approved by:

Dr. Paul Stanfield Major Professor	Dr. Lauren Davis Committee Member
Dr. Xiuli Qu Committee Member	Dr. Steven Jiang Committee Member
Dr. Tonya Smith-Jackson Department Chair	

Dr. Sanjiv Sarin
Dean, The Graduate School

© Copyright by

LaKausha Tanette Simpson

2014

Biographical Sketch

LaKausha Tanette Simpson joined the staff of Hewlett Packard's Global Supply Chain Systems as a supply chain engineer in June 2013. Although she has been with the company less than one year, Simpson has made advances in integration for inventory optimization, assessments of social media systems, and evaluating supply chain performance and best practices.

Simpson spent the majority of her childhood in Tallahassee, Florida. She attended Godby High School where she was in the top of her class, class president, and an MVP on the girls' soccer team. Upon visiting the University of Florida during a middle school field trip, she quickly decided to major in Industrial and Systems Engineering. In 2007, Simpson obtained her B.S. in Industrial and Systems Engineering from the University of Florida. As a student at the university, she completed a four-term co-op program with the Walt Disney World Department of Industrial Engineering and conducted research at UF Health Shands Hospital to develop a quality assurance program.

Immediately after obtaining her B.S., Simpson enrolled at North Carolina A&T State University where she obtained a full fellowship along with a collection of scholarships for her M.S. and Ph.D. studies in Industrial and Systems engineering. She was award the Clare Boothe Luce Fellowship, NSBE Alumni Extension Technical Excellence Fellowship, Alfred P. Sloan Scholarship, and recognized as a Dr. Wadaran Latamore Kennedy Scholar. Simpson obtained her M.S. from NCA&T in 2009 with her thesis entitled "A Markov Chain Model for Quantifying the Value of Information from RFID in Healthcare Systems." As with her undergraduate career, she devoted the majority of her research efforts toward public health and healthcare systems.

Table of Contents

List of Figures

List of Tables

Abstract

The primary goal of this research is to determine the strategic system integration opportunities for a segmented healthcare system with cost minimization and efficacy maximization objectives. This research is inspired in part by the Defense Logistics Agency, which is trying to assess the impact of integrating treatment selection processes across service clinicians. Specifically, physician bias, patient volumes, leveraging economies of scale or costing structures, and complex treatment efficacy calculations are considered by mathematically modeling three forms of integration.

Multiple objective optimization problems are used to define efficient frontiers based on cost and treatment efficacy. A novel comparative analysis method is applied to measure improvements in efficient frontiers and a customized genetic algorithm solution is applied for the more complex treatment selection problem. Results indicate that more integrated treatment selection protocols lead to decreases in cost alongside increases in efficacy. Complex healthcare systems or systems with higher variability in performance factors are found to have the greatest opportunity for performance improvement.

The three studies in this research apply systems engineering concepts to flexibly characterize and parameterize systems; inform policy including characteristics of attractive treatments; and capture system dynamics and insights. However, this research is not intended to dictate treatments to health professionals; set policy or give practitioners optimal allocations; fully capture all of the intricacies of the treatment design process; or constrain research processes associated with treatment design.

CHAPTER 1

Introduction

The United States has the highest per capita healthcare costs in the world, yet lags behind many developed countries in terms of population health and other system quality measures. Pate (2008) estimates that the healthcare industry spends up to 54 percent of cost in waste each year. Within the next ten years, healthcare costs are expected to double to 4.5 trillion dollars, approximately one-fifth of the country's gross domestic product (Terry, 2010). This ratio implies that the United States will spend over 2.5 trillion dollars per year in preventable costs or system inefficiencies by the year 2020.

In recent years, a significant portion of domestic public policy has focused on healthcare. Concentrated policy debate seeking to address this issue has been ongoing since the early 1990's. Elected officials and system administrators have focused on healthcare accessibility and funding, with limited attempts toward optimizing cost reduction or healthcare quality. The most well-known and significant legislation came with the 2010 passage of the Patient Protection and Affordable Care Act. This act, like other policy efforts, seeks to encourage health system integration to decrease costs and improve effectiveness. Implementation of policy-based integration efforts is complicated by a variety of issues including patient privacy, technology, and patient and physician choice.

Much of the integration effort to date is heavily influenced by the fields of public policy and healthcare economics. Yet, much of the integration potential involves medical supply chain and information technology integration. Consequently, this dissertation provides a quantitative approach to influencing healthcare policy with emphasis on supply chain and information system integration.

1.1 Healthcare System Integration

Specifically, this research is motivated by the system segmentation challenges, and the associated integration opportunities, faced by the U.S. Department of Defense (DoD) Defense Logistics Agency (DLA). These challenges and opportunities are reflective of those within the broader healthcare system.

1.1.1 Defense Logistics Agency integration trends. The DLA is the Medical Material Executive agent for the DoD and is responsible for end-to-end DoD supply chain logistics. Though separated from other categories of DLA-managed products, the healthcare mission is consistent with the broader DLA mission of consolidating and optimizing supply chains across the defense enterprise. This mission is expressed in its National Inventory Management Strategy (NIMS) to integrate DLA and service (e.g. Army, Navy, or Air Force) inventories into a single system to take advantage of asset pooling and other supply chain best practices.

To facilitate integration, the DLA utilizes an electronic information system for medical purchases. The system is referred to as the Theater Enterprise Wide Logistics System (TEWLS). Using DLA funds, military services purchase medical supplies via TEWLS. The system is capable of managing supply catalogs, orders, and invoices, as well as, the financial accounting required for procurement in the enterprise business system.

1.1.2 Defense Logistics Agency segmentation challenges. Despite the intent of the NIMS and the function of the TEWLS, significant segmentation remains. Individual services are responsible for determining treatment protocols and managing associated order payments. DLA has determined that different treatment protocols are used across different services for the same patient condition. Service process managers may use different carrier services for the same set of supplies with similar destinations.

When investigated, the DLA finds these non-standardized treatment protocols have no underlying explicit rational. The creation and persistence of such protocols is driven by two factors: disparate views of treatment efficacy and limited view by treatment designers (note: treatment designer is synonymous with physician in this research) of cost / supply chain impact. Additionally, the service medical organization is increasingly being divided into specialists with a sub-discipline biased view of a patient. The physician treats the patient according to the disease associated with the specialty.

This segmentation leads to increased uncertainty, increased costs, and, likely, decreased treatment efficacy. The impact of integration is further undermined as users make use of legacy manual operations instead of exploiting the automation of TEWLS.

1.1.3 US Healthcare System integration trends. Many of the integration trends and segmentation challenges found in DLA are mirrored and magnified in the US system. Independent hospitals have united into regional healthcare systems to take advantage of economies of scale. Group purchasing organizations (GPOs) have been created to translate volume discount purchases into lower costs.

As with the DLA, much of the integration potential for the US healthcare system is related to information systems. During early years, healthcare systems deployed information systems as basic data storage systems to maintain schedules, patient treatment records, and financial information. As technologies and analytics have evolved, systems are able to leverage healthcare data to inform patient treatments, optimize logistics, and provide better coordination for patient care.

Studies suggest that the adoption of emerging technologies for healthcare supply chain management can mitigate healthcare concerns with coordination and communication (Zhenga et

al., 2006). Many healthcare systems have implemented enterprise planning systems intended to facilitate cost reduction and improve work process to improve patient care. Going forward, systems like the National Health Information Network (NHIN) are being developed to improve the quality and efficiency of public hospitals through establishing a mechanism for nationwide health information exchange.

1.1.4 US Healthcare System segmentation challenges. The same segmentation issues exist within the US system as in the DLA. Cost and efficacy is heavily influenced by treatment selection at the individual or regional healthcare system level. Physicians (as treatment designers) are shielded from a clear understanding of cost structure. Physicians have few mechanisms to facilitate a collaborative process which might yield a system's views of efficacy and enable opportunities for standardization. As with DLA, the impact of increased physician specialization also is a major source of segmentation in the US system.

The impact of information system integration is still limited. For example, providers are reluctant to utilize shared information systems such as NHIN at the risk of compromising patient confidentially, competitive advantages, or exposing system failures.

Segmentation also exists among healthcare providers abroad. Blendon et al. (2003) survey issues in healthcare across five countries and discover high levels of patient dissatisfaction with medical errors, communication, and coordination. As many U.S. providers have implemented coordinated healthcare delivery practices to mitigate Medicare expenditures, Peikes (2009) concluded that viable coordination programs do not yield significant Medicare savings or improvements in quality.

1.2 Problem Statement

Based on the challenges listed above, three forms of segmentation may be identified: (1) separation of the physician / treatment designer from a systemic view of efficacy, (2) separation of the physician / treatment designer from an understanding of supply chain cost structure, and (3) separation of a patient condition according to specialization. Information system integration efforts might assist in overcoming issues associated with the segmentation. However, integration can be expensive and complex. As a result, this dissertation seeks to solve the problem: *how might one model and analyze pre- and post-integration system performance in a manner that enables understanding of key system dynamics and prioritization of effort*?

1.3 Research Objective

The primary goal of this research is to determine the strategic opportunities of system integration for a segmented healthcare system seeking to balance cost and efficacy. Specifically, the research serves to achieve five objectives:

1. *Characterize forms of segmentation and associated integration opportunities*. Three different forms of segmentation are defined and considered associated with physician/system efficacy valuation, supply chain behavior, and specialized medicine (to include efficacy structure).
2. *Represent pre- and post-integration policies in general quantitative models facilitating optimization solution*. Mathematical models build upon the proven systems engineering principle that supply chain costs are best influenced during the design stage. Treatment protocol determination provides the analog with product design. The intent of these models is not to solve for implementable treatment protocols, but to represent the function of rational systems given the associated integration policy. Four such models are developed.

3. *Develop solution approaches which enable the discovery of characteristics of "good solutions."* For the different models developed in objective two, solution methods are created, ranging from enumerative to heuristic to meta-heuristic. The solution methods yield further understanding between the relationships of "good solutions."
4. *Recognize situations which present key integration opportunities*. Based on factors exposed during the model formation, experiments are designed and conducted to demonstrate situations where integration might provide the most value.
5. *Suggest specific policy adjustments based on the research insights*. Throughout the research, inferences are made from experimental results to determine the managerial, technical, and social implications of treatment designs.

The issue of treatment design to promote cost effectiveness is highly sensitive due to its potentially critical impact on a quality of life. As a result, the limitations of this research are explicitly stated. This research does not intend to do the following:

1. *Dictate treatments to health professionals*. The key relationship in healthcare is that between the physician (or other healthcare professionals) and the patient. The physician should retain the choice of treatment, not be replaced by some automated optimization approach.
2. *Fully capture all the intricacies of the treatment design process*. The models in this dissertation depict general situations relative to cost/efficacy tradeoff and associated dynamics. The models are not intended to capture all of the relevant information needed for treatment design of a specific set of diagnoses.
3. *Set policy or give optimal treatment allocations*. Due, in part to items one and two above, the models are never expected to be used to suggest a treatment for a specific set of patient types.

4. *Constrain research process associated with treatment design*. Treatment design is an expensive and sophisticated process including much investment in research and development. The impact of these costs and associated strategic decisions are not the main considerations in model development (though fixed costs are included).

1.4 Research Significance

This research is innovative in that it borrows from core concepts found in systems engineering, specifically in product design to explore opportunities in medical treatment protocol design. Similar to the objectives in product design, this research seeks to determine the policies for treatment design that minimize costs while maximizing efficacy for patients. System transparency, process standardization, and resource consolidation are shown to lead to efficiency.

The research proposes a novel systems engineering approach to considering system integration opportunities. Quantitative models are built as generalized representations of pre- and post-integration systems and used to emulate rational system behavior. The models are solved under a variety of situational factors to better understand the nature of the integration opportunity.

For optimization modeling and simulation, modification to the traditional multi-objective optimization problem (MOOP) are proposed. Extensions to multi-objective representation, solution, and performance management are noted as progressively complex models are formed. The models incorporate the idea that patients may seek care for multiple health problems through varying physicians and that treatments can affect more than one diagnosis for a patient. The idea the supply chain cost and treatment efficacies are nonlinear and vary by volume of system usage makes the research model more robust.

1.5 Dissertation Outline

The remainder of the dissertation is organized as follows. Chapter 2 includes a description of the integration process, treatment selection concept, modeling, experimentation, and analysis consistent across the three chapters that follow it.

In Chapters 3 through 5, each of three different forms of integration are considered. Figure 1.1 illustrates these forms of integration along with the models and characteristics associated with the system before or after integration. The figure also shows the associated forms of segmentation. Although the integration is shown as progressive, the integration need not occur in this same sequence.

Efficacy	Supply Cost	Patient View	Treatment Flexibility
Individual	Unit	Disease / Specialty	Binary
Efficacy Integration (Information Visibility)			
System	Unit	Disease / Specialty	Binary
Supply Chain Integration (Life Cycle Engineering)			
System	System	Disease / Specialty	Binary
Whole Patient ; Cost Efficacy Relationship Integration			
System	System	Whole Patient	Real

Figure 1.1. Three forms of healthcare system integration.

Chapter 3 investigates limited information visibility or bias in physician efficacy valuation influences system performance. A review of physician tendencies in assessing efficacy and cost is provided. Supply chain cost impact is evaluated using efficient frontiers of total treatment cost versus total treatment efficacy. Techniques to mitigate physician (or institutional) bias are proposed along with strategies to make practitioners informed participants of medical supply chain management.

Chapter 4 investigates physician understanding of supply chain cost structure and its impact in designing medical treatment protocols. An optimization model is used to help determine treatment selection strategies change under varying costing structures.

Chapter 5 builds on the preceding chapters to examine whole patient consideration and more detailed efficacy models might impact system cost effectiveness. A modified multi-objective optimization problem is modeled to recognize synergies in treatment efficacies and patient populations to achieve economies of scale.

Chapter 6 concludes this dissertation with an overview of the research methodology and a discussion of possible extensions of this research. This chapter provides a consensus of the conclusions found in the three research studies. Key findings are highlighted and presented with strategic implications.

Throughout the document, associated literature is reviewed and described as an integral part of the text. This approach is taken due to the multi-disciplinary nature of the research.

CHAPTER 2

Research Foundation and Method

This dissertation provides a study of the impact of three forms of healthcare system integration that might positively influence healthcare system cost (with emphasis on supply costs) and quality. The three forms are considered progressively and associated with three forms of segmentation introduced in Chapter 1. For each form of integration, a consistent process is undertaken to enable modeling and analysis of pre- and post-integration system performance in a manner that enables understanding of key dynamics and prioritization of integration effort. The process includes the following steps:

1. Review literature relevant to the identified form of segmentation to characterize its impact and develop an associated integration concept.
2. Create generalized quantitative models which might be solved to mimic rational system behavior for pre- and post-integration policies. Again, the intent of these models is not to solve for implementable treatment protocols. Four such models are developed. The models use treatment design as the decision variable with a multi-objective goal to minimize cost and maximize efficacy.
3. For each model developed, a solution method(s) is created, ranging from enumerative to heuristic to meta-heuristic. The solution methods yield further understanding between the relationships of "good solutions." As a result, solution methods are constructed to produce "efficient frontiers."
4. Based on the characterization in step one, key environmental factors are identified and an associated experimental design constructed and executed.

5. Experimental output is compared and analyzed in order to determine the characteristics of high leverage integration opportunities.
6. Inferences are made from experimental results to determine the managerial, technical, and social implications of treatment designs.

This process is generalizable for a number of policy areas where system integration might be considered and forms a key part of the contribution of this research.

The three forms of integration are addressed in Chapters 3 through 5 respectively. The remainder of this chapter provides an overview of the common foundation across all the research organized according to the process above.

2.1 System Segmentation and Integration Opportunities

Each chapter begins with a review of literature relevant to identified form of segmentation. This review is used to characterize segmentation impact and develop an associated integration concept.

2.1.1 Segmentation types. The three forms of segmentation were revealed in the case of the Defense Logistics Agency and reflected in the US healthcare system as a whole. These forms of segmentation are:

1. Those making treatment decisions are often separated from a system understanding of the effectiveness of the treatment for the given patient condition. Treatment decisions may be based on past experience often limited in frequency. Additionally, pharmaceutical companies spend significant marketing dollars to influence physician views and, more recently, to influence patient views. Deviations in efficacy evaluation for an individual physician (or healthcare system) from a comprehensive system view, is termed treatment

bias. The existence of such bias, even when unintentional on the part of the physician, can undermine system performance.

2. Those making treatment decisions are often separated from an understanding of supply chain cost structure. Doctors are often unaware of the costs associated with the treatments they are designing. Even when cost is a consideration, variable unit costs are considered. This view might not lead to cost efficient solutions given common supply chain cost structures.
3. Patient treatment is separated according to physician specialization. This approach tends to drive a reactive approach to healthcare. It leads to different treatment designs being administered to the same patient. Problems associated with this include limited proactive actions, missed opportunities for treatment synergy, and potentially harmful side effects from treatment mixing. Resolving this form of segmentation uses a novel form of efficacy representation with enables consideration of the sensitive subject of healthcare rationing.

These forms of segmentation are described in more detail in Chapters 3 through 5.

2.1.2 Integration opportunities. For each case of segmentation, the mechanism suggested to facilitate integration is information systems. For a number of years, healthcare information systems have been used to automate business processes and in clinical applications such as diagnoses, therapy, and surgery (Kulkarni, 2006; Mantas, 1992; Sneider, 1987). More recently, healthcare reforms and alliances are prompting greater focus on integrating and coordination throughout healthcare delivery systems worldwide (Kodner and Spreeuwenberg, 2002). Regulations such as HIPAA and requirements for cost reductions and quality improvement are prompting greater use of shared information systems (Fadlalla & Wickramasinghe, 2004).

Despite the availability of healthcare information systems, enterprise integration has always been problematic and slow in healthcare organizations (Khoumbati et al., 2006). Over time, the implementation of healthcare information systems has become more complex due to concerns for quality and integrity; the growth and evolution of the clinical services; hardwired legacy systems; and the competitive nature of the healthcare industry (Berger & Ciotti, 1993; Mandke, Bariff, & Nayar, 2002; Teisberg & Harvard Business Review, 1994). Adding to these concerns, clinician awareness and acceptance is often regarded as a leading barrier to successful adoption of healthcare innovations (Omachonu and Einspruch, 2010; Birken et al., 2012; and Holden and Karsh, 2010). These difficulties provide the justification for this research, which serves to identify integration opportunities with good value.

2.2 Policy Modeling

For this research, policies are modeled with treatment design as the decision variables and dual objectives of cost minimization and efficacy (measure of healthcare system quality) maximization. The model constraints are created to mimic rational behavior under the associated policy.

Four policy models are needed to provide pre- and post-integration models for each of the three integration forms. Policies in the system increase in terms of complication and integration as one moves from the top to the bottom of Figure 2.1. Also note that the policy models are titled according to the state of the segmentation factors related to efficacy, supply chain cost, and patient view.

Efficacy	Supply Cost	Patient View	Treatment Flexibility	Policy Model
Individual	Unit	Disease / Specialty	Binary	IUD
↓ Efficacy Integration (Information Visibility) ↓				
System	Unit	Disease / Specialty	Binary	SUD
↓ Supply Chain Integration (Life Cycle Engineering) ↓				
System	System	Disease / Specialty	Binary	SSD
↓ Whole Patient ; Cost Efficacy Relationship Integration ↓				
System	System	Whole Patient	Real	SSW

Figure 2.1. Policy models developed in the research.

2.2.1 Treatment selection. Many cost efforts in healthcare attempt to reduce costs for an already established mix of supply. This research builds on the system engineering concept that the key opportunity for impact on supply chains is at the product design stage. The analogous decision in the healthcare realm is the selection of treatment protocol by the physician or healthcare provider. The term "treatment designer" is used for this role in the research.

Specifically, most healthcare expenditures in the U.S. are based on physician or healthcare system decisions. Therefore, it is important to consider clinician perspectives when examining treatment selection strategies. Studies suggest that many "decisions regarding medical tests and treatments are influenced by factors other than the expected benefit to the patient, including the doctor's demographic characteristics and concerns about cost and income" (Bovier et al., 2005). Fortunately, recent studies are proving that healthcare integration systems have significant, positive impacts on prescribing behaviors (Fortuna et al., 2009).

Note that a treatment protocol may consist of a single medicine or a combination of medicine. Additionally, a treatment protocol might consist of a mix of proactive and reactive treatment or a decision to "do nothing". This generalization of treatment protocol provides much

flexibility and enables more focus on the policy aspect of this research (rather than on solving specific treatment design problems).

2.2.1.1 Patient condition / disease. The selection of treatment protocol is made in response to the condition of the healthcare client – the "patient." In the systems engineering analogy, design decisions are made in an effort to provide good value in meeting customer needs. Similarly, in healthcare, treatment designs should be made in a manner to address health concerns, proactively and reactively, in a cost efficient manner.

The view of the patient in light of specialization influences the understanding of the condition. For instance, a healthcare provider with an emphasis on reactive care might see the patient as a specific disease or illness. This "disease" view is consistent with the majority of current practice and is taken in Chapters 3 and 4. Chapter 5 addresses systems that enable "whole patient" views.

2.2.1.2 Assignment problem. Establishing the connection between a patient type or disease and a treatment protocol is a form of the classic assignment problem. For this situation, operations researchers often make use of assignment models to determine the most appropriate allocation of resources (Winston & Goldberg, 2004). A brief review of assignment problem approaches follows.

The goal of any assignment problem is to determine the best solution for assigning some set of items to another set of items. There are several different techniques or assignment problem formulations designed to reach this goal according to vary system constraints. The main difference between assignment problems is the use of cost / distance matrices for sites, resources allocated to those sites, and the criteria or restrictions given in addition to the requirements of the basic assignment problem (Levin, 2004). Figure 2.2 exhibits this taxonomy.

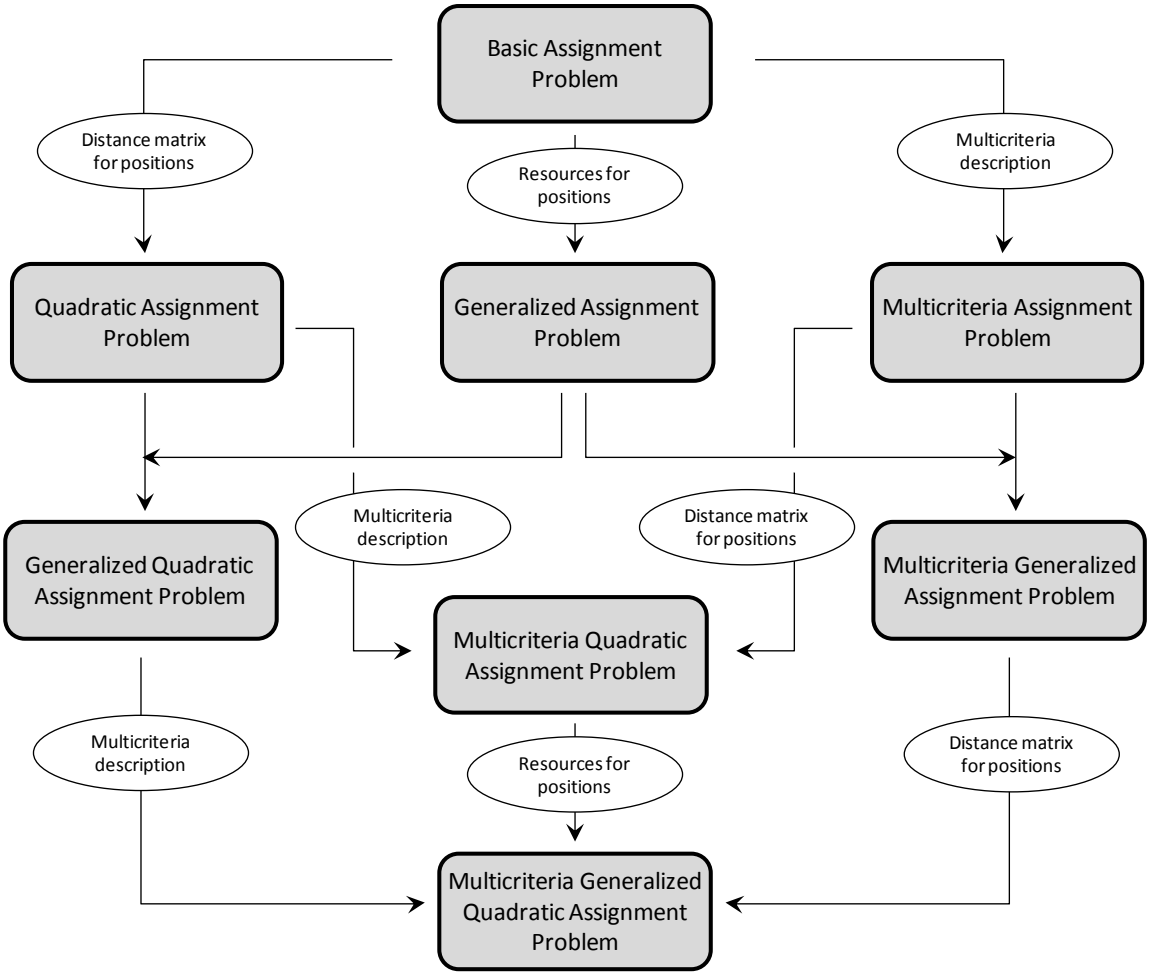

Figure 2.2. Evolution of the basic assignment problem adapted from Levin (2004).

Assignment problem differences can be denoted by the terms quadratic, generalized, and multi-criteria. Any quadratic assignment problem is given added dimensionality to account for the value of distances between sites that will be important in allocating assignments. When resources are to be constrained to different positions, the problem becomes generalized to account for the requirements of subsystems. As additional system constraints are added, the problem is considered to have multi-criteria.

There are many specific assignment problems that fit within the categories outlined in Figure 2.2. The following sections of this chapter discuss the formulation and implications of four common assignment problems and describe how they may be used to allocate resources.

Specifically, the properties of the mapping problem, general assignment problem, quadratic assignment problem, and weapon-target assignment problem are explained.

2.2.1.2.1 Matching problem. The matching problem is fundamental in graph theory as it seeks to find the optimal pairwise relationships for a set (Goemans, 2009). The focus of this section is the minimum weight perfect matching problem for a bipartite graph. Consider the graph shown in Figure 2.3. The set of elements in the graph are partitioned into two parts, thus the name bipartite graph.

The goal of this type of matching problem is to find the minimal number of connections called "edges" between the partitioned sets such that the total cost of said connections is minimal. Additionally, the problem is said to be a perfect matching problem if none of the nodes remain exposed, that is, there exist a connection across the partition for each node. A perfect matching problem for a bipartite graph is also referred to as an assignment problem.

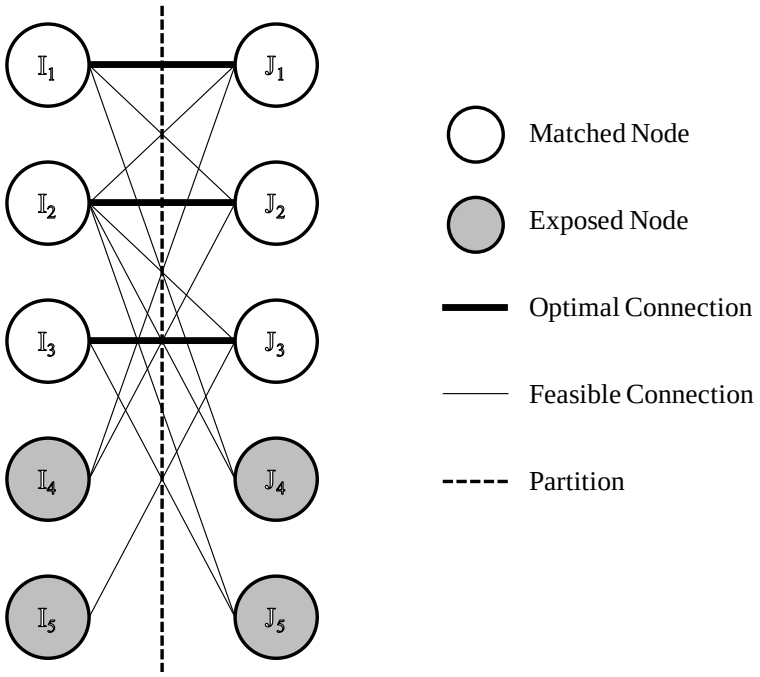

Figure 2.3. Illustration of bipartite graph matching problem.

2.2.1.2.2 General assignment problem. Assignment models have been used commonly in healthcare for scheduling staff members (Day, 1985; Ozkarahan, 1991). These models have been used to find solutions for more efficient treatment processes as well (Wang-Rodriguez, Mannino, Liu, & Lane, 1996). On a larger scale, scientists have long argued that the use of assignment models can be instrumental in quantifying performance and provide an effective tool for selecting operational priorities, or strategic planning (Hannan, O'Donnell, & Freedland, 1981; Lusk, 1979).

Consider a system in which a finite set of resources may be used to complete a number of tasks. Some cost is incurred whenever a resource is assigned to a task. All tasks must but completed and each task is assigned exclusively to one resource. For this scenario, the general assignment problem may be used to determine the impact of various assignment strategies or the structure of the most optimal resource assignment strategy (Eiselt & Sandblom, 2000).

First, let the decision to assign resource i to task j be represented by x_{ij}. Decision variable x_{ij} will equal one if resource i is assigned to task j and zero otherwise. Second, assume that a cost c_{ij} will be incurred if resource i is assigned to task j. Then, the nonlinear integer programming model to solve the general assignment problem is formulated as follows:

$$\min \sum_i \sum_j c_{ij} x_{ij} \tag{2.1}$$

$$\text{s.t. } \sum_i x_{ij} = 1 \ \forall \ j \tag{2.2}$$

$$\sum_i x_{ij} = 1 \ \forall \ i \tag{2.3}$$

$$x_{ij} = 0 \text{ or } 1 \ \forall \ i \text{ and } j \tag{2.4}$$

The objective function for this model ensures that the lowest cost strategy is chosen for assignments. Equation 2.2 ensures that each task is assigned exactly once. Equation 2.3 ensures

that each resource is assigned to complete exactly one task. Equation 2.4 is a binary constraint for the decision variable.

Goemans (2009) provides insight on the characteristics of the general assignment problem. If the integrality constraints of the general assignment problem are relaxed the problem takes on an infinite number of fractional solutions. With the relaxation, the problem linear program is polynomially solvable. However, with integrality constraints the problem becomes nonlinear and more constrained than the linear program. Integrality constraints imply that the minimizing solution for the integer program no less than the minimizing solution for the linear program.

Of all assignment problems discussed in this chapter, the model for general assignment problem is the most common as it may be adapted and applied for many scenarios. In healthcare decision making, this model may be used to determine which tools to use during surgery, how to assign patients to rooms, or the best treatment to prescribe for specified diagnoses. The latter will be explored in following sections.

Unfortunately, most assignment problems are not a simple as assigning one resource to one task for one cost. There are instances in which resources or tasks may require sequencing as seen in an assembly line. Resource may be incapable of completing certain tasks as is seen in a system with novice workers. It may also be the case that the cost of assigning a task is not constant; it may be a function of time, quality, or depend or other assignments. Whatever the case may be, the general assignment problem is commonly referenced due to its adaptability. A few adaptations of the assignment model follow.

2.2.1.2.3 Quadratic assignment problem. The quadratic assignment problem has been adapted to solve problems in many fields and applications including scheduling and location

problems (Carlson & Nemhauser, 1966; Geoffrion & Graves, 1976; Koopmans & Beckmann, 1957) as well as in healthcare engineering (Elshafei, 1977). Commander (2005) provides a detailed survey of various applications of the quadratic assignment problem. Similar to the general assignment problem, the quadratic assignment problem seeks to determine the most cost efficient way to assign resources to complete tasks. The key difference between the quadratic assignment problem and the general assignment problem is that its cost function is a second degree polynomial or quadratic because resources are not assigned independently (Koopmans & Beckmann, 1957).

The quadratic assignment problem assumes a distance associated with completing a task in addition to a cost. For example, consider a group of resources k and group of tasks l. Resource i belongs to group k and task j belongs to group l. The decision in the quadratic assignment problem is x_{ab}; whether object a of group b be assigned to an object from another group. This decision is based primarily on the cost c_{ij} of assigning resource i to task j and the distance d_{kl} between resource group k and task group l. The mathematical model for the 0-1 quadratic assignment problem as defined by Commander (2005) is as follows:

$$\min \sum_i \sum_j \sum_k \sum_l c_{ij} d_{kl} x_{ik} x_{jl} \tag{2.5}$$

$$\text{s.t. } \sum_i x_{ij} = 1 \; \forall \; j \tag{2.6}$$

$$\sum_j x_{ij} = 1 \; \forall \; i \tag{2.7}$$

$$x_{ij} = 0 \text{ or } 1 \; \forall \; i \text{ and } j \tag{2.8}$$

The objective function in the formulation shows that associated costs and distance values are incurred only when x_{ik} is assigned to x_{jl}. The objective function ensures that the low cost,

short distance strategy is chosen for assignments. Additionally, for the 0-1 assignment problem, there is a one-to-one relationship between tasks and resources secured by model constraints.

Sahni and Gonzalez (1976) proved that the quadratic assignment problem is NP-complete and also proved that ε-approximate solution for the problem is also NP-complete, making the quadratic assignment problem among the "hardest of the hard" of all combinatorial optimization problems. These characteristics suggest that finding optimality in polynomial time is unlikely (Finke, Burkard, & Rendl, 1987).

2.2.1.2.4 Weapon target assignment problem. The weapon target assignment problem is named such for its use in warfare (Eckler & Burr, 1972). This problem is concerned with assigning weapons to hit a set of targets so the expected survival value of a target, more commonly that of an enemy, is minimized (Ahuja, Kumar, Jha, & Orlin, 2003).

In this problem, the decision variable x_{ij} is the number of weapon type i to engage with target j. Assume there are W_i weapons of type i available. Let c_j be the value of target j. Let p_{ij} be the probability of target j surviving a hit from one of weapon type i. Then $p_{ij}^{x_{ij}}$ is the probability of target j surviving. The nonlinear integer programming model to solve the general assignment problem is formulated as follows:

$$\min \sum_j c_{ij} \prod_i p_{ij}^{x_{ij}} \tag{2.9}$$

$$\text{s.t. } \sum_j x_{ij} < W_i \ \forall \ i \tag{2.10}$$

$$x_{ij} \geq 0 \text{ and integer } \forall \ i \text{ and } j \tag{2.11}$$

The objective function for this model also ensures that the lowest expected value of survival is chosen for assignments. Equation 2.10 ensures that the assignment of weapons does not exceed the number of weapons available. Equation 2.1 ensures that a positive, whole number of weapons is assigned.

The weapon-target assignment problem becomes complex due to the nature of information required to develop the model and its implied dimensionality. Consider that there are two categories of data for this problem: target data and weapon data. The target data must provide accurate information about the worth, strength, and two dimensions (a third may be added for altitude) of location of each target. Only the location information is truly capable of being quantified. Secondly, weapon data must provide insight on the chances that weapons will hit a target and the gain of each hit.

Beyond data definition, much of the complexity of the weapon target assignment problem is due to the dimensionality of the problem. For example, let there be t targets of interest for a scenario that allows the use of p weapon types with n_i being the number of weapons of type i. This problem would require $4t$ items of target data and $t\sum_i n_i + p$ items of weapon data. Totally, $t(\sum_i n_i + 4) + p$ items of data would be required to solve this problem.

Day (1985) confirms the complexity of data requirements in an example where 600 individual targets, ten weapon types and 100 weapons of each type, result in a required 602,410 items of data to account for in assigning values and for the 3000 choice variables. Additionally, the model for the weapon target assignment problem is nonlinear, so the problem grows more unsolvable as dimensions are expanded. Targeting specialists commonly subdivided a set of enemy targets into complexes to circumvent the complexity of large scale target assignment problems. Concepts from the weapon target model are used in this research.

2.2.2 Solution representation. Three notions have been commonly applied to drive solution of multi-objective optimization problems: utility theory (Keeney and Raiffa, 1976.), weighted sum method (Kim and de Weck, 2006), and Pareto or efficient frontier optimization. Konak et al. (2006) justifies the use of efficient frontier optimization over other methods.

Figueira et al. (2005) provides a detailed discussion on this topic. As this research serves to inform policy and not suggest specific cost-efficacy solutions, the efficient frontier method is used for all models. An overview of efficient frontiers is provided below.

The concept of the efficient frontier was developed by economists to help select the most optimal investment portfolio, the most efficient being that with the highest expected return for a level of risk (Markowitz, 1959). Since inception, efficient frontiers have been a tool for defining operational efficiencies across industries. Hollingsworth (2003) provides a review of the wide use of efficient frontiers in private and public healthcare systems, including the Defense and Veterans' Administration hospitals.

Figure 2.4 is an adaptation of an efficient frontier for a healthcare system that defines operational efficiency as the highest expected value of efficacy for a given cost (Kerno, 2008). The frontier is the upper bound of the region of feasible operational strategies. The set of all points that compose this boundary is the efficient set. Any physician that operates according to an efficient point is utilizing best practices. In this diagram, physicians "A" and "B" are noted as utilizing best practices.

In the case of in individual physician decision, physician "C", the investment for current operations could be used for more effective processes as with physician "A" or the current level of efficacy could be reached at a lower cost as with physician "B". For this reason, physicians operating outside of the efficient set, such as physician "C", are termed inefficient. This information can be aggregated to produce system efficient frontiers.

Using efficiency frontiers, healthcare decision makers can better evaluate the cost structure required to meet a given level of efficacy. The efficiency frontier provides a range of optimality. Compared to using frameworks that output a single optimal solution for operation,

the efficiency frontier allows flexibility for decision makers to determine how to align operational goals with practices (Marler and Arora, 2004). Additionally, the relationships between optimal and suboptimal solutions are easy to visualize.

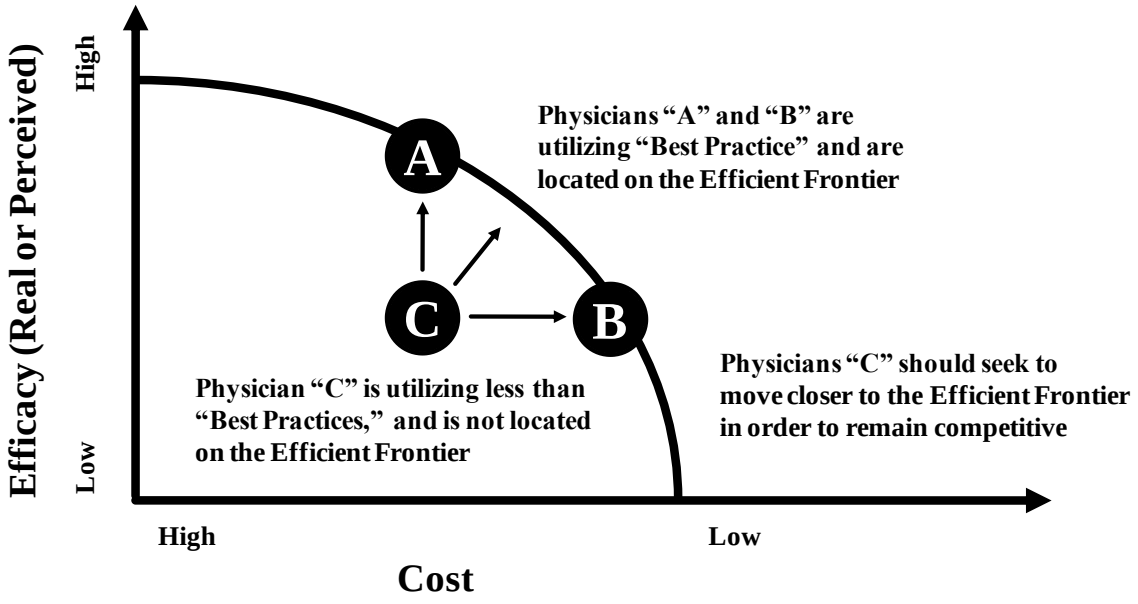

Figure 2.4. Efficient frontier for providers evaluating cost versus efficacy.

2.3 Policy Model Solution

For each model developed, a solution method(s) is created. The solution methods are used to produce efficient frontiers which yield further understanding between the relationships of "good solutions." A key assumption in the modeling and solution approaches is that the allocation is static (one-time). Issues of treatment impact over time are not included. Given the desired use of the model output, the value of including time is minimal relative to its cost.

2.3.1 Multi-objective solution. The generic form of the multi-objective optimization problem (MOOP) is commonly stated as follows:

$$\min\left[F_1(\mathbf{x}), F_2(\mathbf{x}), \ldots, F_k(\mathbf{x})\right]^T \tag{2.12}$$

$$\text{s.t. } g_j(\mathbf{x}) \le 0 \forall j = 1, 2, \ldots, m \tag{2.13}$$

$$h_i(\mathbf{x}) = 0 \forall i \tag{2.14}$$

Equation 2.12 represents *k* objective functions, equation 2.13 represents *m* inequality constraints, and equation 2.14 represents *e* equality constraints. The vector **x** is the set of decision or design variables.

Use of the MOOP stems from concepts in economics and mathematics such as equilibrium and game theories. Concepts have been adapted to solve various engineering design problems (Jendo et al. 1985; Psarras et al. 1990; Tseng and Lu 1990). Stadler (1988) provides an extensive historical account and tutorial for the application of these problems. For the problems in this research, multi-objective optimization is applied to manage the minimization of total costs and maximization of efficacies simultaneously.

2.3.2 Solution methods. A wide variety of techniques exist for producing solutions to optimization / search problems. A partial taxonomy of such methods is shown in Figure 2.5. These research seeks to develop a variety of such methods with the additional requirement of creating an efficient frontier rather than a single solution. Three general solution methods are used for one or more of the models.

1. *Complete Enumeration*. This approach identifies all different combinations of decision variables. Once evaluated, the set of solutions is processed to produce an efficient frontier. That frontier is the known optimal frontier for the model.
2. *Heuristic Search*. This approach uses problem specific information in an attempt to improve and broaden the perceived efficient frontier. Heuristic search is the fastest of the three solution methods.
3. *Genetic Algorithm*. Evolutionary algorithms are unique among the listed meta-heuristic methods because they maintain a set of solutions. That set might be improved and

broadened and then processed to produce an efficient solution. The evolutionary algorithm provides a balance of solution search breadth and speed and handles the final non-binary representation of decision variables.

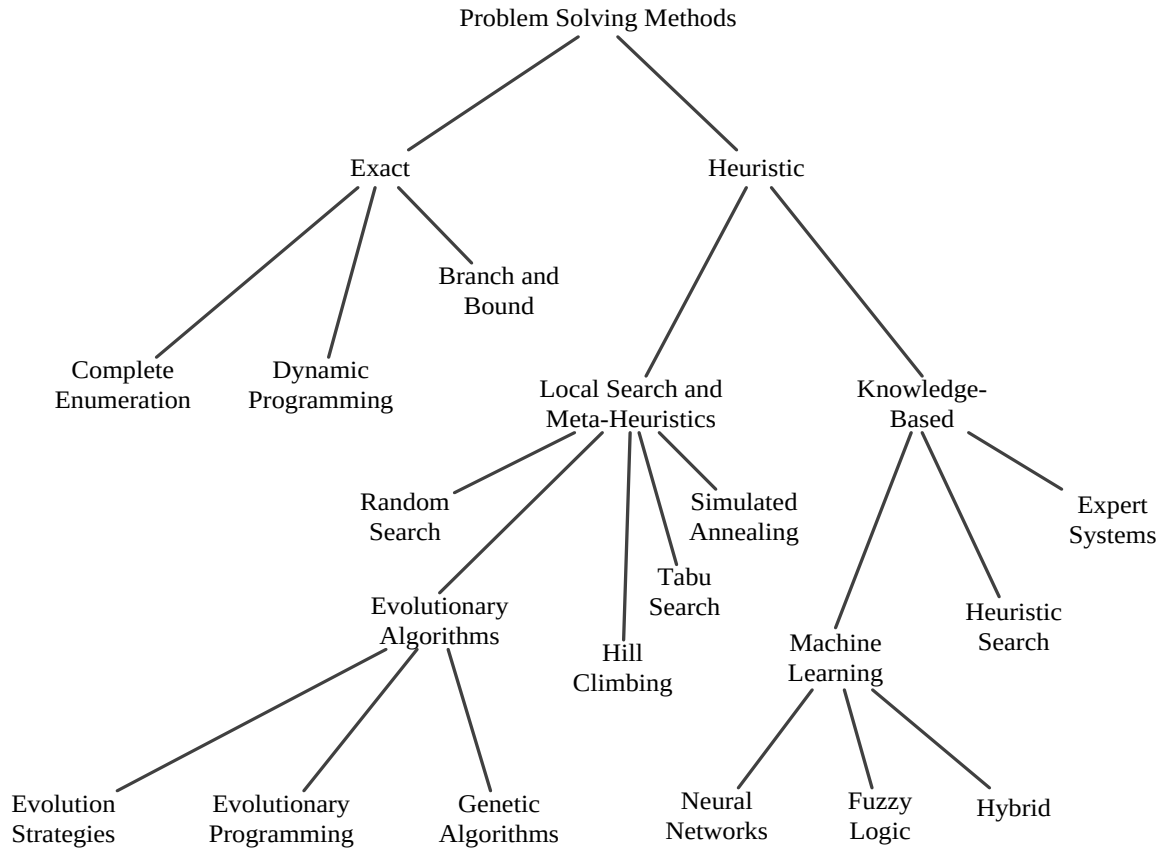

Figure 2.5. Problem solving methods.

However, due to its role of mimicking rational behavior under a given policy to find good solutions, the exact efficient frontier is not critical.

2.4 Integration Assessment

During the initial analysis of the segmentation challenge and associated integration opportunity, key environmental factors are identified. These factors form the basis for the experimental design constructed and executed in order to compare policies. Factor definitions,

levels, and variable generation processes are described in the chapters. Relevant factors considered in the dissertation include:

1. Variance in perceived treatment efficacy among treatment designers
2. Variance in fixed costs between treatments
3. Variance in variable costs between treatments
4. Variance in disease/patient condition severity
5. Variance in patient volumes / proportions
6. Opportunity for decreasing unit costs / volume discounts
7. Treatment efficacy as a function of dosage

2.4.1 Problem generation. Each of these seven factors is listed in the following table with the levels used in the design of experiments. The combination of factors and factor levels varies across policies. At most, when all factor levels are studied, total of $2^6 \times 4 = 256$ blocks of experiments are implemented. Each block of experiments consists of ten replications for all studies. This implies are maximum of 2560 observations.

Table 2.1

All possible experimental problem factors and levels.

Factor	Factor Description	Factor Levels
1	Variance in provider's perceived efficacy	1-High, 2-Low
2	Variance in fixed cost	1-High, 2-Low
3	Variance in variable cost coefficient	1-High, 2-Low
4	Variance in disease severity	1-High, 2-Low
5	Variance in provider's patient volumes	1-High, 2-Low
6	Degree of volume discounts	1-Concave, 2-Linear
7	Degree of efficacy curvature	1-Concave, 2-Linear, 3-Convex, 4-Random

For each form of integration examined in this research, an observation consist of the same set of generated values. These values are shared between the pre- and post- integration models and aggregated according to model formulation criteria. The manner in which the values

are shared, along with solution transformation (see section 2.5.1), ensures that solution comparisons are fair.

The levels for factors 6 and 7 are set to determine the curvature of costs functions and efficacy functions, respectively. It is assumed that costs may have a linear relationship with volume or a non-linear such that unit costs decrease as demands increase. Two levels are included for this factor in the design of experiments to study the impact of these two relationships. It is also assumed that efficacy may have a linear or non-linear relationships with treatment volumes. Three levels are included in the design of experiments to study the impact of three relationships. The first relationship, factor level 1, assumes that efficacies initially have rapid growth utilization increases and eventually, the growth diminishes until virtually no gains are possible. The second relationship, factor level 2, assumes that the growth in efficacies is constant for each added unit of utilization. The third relationship, factor level 3, assumes that efficacies initially have slow growth utilization increases and eventually, the growth increases rapidly until virtually no gains are possible.

2.4.2 Random variable generation. The variables for factors 1 through 5 must be generated to fit within a given interval and maintain a given level of variance. A beta distribution Beta(α, β) is used to generate these random variables because its parameters can implicate the mean, variance, and skewness of random variables on intervals of finite length. The beta distribution is a continuous probability distribution defined on interval [0, 1]. Its parameters, α and β dictate the shape of the distribution. The probability distribution function of the beta distribution for $0 \leq x \leq 1$ and the parameters α and β is as follows:

$$f(x;\alpha,\beta) = \frac{1}{\mathrm{B}(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1} \tag{2.15}$$

In equation 2.15, B is a normalization function to ensure that the total probability integrates to unity. The mean $\mu(X)$ and variance $\sigma^2(X)$ of a beta distribution random variable X with parameters α and β are explained algebraically in equations 2.16 and 2.17, respectively.

$$\mu(X) = \frac{\alpha}{\alpha + \beta} \tag{2.16}$$

$$\sigma^2(X) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{2.17}$$

Notice that when $\alpha = \beta$, the mean $\mu(X) = 0.5$ and is at the center of the distribution. In the event that $\alpha \neq \beta$, the distribution is skewed. Based on these relationship between the distribution parameters and the mean, if a factor of the experimental design requires values that are low, centered, or high, the parameters of the distribution will be set such that $\alpha < \beta$, $\alpha = \beta$, or $\alpha > \beta$, respectively. As an example, Figure 2.6 illustrates relationship to the center for three different beta distributions where $\alpha = 2, \beta = 5$; $\alpha = 2, \beta = 2$; and $\alpha = 5, \beta = 2$.

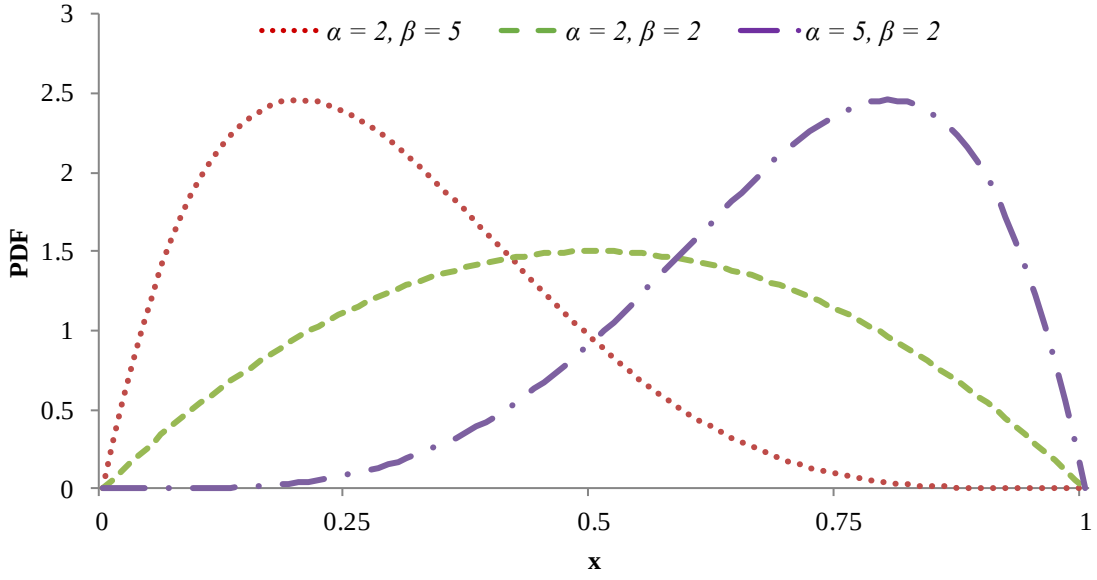

Figure 2.6. Example PDFs from beta distribution.

Figure 2.6 also illustrates how variance increases as the parameters for the beta distribution approach zero. This can be proved by taking the limit as α and β approach zero of equation 2.17. In general, variance increases as α and β are closer in value and/or closer to zero. An exceptional feature of the beta distribution is that $\alpha = \beta = 1$. Therefore, if an experimental design requires high variance between randomly generated variables, $\alpha = \beta = 1$. If an experimental design requires low variance between randomly generated variables, $\alpha = \beta = 6$. The latter structures a beta distribution that is closer to a normal distribution that is truncated to fit within a finite interval whereas the normal distribution lies on an infinite interval and, therefore, is not reasonable the variables used in this research.

When a high level of variance is required for a variable in an experiment, a beta distribution of high variance is used to generate random numbers and an accept/reject algorithm is used to ensure that a sample variance no less than the population variance. Likewise, when a low level of variance is required for a variable in an experiment, a beta distribution of low variance is used to generate random numbers and an accept/reject algorithm is used to ensure that a sample variance is no higher than the population variance. For example, the following is an example of an accept/reject algorithm to generate random number set R having high variance.

Step 1: Generate $x_1, x_2, \ldots, x_n$ i.i.d. from Beta(1,1)

Step 2: Compute the variance of the generated $x_1, x_2, \ldots, x_n$ as S^2

Step 3: If $S^2 < \sigma^2$, return to step 1. Otherwise, R = $x_1, x_2, \ldots, x_n$.

Now that the mean, variance, and skewness of randomly generated variables of the experiments have been explained, the method to ensure a given interval for a random variable is provided. Given constants a and b, expected value E(X), and variance V(X), the random variables are translated that fundamental properties are maintained as follows:

$$E\left[aX+b\right]=aE(\mathrm{X})+\mathrm{b} \tag{2.17}$$

$$V\left[aX+b\right]=a^2\,\mathrm{V}(\mathrm{X}) \tag{2.18}$$

Recall the accept/reject algorithm and parameters for the beta distributions used in this research. The values for α and β (i.e. $\alpha = \beta = 1$ and $\alpha = \beta = 6$) dictate that low variance corresponds to $\mathrm{Var(X)} \leq 0.0192$ and high variance corresponds to $\mathrm{Var(X)} \geq 0.0833$. Consider a simple example for generating fixed cost d_i on an interval of [0, 100]. Matlab is used to generate instances of low variance and high variance of fixed cost between three treatments. The instance of low variance in fixed cost between treatments results in $d_i = 61.50$, 48.79, and 53.42 which has a variance of $41.38 \leq 100^2 \times 0.0192$. The instance of high variance in fixed cost between treatment results in $d_i = 78.31$, 18.04, and 9.87 which has a variance of $1397.10 \geq 100^2 \times 0.0833$.

2.5 Integration Analysis

Experimental output from the prior step is compared and analyzed in order to determine the characteristics of high leverage integration opportunities. This comparison consists of three steps: solution transformation, solution comparison, and analysis of results.

2.5.1 Solution transformation. First, the solutions generated by both policies may need transformation to a form for consistent evaluation. Evaluation is based on the more realistic assessment of cost and efficacy given the form of integration under consideration. Specific transformation details are described in the relevant chapters. In general, the efficient solutions for a pre-integration model are transformed so that they may be evaluated under the objective functions of the post-integration model formulation. This implies that solutions from the pre-integration model cannot dominate solutions of the post-integration model as the transformed solutions must fit within the post-integration model's feasible solution set. The transformation allows for fair comparison of the pre- and post- integration policies.

2.5.2 Solution comparison. Second, efficient frontiers for the same experiment produced by the two policies are compared. These efficient frontier analyses are used to compare the performance of the problems generated and help characterize some properties of optimal treatment protocols. Three measures were used to assess the relative performance of the models: average improvement in efficacy, average improvement in cost, and average error rate. The average improvement in efficacy and average improvement in cost are based on the "area of opportunity" between the frontier of the pre- and post-integration policies.

This study employs a novel efficient frontier comparison method. Prior to developing this analysis method a few existing metrics are explored. Namely, the methods employed by Veldhuizen (1999), Veldhuizen & Lamont (1998), Zitzler & Thiele (1999), Srinivas & Deb (1994). These strategies are described in Table 2.1. For example, Veldhuizen (1999) defines error rate as the percentage of solutions from the pre-integration policy that are not on (inferior to) the post-integration policy. Whereas, Veldhuizen & Lamont (1998) defines the general distance between two problems as the average Euclidean distance between the closest members from the pre-integration policy and the post-integration policy.

Table 2.2

Classic comparative metrics for multi-objective optimization problems.

Reference	Metric	Description
Veldhuizen (1999)	Error rate	Percentage of solutions from EF_{pre} that are not members of EF_{post}
Veldhuizen & Lamont (1998)	General distance	Average Euclidean distance from EF_{pre} to closest members of EF_{post}
Zitzler & Thiele (1999)	Space covered	Size of the global dominated set in objective space
Srinivas & Deb (1994)	Spread	Distribution of individuals in EF_{pre} over the non-dominated region.

For some experiments in this research, the global efficient frontier EF_{post} is known through enumeration approaches. The efficient frontier EF_{pre} is compared against EF_{post} to test the performance based on measures of error. To improve on existing strategies, the comparison used in this study allows for fair, scalable comparison under varying factors. The "area of opportunity" is a practical measure used for error assessment. Figures 2.6 and 2.7 illustrate the area for opportunity for efficacy and cost as A_2x and A_2y, respectively. These areas of opportunity are calculated by integrating the difference between EF_{post} and EF_{pre} for ranges intersecting costs or efficacy. Once the area of opportunity is determined, the average percent efficacy and cost improvement of transitioning from EF_{pre} to EF_{post} is calculated as $A_2x/A_1x \times 100\%$ and $A_2y/A_1y \times 100\%$, respectively. A high percent increase indicates poor performance of the solutions for EF_{pre} in comparison to the solutions for EF_{post}.

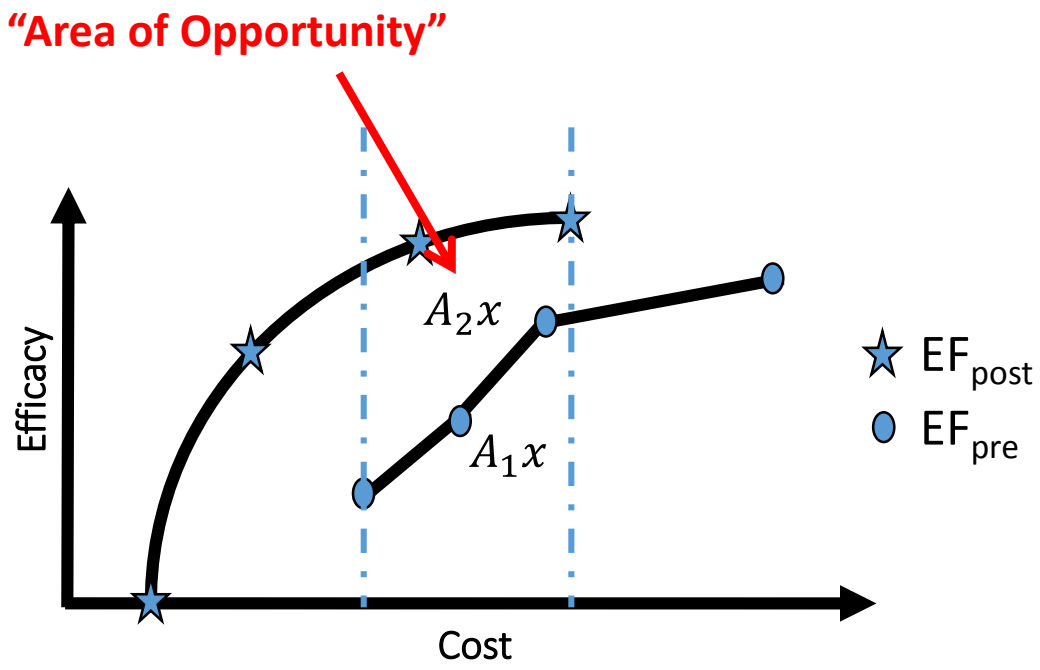

Figure 2.7. Regions of integration to assess efficacy improvement given cost range.

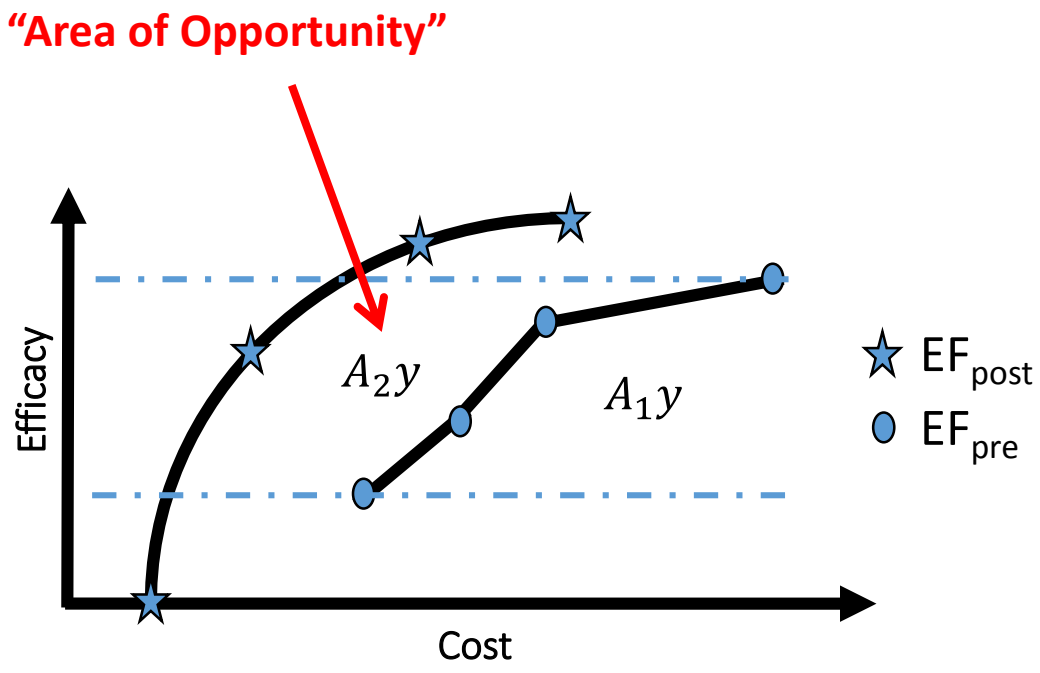

Figure 2.8. Regions of integration to assess cost improvement given efficacy range.

In instances where the ranges of cost (or efficacy) for two compared models do not intersect, the area of opportunity for efficacy (or cost) is classified as undefined. This provides two additional metrics which can be reported.

1. *Efficacy undefined*. The entire pre-integration frontier lies to the right of the post-integration frontier.
2. *Cost undefined*. The entire pre-integration frontier lies to the below of the post-integration frontier.

Both improvement metrics are presented as percentage and are easily interpreted.

2.5.3 Analysis of results. The third step of solution analysis involved the comparison of pre- and post-integration policies, both holistically and according to experimental factors. The five metrics are reported for each dimension of comparison. Underlying reasons for the results are listed and unexpected results explained.

2.6 Integration Implications

Finally, any policy suggestions that result from the research study are given along with a summative description of that integration analysis. Chapters 3 to 5 follow addressing efficacy integration and visibility; supply chain cost structure; and whole patient integration respectively.

CHAPTER 3

The Impact of Efficacy Visibility

This chapter addresses the strategic implications of providing a system view of treatment efficacy in a healthcare system where minimized cost and maximized efficacy are desired. Two quantitative policy models are created to imitate pre- and post- integration and compared using an experimental design assessing the performance impact of provider perspective variance, varying costs, and demand volumes. System efficacy visibility is shown to help mitigate cost and improve efficacy. Recommendations are made to leverage provider expertise and integrate systems for shared treatment decision making.

3.1 Segmentation Challenge and Integration Opportunity

3.1.1 Background. Treatment design might be generalized as providing a high efficacy treatment at a low cost. The understanding of the designer of treatment efficacy and cost is critical for healthcare quality and supply costs. The clarity of design might be negatively influenced by lack of visibility in terms of efficacy and cost, as well as issues of medical liability and associated cost apathy, limited designer experience, and pharmaceutical/medical industry marketing. A summary of efficacy and cost visibility follows.

Deviations in efficacy evaluation for an individual physician (or healthcare system) from a comprehensive system view, is termed treatment bias. The existence of such bias, even when unintentional on the part of the physician, can undermine system performance. Clinician bias in patient treatment has been studied for years; mostly to help interpret variance in patient outcomes. There is minimal study regarding the impact of clinician bias on supply cost of large scale healthcare systems. Isbister et al. (2007) recommend that more emphasis be placed on the issues of clinicians and patients rather than the vendor side of healthcare supply chain

management. Clinician bias might also be institutional in the form of high intra-organization variance in treatment protocol.

The idea of cost awareness (or lack of) among clinicians was presented as early as 1993 by Blum and Miller. The Blum and Miller (1993) research concluded that a large majority of observed physicians could not accurately estimate treatment costs. Several other studies followed with similar results. In 2000, Ernst et al. reported that a majority of physicians tend to underestimate the cost of expensive treatments and overestimate the cost of less expensive treatments. Reichert, Simon, and Halm (2000) concluded that most physicians have inadequate access to drug cost data but stated that cost was an important factor. McGuire et al. (2009) argues that cost should be of equal, if not more important, consideration in cases where treatments are cost prohibitive. So why aren't treatment costs weighed more heavily in physician decision making? It may be because cost information is not as readily available during treatment decision processes (Walzak et al., 1994; Fortuna et al., 2008; and Tseng et al., 2006). This is due in large part by procurement and charging policies.

In addition to provider awareness, patient cost awareness also plays a role in cost containment for healthcare and supply chain management. Many patients are concerned about the cost of healthcare but few are fully informed about the costs included in healthcare expenditures. It is also noted that patients are not likely to voice their cost concerns with providers while treatment decisions are being made. Providers and patients often have the desire to discuss out-of-pocket costs but can be dissuaded by perceived time constraints or because they believe clinicians cannot offer solutions to out-of-pocket cost (Bovier, 2005).

In addition to enabling patient-clinician discussions for cost, public awareness campaigns may also aid in reducing treatment costs. Studies show that affective public education

campaigns can decrease prescription cost by almost $3 billion per year (Donohue et al., 2008). Unfortunately, effectively public education campaigns are rare.

Instead of relying on uninformed clinician and patient decisions, decision support systems can be integrated to improve awareness, decrease costs, and improve the quality of care. Simple methods can be used to improve cost awareness and containment in healthcare (Roth et al., 2001). Providing cost "cheat sheets" can lead to better awareness and less expensive treatment decisions. Korn, Reichert, Simon, and Halm (2003) illustrate that information pamphlets and basic cost education interventions improve physician knowledge and willingness to consider costs in prescribing. The British Columbia has enacted healthcare cost containment policies implementing generic substitution and a reference drug program. Polinski et al. (2008) survey B.C. providers to determine that a majority believes that these policies are economically and clinically appropriate. However, the study infers the government legislation has more impact on acceptance than standalone physician training programs and controversy will ensue. Andersson et al. (2009) conduct an extensive study of Swedish physicians and witness stronger adherence to prescribing indicators over three years when decentralized budgets are presented to providers, with greater buy-in from clinics in the public sector and from younger, residency-trained physicians. Sweden has also established a medication guide book that has been widely accepted by physicians to inform medical decision making (Axelsson, 2008).

More advanced and computerized decision support systems also have a positive effect on prescribing behaviors. In a clinical trial, Fortuna et al. (2009) experience 88% of clinicians endorsing a medication alert system with 70% reporting that the system does not interfere with work flow across 14 provider sites.

Based on results from literature, it can be concluded that cost education programs and decision support systems allow physicians to make more economic treatment decisions. Additionally, more consolidated or standardized processes for treatment formulary lead to lower system costs. Although, previous researchers have concluded that clinician awareness of costs is an important factor in patient treatments, they have not modeled or conducted detailed quantitative analyses on the impact of awareness. Additionally, they have not considered strategies for obtaining or maintaining efficient treatment policies based on the trade-off between cost and efficacy as established in this research.

3.1.2 Characterization. This chapter focuses on the integration opportunity for efficacy visibility for treatment designers. Quantitative measures for treatment efficacy are numerous. A common example is the disability-adjusted life year (DALY). This research keeps the measure general due to its policy emphasis.

3.1.2.1 Patient volumes and treatment efficacies. Given a set of treatment designers and their associated perceived treatment efficacy, a system view of efficacy needs to be derived. Large healthcare organizations, such as the Veterans Affairs Health Administration, implement incentive policies that encourage patient treatment based on expertise. Providers with expertise in managing a given diagnosis are given incentives for serving high volumes of patients with the diagnosis. Providers that serve low volumes of patients for given diagnoses are encouraged to refer patients to providers with greater expertise. It is assumed that such a system will experience treatment efficacies that are mostly impacted by such experts. Thus, system efficacy is computed so that treatment efficacies are based on an average of volume-weighted provider efficacies.

3.1.2.2 Patient treatment unit costs. Several studies have concluded that providers generally lack awareness of treatment cost. This lack of knowledge is assumed to increase healthcare costs and, in turn, decrease the overall quality of care. Three views considered to incorporate cost are as follows:

1. Designer is cost indifferent, seeking just to maximize efficacy.
2. Designer is cost considerate, trying to choose lower cost options for high similar levels of efficacy.
3. Designer is cost educated, with some understanding of product price and its associated variable and fixed costs.

In this study, models assume the third view in the form of a unit treatment cost, independent of treatment volumes, or other procurement factors. This is the most generous view for designers and most conservative in terms of making conclusions. Details of cost structures are considered in Chapter 3. Basic characteristics of treatment cost variance are assessed to aid solution strategies for more complex costing structures in future research studies.

3.2 Policy Model Formulation

Two policy models are developed. The first models a pre-integration policy of treatment design based on individual efficacy perceptions, unit cost consideration, and disease specific / specialization (IUD model). The second models a post-integration policy of treatment design based on system efficacy, unit cost consideration, and disease specific / specialization (SUD model).

3.2.1 IUD model (pre-provider integration). For the IUD, treatment efficacy values are decided on a more independent basis by individual providers as opposed to a system with more shared decision making. Supply costs are considered per unit. As previously mentioned,

the cost factor assumed in this problem does not consider leveraging economies of scale from combined treatment selection volumes or discount cost. Entry costs are not legitimately accounted for either. The patients in this problem are viewed according to the providers' specialties or per diagnosis as opposed to a system with capability to view and treatment the whole of a patient's complications. The main goal of treatment selection decision makers at the clinical theatre level is to find the optimal values of the decision variables x_{ijk}: the binary decision of provider k to select treatment i for diagnosis j to maximize the total perceived value of efficacy while minimizing total cost.

The efficacy coefficient for this problem is $\tilde{e}_{ijk}$, provider k's perceived efficacy value of treatment i when applied to diagnosis j. The cost factor for treatment i is c_i, the perceived unit cost which is the sum of fixed cost d_i and variable cost s_i.

The optimization problem of selecting the x_{ijk} decision variables are formally expressed in the IUD model as follows:

$$\max \sum_i \sum_j \tilde{e}_{ijk} x_{ijk} \tag{3.1}$$

$$\min \sum_i \sum_j c_i x_{ijk} \tag{3.2}$$

$$\text{s.t. } \sum_i x_{ijk} = 1 \ \ \forall \ \ j \text{ and } k \tag{3.3}$$

$$x_{ijk} = 0 \text{ or } 1 \ \ \forall \ \ i, j, \text{ and } k \tag{3.4}$$

Note that this model emphasizes the disease and designer independence of the decision-making process. This approach has the physicians assign a single treatment to each diagnosis with binary representation. Patient volume is not considered. Recall that a treatment might be a combination of proactive and reactive or a "do nothing" option. Unit cost information is considered visible by the designer (which is generous based on the research indicating much cost indifference). An alternative model might be including some representation of cost utility and

visibility. Including this would further promote the attractiveness of integration. A key modeling desire was to have a simple model of the most segmented policy in order to accommodate the increasing complexity of modeling the integration impact.

3.2.2 SUD model (post-provider integration). With a system view of efficacy (given a system view of unit cost), then treatment decisions are made on a system level. The decision variable x_{ijk} is replaced by x_{ij}, the binary decision of for the system to select treatment i for diagnosis j. The efficacy coefficient for this problem is $\tilde{e}_{ij}$, the system assessment of efficacy for treatment i when applied to diagnosis j. Note that this research estimates $\tilde{e}_{ij} = \left(\sum_k V_{jk}\tilde{e}_{ij}\right)/\left(\sum_k V_{jk}\right)$ where V_{jk} is volume of patient with disease j treated by physician k. Again, this estimate is conservative in terms of revealing the impact of integration. The optimization problem of selecting the x_{ij} decision variables are formally expressed in SUD model as follows:

$$\max \sum_i \sum_j \tilde{e}_{ij} x_{ij} \tag{3.6}$$

$$\min \sum_i \sum_j c_i x_{ij} \tag{3.7}$$

$$\text{s.t. } \sum_i x_{ij} = 1 \ \ \forall \ \ j \tag{3.8}$$

$$x_{ij} = 0 \text{ or } 1 \ \ \forall \ \ i \text{ and } j \tag{3.9}$$

Note that this model emphasizes the disease independence of the decision-making process. This approach has the system assign a single treatment to each diagnosis with binary representation. Patient volume is not considered. Unit cost information is considered visible by the designer (which is generous based on the research indicating much cost indifference).

3.3 Policy Model Solution

For this research, problems are sized and generated to enable solution through enumeration. This approach is taken so that the entire set of feasible solutions might be compared to the efficient frontier as in Figure 3.1. In practice when seeking the efficient frontier, treatments might be ordered by cost or by efficacy. Treatments which are cost-efficacy dominated might be eliminated on a disease by disease basis. Only disease cost efficient solutions need be considered when constructing the system efficient frontier.

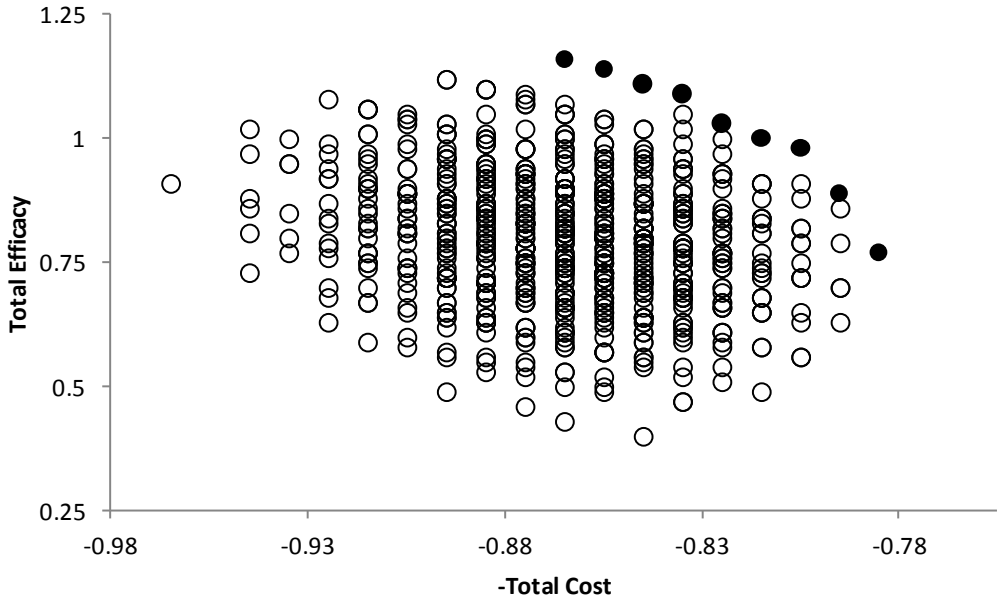

Figure 3.1. Feasible solutions with highlighted efficient frontier.

3.4 Integration Assessment

In the event that your figure caption is longer than one line, be sure that the subsequent line is flushed to the left margin. See figure 2 for visual confirmation.

3.4.1 Selected factors. Based on the characterization in Section 3.1, the following factors are considered in developing a design of experiments.

3.4.1.1 Patient volume factors. For each problem, the demand volume of patients with disease j serviced by each provider k, V_{lk}, must be known. The variability in V_{lk} is one factor considered in the model. High variability indicates presence of treatment design by both experts and inexperienced providers. Low variability indicates that patient volumes are balanced across designers.

3.4.1.2 Efficacy factors. The parameters used to evaluate efficacy values in this study are e_{ijk}, and V_{lk}. The basis for evaluating total efficacy between problems is that the perceived efficacy may differ between providers and true efficacy may be estimated based on provider expertise. Values of efficacy are normalized to range from 0 to 1 with 0 being ineffective and 1 being completely effective. Given that provider k's perceived efficacy of treatment i for diagnosis j is e_{ijk}, the system calculated efficacy according to $\tilde{e}_{ij} = \left(\sum_k V_{jk}\tilde{e}_{ij}\right)/\left(\sum_k V_{jk}\right)$. The variability in perceived efficacy is a factor in the design of experiments. High variability indicates many different perceptions regarding treatment efficacy. Low variability indicates some consensus in terms of efficacy.

3.4.1.3 Cost factors. The parameters used to generate cost values in this research are d_i and a_i. The parameters for fixed costs and variable cost coefficients are varied from 0 to 100 and then added to form a unit cost. It is assumed that the variable cost of treatment i tends to be directly related to the relative efficacy of treatment i. Understanding that not all variable cost are correlated to treatment efficacies, a "noise" factor is added such that the variable cost of treatment i is $s_i = a_i(\sum_j \tilde{e}_{ij})/(\sum_j \tilde{e}_{ij}) \pm \sigma_i$. The cost of selecting treatment i is calculated as $d_i + s_i$. Variability in fixed costs and variability in variable costs are factors in the design of experiments.

3.4.2 Design of experiments. The objective of the experimental design is to understand the relationship between parameters (variance in provider's patient volumes, variance in

provider's perceived efficacy, variance in fixed cost, and variance in variable cost coefficient) on the performance of treatment selection protocols of the IUD in comparison to the SUD. Parameters are set in a manner that allows for inferences to be made relative to a system progressing through the four policies of treatment protocols selection.

3.4.2.1 Generation of problem instances. In this study, a problem with three treatments, three providers, and three diagnoses is examined. These parameters are generated using custom programs in Matlab 7.12.0 (R2011a). This section discusses the design of experiments for comparing solutions of the two formulated models. First, the methods used to generate their values for the design of experiments are provided.

3.4.2.2 Random value generation. The values for d_i, e_{ijk}, a_i, and V_{lk}, are randomly generated from beta distributions. The parameters are translated using fundamental properties of expected value and variance so that the expected ranges and variances for problem parameters are maintained. For parameters d_i, e_{ijk}, a_i, and V_{lk}, low variance corresponds to $\text{Var}(X) \leq 0.0192$ and high variance corresponds to $\text{Var}(X) \geq 0.0833$. Accept and reject logic is used to ensure that the desired level of variance is maintained in the generation of the four factors with different levels of variance. The purpose of using the beta distribution is to ensure that values lie within a desired range (avoiding the disadvantages of truncation), maintain the desired level of variance, and may be skewed from the central value of parameter's range (Fox, 1963).

The Matlab code for each of the random number generators is found in the APPENDIX. The problem parameters varied as a part of this research are summarized in Table 3.1. Ten replications are executed for each factor level combination in the design for a total of 160 observations.

Table 3.1

Study 1 experimental problem factors and levels.

Factor	Factor Description	Factor Levels
1	Variance in Perceived Efficacy between Providers	1-High, 2-Low
2	Variance in Patient Volumes between Providers	1-High, 2-Low
3	Variance in Fixed Costs between Treatments	1-High, 2-Low
4	Variance in Variable Costs between Treatments	1-High, 2-Low

3.5 Integration Analysis

Based on the design of experiments, a total of 160 observations were used to assess model performances. Enumeration was used to find optimal efficient frontiers. Efficient solutions of the IUD and SUD models are evaluated using the system efficacy. Note that given the model solutions, only the decision variables, unit cost, and system efficacy are used in this evaluation. Patient volume, supply chain cost structure, and efficacy utility are not considered (though these factors are considered in subsequent chapters).

3.5.1 IUD-SUD Comparison. Table 3.2 lists the performance results of the IUD relative to the SUD by experimental factor. Note that in this study, there were no instances of undefined solutions (either efficacy or cost). As expected, the SUD outperforms the IUD across all performance measures. Also, the IUD tends to perform closer to the SUD when there is low variance in problem factors. Overall, the average improvement in cost, improvement in efficacy, and error rate are 10%, 5%, and 42%, respectively. The following sections provide details of performance based on each experimental factor.

3.5.1.1 Provider variances. The average improvement in efficacy is a consistent 5% for high and low levels of variance in perceived efficacy. It is expected that this result occurs as variance in perceived efficacy between providers can be mitigated through system integration

and share treatment protocol selections. However, the corresponding cost of solutions made based on perceived efficacies results in decreased performance. This decreased performance is amplified as the variance in perceived efficacy between providers is increased. According to the average changes in cost and error rate, the IUD performs better when the variance in perceived efficacy between providers is classified as low. The average change in cost is increased from 6% to 12% and the average error rate is increased slightly from 41% to 43% as variance in this factor is changed from low to high.

Table 3.2

Average improvement of the SUD relative to the IUD.

Factor Levels	Improvement in Cost	Improvement in Efficacy	Average Error Rate
Variance in Perceived Efficacy between Providers			
High	12%	5%	43%
Low	6%	5%	41%
Variance in Patient Volumes between Providers			
High	11%	6%	42%
Low	6%	5%	43%
Variance in Fixed Costs between Treatments			
High	11%	8%	51%
Low	7%	2%	33%
Variance in Variable Costs between Treatments			
High	9%	6%	42%
Low	8%	5%	43%

3.5.1.2 Patient volumes. Patient volumes were also used in the formulation of total cost and total efficacy values. Accordingly, this factor does show an impact on performance that is similar to the trends in variance in perceive efficacy between providers. The average change in cost is increased from 6% to 11% as variance in patient volumes between providers is increased.

The average change in efficacy is relatively unchanged, increasing from 5% to 6%. The average error rate is decreased slightly from 43% to 42% as the variance in this factor is changed from high to low.

3.5.1.3 Fixed costs. All performance values are increased as the variance in fixed cost between treatments is increased. The average change in cost is increased from 7% to 11%, the average change in efficacy is increased from 2% to 8%, and the average error rate is increased from 33% to 51% as variance in this factor is changed from low to high. Improvements due to this factor provide the most support for providers to share treatment selection protocols. Over all factor levels, the lowest improvement in efficacy (2%) occurs when the variance in fixed cost between treatments is classified as low. Efficacy is not directly implied in the calculation of fixed costs as in the case of variable cost (refer to cost factors). Fixed costs are random or unbiased to treatment efficacies. Without making decisions for treatment selection in lieu of fixed cost, performance in decreased.

3.5.1.4 Variable costs. Unlike fixed costs, variable costs are assumed to be related to treatment efficacies in this research. Therefore, if the efficacy of a treatment increases, the variable cost of the treatment is also expected to increase. The efficient frontiers for the IUD and SUD will make proportionate shifts. Study results show relatively little change for all performance measures as levels for variance in variable costs between treatments change. The average improvement in cost changes from 8% to 9%, improvement in efficacy changes from 5% to 6%, and error rate changes from 43% to 42% as the variance in this factor is changed from low to high.

3.6 Integration Impact

3.6.1 Solution characterization. All solutions that satisfy the inequalities of the SUD model are basic feasible solutions. These solutions can be elements of the efficient frontier or inefficient solutions. Recall that the design of experiments employed three treatments and three diagnosis. Therefore, 27 basic feasible solutions were generated for each of the 160 observations of the SUD. This advantage was exploited to explore characteristics of basic feasible solutions that are more difficult to observe in the following studies. The solution sets of models in the following chapters will have more complex dimensions and contain non-integer elements making these models exponentially more difficult to solve. Before developing the studies in chapters 3 and 4, key characteristics were observed to provide insight for solution heuristics.

3.6.1.1 Frontier navigation. Given a solution on the efficient frontier, one need only consider changes in the treatment mix that increase efficacy in moving toward a higher efficacy solution (and vice versa for lowering cost). A heuristic that proved effective was a one treatment swap. The treatment with the highest change in efficacy relative to change in cost is attractive when a higher efficacy point on the frontier is desired. The treatment with the highest change in cost relative to change in efficacy is attractive when a lower cost point on the frontier is desired. When using such a one swap heuristic, one can check for any options where the incremental impact in favorable in terms of efficacy and cost (in case the swap has taken the procedure away from the efficient frontier).

3.6.1.2 Concave efficient frontier. Occasionally, two points on the efficient frontier might be endpoints on a line segment that dominates another point on the efficient frontier. The concave efficient frontier eliminates all such points. That line might represent a gradual shift

between the solutions associated with the endpoints. That shift could lead to better performance than with the dominated efficient frontier point. This concept is used in future chapters.

3.6.1.3 High efficacy/low cost regions. An interesting trend was noticed when adjusting efficacy variances or cost variances. In particular, as the variances between treatment efficacies increase, the frontier solutions within regions of high efficacy tend to remain as frontier solutions in the high efficacy region as other parameters remain unchanged. As the variances between treatment unit costs increase, the frontier solutions within regions of low cost tend to remain as frontier solutions in the low cost region as other parameters remain unchanged.

Figure 3.2 illustrates the change in frontier solution locations as variances in treatment cost and efficacy are increased. Frontier solutions in the high efficacy or low costs regions are highlighted. In order for provider "C" to perform as well as provider "A" or "B", provider "C" must increase total efficacy without sacrificing cost or decrease total cost without sacrificing efficacy. A similar concept was exhibited when exploring the basic feasible solutions of the SUD. When moving from the more inefficient solutions toward solutions of the efficient frontier, it is shown that there are as many as three paths to improve the performance inefficient solution toward the efficient frontier:

1. Decrease costs at a faster rate than decreasing efficacies.
2. Increase efficacy while decreasing costs.
3. Increase costs and increase efficacy.

Referring to Figure 2.4, these types of transitions can lead to efficient solutions to the right of provider "B", efficient solutions between providers "A" and "B", or efficient solutions to the left of provider "A". Thus heuristic solutions much focus on the breadth and quality of the produced efficient frontier.

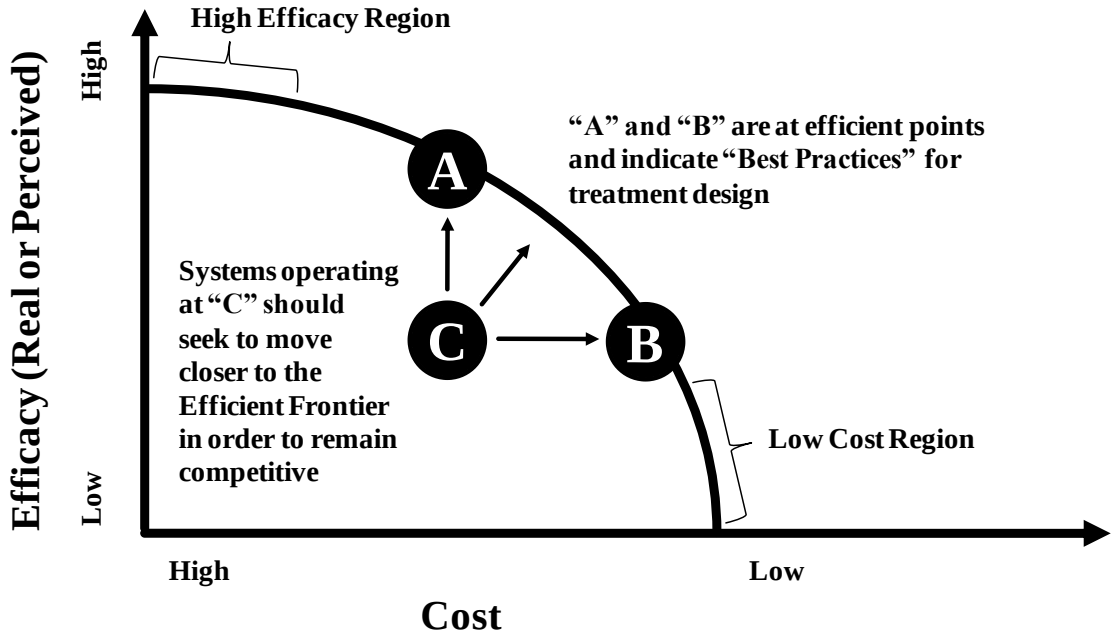

Figure 3.2. High efficacy and low cost regions along the efficient frontier.

3.7 Chapter Summary

The study discussed in this chapter models and assesses provider integration for healthcare supply chain management. The study examined considers the implications of information visibility in informing treatment efficacy. A unique performance assessment method is proposed so that system performance may be evaluated in a more practical manner. Using this assessment method, it is determined that informing provider decisions in treatment selection protocols has a positive impact on overall system cost and efficacy. Subsequent studies will explore the intricacies of treatment interactions, patient severity levels, and complex treatment costing structures. The observed system characteristics are further explored in chapters 3 and 4 to develop solution techniques for more complex models.

CHAPTER 4

The Impact of Supply Chain Visibility

The primary goal of this study is to determine the strategic implications of considering supply chain cost structure in creating treatment designs. A full design of experiments is conducted to determine the performance impact that order-volume discounts, provider variance, varying costs, and demand volumes have on treatment selection protocols. A heuristic solution technique is employed to help comprehend processes involved in efficient treatment protocol selection. Recommendations are made to integrate treatment selection protocols, improve cost awareness among providers, and leverage economies of scale through shared and standardized procurement and allocation policies.

4.1 Segmentation Challenge and Integration Opportunity

A substantial benefit of new healthcare enterprise integration systems is that they may be used to exploit and improve supply chain logistics. Best practices can be observed and used to improve procurement policies. Here, the impact of treatment selection protocols in an integrating healthcare supply chain is explored.

4.1.1 Background. To date, healthcare supply chain operations account for 30 to 40 percent of healthcare cost, second only to personnel costs ("The Integration of Innovation and Clinical Need: ROI - the Mercy Supply Chain Story," 2002). Gupta and Orbe (2009) discuss a report from Arizona State University, labeling United States healthcare inefficient and expensive as providers spend 31 to 67 percent of operating budgets on supply chain processes. Since costs are assumed by patients, physicians are less concerned with cost and equity when prescribing treatment designs. Most hospitals allocate patients' fees according to charges made in perioperative services (i.e. the period around patient operations), pharmaceuticals, and materials.

Likewise, their view of supplies is distributed into these three categories of patient fees. However, in a more transparent supply chain, hospitals have more access to the details in supply costs and are better able to pinpoint and mitigate costs attributed from 3PL, wholesalers, and consignment (Darling & Wise, 2010). Costs from shrinkage and stock-outs are also better understood and easy to avoid.

A few researchers have investigated the impact of consolidating and standardizing treatment selection decisions of clinicians in relation to supply chain management. Kelle et al. (2012) address the conflict of formulary or product variety (preferred by physicians) and economies of scale (preferred by pharmacy directors) for a local hospital. Pre-intervention, the hospital employed an inventory ordering policy, which resulted in high frequency of shortages and emergency refills based on user "experience" instead of statistical predictive modeling solutions. The hospital transitioned to using drug depots that automate orders based on a (s, S) policy. Mathematical models were used to determined that the hospital's previous inventory policies were suboptimal, increased formulary, resulted in increased refills, and, thus, higher order costs and labor requirements.

4.1.2 Characterization. The model in this chapter assumes that treatment efficacies have been defined at a system level based on disease. The volumes of patients with each type of diagnosis are known. This level of characterization is consistent with Chapter 3.

The key difference in this chapter is that supply chain costs are known at the structural level. For the purpose of the research, this is characterized by three measures.
Supply chain fixed cost (entry cost) is the cost associated with operating the supply chain independent of the volume required. If the treatment volume is zero, then there is $0 fixed cost.
Supply chain variable cost is some measure of the rise in cost as a function of the rise in

treatment volume. Cost/volume shape indicates the change in supply chain variable cost as a function of treatment volume. A linear shape means that variable cost is volume indifferent. More common shapes are sublinear based on economies of scale and volume discounts. In the IUD and SUD models from Chapter 3, fixed costs as defined here was $0, variable costs were based on unit costs, and the associated shape was linear.

A graph showing a supply chain with fixed costs and a sublinear shape is shown in Figure 4.1. Note that for a given increase in volume, the associated cost is much higher when the treatment volumes are lower. The assumptions for how each of these known parameters is manipulated to formulate models are provided in the following section.

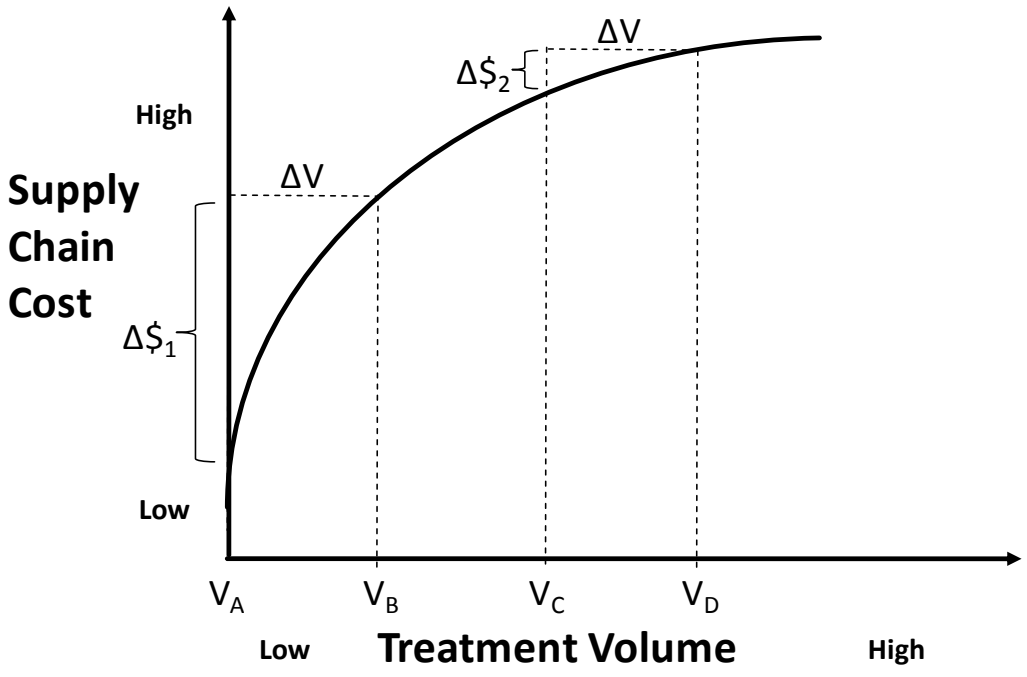

Figure 4.1. Supply chain cost structure.

4.2 Policy Model Formulation

Two policy models are considered. The first models a pre-integration policy of treatment design based on system efficacy perceptions, unit cost consideration, and disease specific / specialization (SUD model from Chapter 3). The second models a post-integration policy of

treatment design based on system efficacy, system cost consideration, and disease specific / specialization (SSD model).

4.2.1 SUD model (pre-cost integration). With a system view of treatment efficacy values, the goal of treatment selection decision makers at the system level is to find the optimal values of the decision variables x_{ij}:, the binary decision for all providers in the system to select treatment *i* for diagnosis *j* to maximize the total system efficacy while minimizing total cost. The MOOP model can be modified and simplified into the following form. Maximize the system's total (or volume-weighted) efficacy value and minimize the total cost, subject to equality constraints to imply that:

- One treatment must be selected for each diagnosis.
- Treatment selection decisions are binary.

The optimization problem of selecting the x_{ij} decision variables are formally expressed in the same SUD model from Chapter 3 as follows:

$$\max \sum_i \sum_j \tilde{e}_{ij} x_{ij} \tag{4.1}$$

$$\min \sum_i \sum_j c_i x_{ij} \tag{4.2}$$

$$\text{s.t. } \sum_i x_{ij} = 1 \ \ \forall \ \ j \tag{4.3}$$

$$x_{ij} = 0 \text{ or } 1 \ \ \forall \ \ i \text{ and } j \tag{4.4}$$

Note that this model emphasizes the disease independence of the decision-making process. Patient volume is not considered. Unit cost information is considered visible by the designer (which is generous based on the research indicating much cost indifference).

4.2.2 SSD model (post-cost integration). The decision variables for this model assume the same form as in the SUD. However, the values of this vector are assessed with a different cost structure. In this model, the cost constants for treatment *i* are variable cost coefficient s_i,

entry cost coefficient d_i, the degree of volume order discounts u_i where $0 < u_i < 1$, and the volume of demand for diagnosis j V_j. These factors are used to drive the assessment of cost associated with treatment i in the SSD as $c_i = d_i y_i + s_i(\sum_j V_j x_{ij})^{u_i}$ where x_{ij} is defined as with the SUD model and y_i is a binary variable set to 0 if no treatment i is used and 1 otherwise. However, in the SUD, $c_i = d_i + s_i$ as explained in Chapter 3.

The SSD model can be modified into the following form. Maximize the system's total efficacy value and minimize the total entry cost plus variable volume-based cost of treatments, subject to the each of the following constraints:

- One treatment must be selected for each diagnosis.
- A one-time entry cost is incurred when at least one unit of a treatment is supplied.
- Treatment selection decisions are binary.

The optimization problem of selecting the x_{ij} decision variables for this problem is formally expressed in the SSD model as follows:

$$\max \sum_i \sum_j \tilde{e}_{ij} x_{ij} \tag{4.5}$$

$$\min \sum_i \left[d_i y_i + s_i \left(\sum_j V_j x_{ij} \right)^{u_i} \right] \tag{4.6}$$

$$\text{s.t. } \sum_i x_{ij} = 1 \ \forall \ j \tag{4.7}$$

$$\sum_j x_{ij} - M y_i \leq 0 \ \forall \ i \tag{4.8}$$

$$x_{ij} = 0 \text{ or } 1 \ \forall \ i \text{ and } j \tag{4.9}$$

$$y_i = 0 \text{ or } 1 \ \forall \ i \tag{4.10}$$

Equation 4.8 is used to set the value of y_i to drive the inclusion of entry costs in the total cost computation. In this equation, y_i is one if any x_{ij} is one for any j and zero otherwise. This formulation requires the M must be as large as any reasonable value for $\sum_j x_{ij} \forall i$.

4.3 Policy Model Solution

For this research, problems are sized and generated to enable solution through enumeration. This approach is taken so that the entire set of feasible solutions might be compared to the efficient frontier. However, for larger more complex problems, the non-linear objective function and fixed cost inclusion complicated traditional optimization methods. A heuristic solution technique is proposed to estimate the frontier of efficient treatment protocols.

4.3.1 Heuristic solution. The heuristic is established in the section based on the following:

- Adjacent points on the efficient frontier tend to have similar solutions.
- Entry costs have a higher impact on unit cost when treatment volumes are low.
- Transitions between efficient points tend to avoid incurring entry costs.
- As costs increase, the increase in efficacy tends to diminish creating points that appear to be on a concave front.

For this heuristic, the fixed cost d_i, the standard value of efficacy e_{ij}, the discount cost coefficient s_i, the degree of the volume discount u_i, and the demand population V_j must be known. A current operational strategy or feasible solution x_{ij} must be established such that exactly one treatment is selected for each diagnosis. Considering that few decision swaps tend to occur between adjacent efficient solutions, the heuristic estimates the change in total efficacy and total cost incurred from swapping one treatment at a time.

4.3.1.1 Efficacy and cost tradeoff estimation. A strategy to estimate alternative treatment costs and efficacies is proposed from observing that very few decision swaps occur between efficient solutions and that entry costs have a higher impact on unit cost when treatment volumes are low. For this strategy, it is assumed that swaps are evaluated one at a time. When

considering alternative treatment decisions in this problem, it is relatively easy to capture the change is efficacy. The change in efficacy ΔE is the difference in the dot product of efficacy and the new decision and the dot product of efficacy and the initial decision. The nonlinear relationship between costs and treatment volumes makes the treatment selection problem more complex than if there were a linear relationship between costs and treatment volumes.

A rapid estimate for the increase in cost due to one treatment swap is proposed. Consider that a swap is made in an initial solution x_o resulting in solution x_f. The initial volume of a treatment V_o, becomes V_f after the swap. Decision makers may decide to consider the average unit cost for treatment to evaluate the change in cost. In this problem, the average unit cost for a treatment with volume v is as follows:

$$AC(v) = \begin{cases} \dfrac{d + sv^u}{v}, & v > 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.11}$$

The cost AC incorporates the fixed cost of d, variable cost s, and some degree of curvature u which implies a nonlinear relationship between cost and volume when $u \neq 1$. Decision makers may decide to consider the instantaneous change in cost for treatment i to evaluate the change in cost. In this problem, the instantaneous change in cost is the derivate of the cost function with respect to u at a given volume v or

$$SC(v) = \begin{cases} usv^{(u-1)}, & v > 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.12}$$

The iterative solution technique for this research considers a weighted combination of the values AC and SC. When there is a small change in treatment volumes, AC is given more weight so that fixed costs will have a greater impact of total cost. When there is a large change in treatment volumes, SC is given more weight because the impact of fixed costs begins to diminish. The weighted estimate for the change in cost is evaluated as $\Delta C = \alpha AC + (1 - \alpha)SC$

where

$$\alpha = \begin{cases} \dfrac{V_f - V_o}{V_f}, & V_o < V_f \\ \dfrac{V_o - V_f}{V_o}, & V_f < V_o \end{cases} \tag{4.13}$$

Figure 4.2 illustrates the relationships between AC and SC along a cost curve where $V_o < V_f$ and when $V_f > V_o$. Notice from this figure that the average unit cost for a treatment AC is always based on the initial volume of a treatment. This reflects the idea that decision makers use the baseline average to predict opportunity cost.

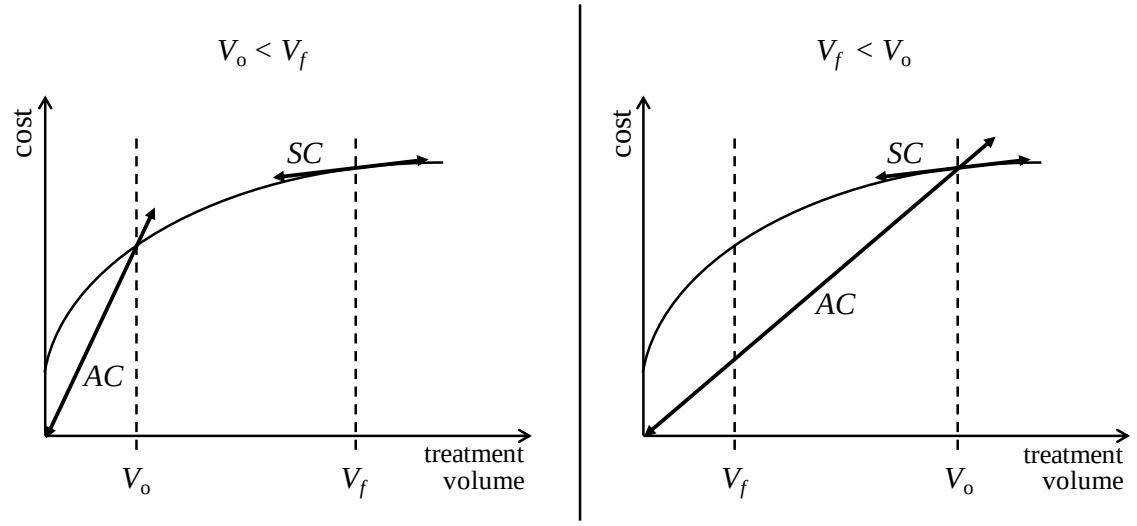

Figure 4.2. AC and SC evaluation based on shifts in treatment volumes.

The instantaneous change in cost SC is always based on the large treatment volume. This reflects the idea that decision makers are most interested in the rise or fall in cost based on the highest volume of use. At this volume, the incremental cost per unit is relatively low. This strategy to estimate alternative treatment costs is used in the heuristic proposed in the following section.

4.3.1.2 Locating improved solutions. After estimating for alternative treatment costs and efficacies have been determined, it is possible to evaluate which alternative improves upon

current operations. The observations discussed in Chapter 3 help to determine these alternatives. Specifically, if it is possible to improve upon the current operations, it is done by in one of three ways: (1) finding a solution with lower cost and lower efficacy that has equal or higher efficacy per cost, (2) finding a solution with lower cost and higher efficacy that has equal or higher efficacy per cost, or (3) finding a solution with higher cost and higher efficacy that has equal or higher efficacy per cost. Based on these three criteria, a search routine is proposed to find improve frontier points within three regions. The relationship between an efficient frontier, these regions and a feasible solution is exhibited in Figure 4.3.

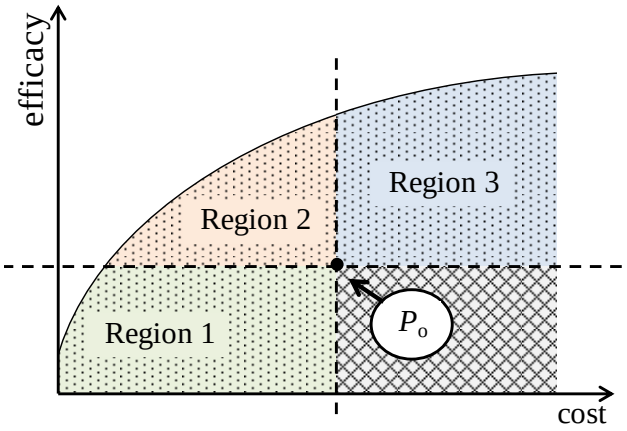

Figure 4.3. Regions of improvement for candidate swap decisions.

In Region 1, a swap will have a lower cost and lower efficacy than the current solution values, P_o. The swap in this region will result in the lowest possible change in efficacy per change in cost. In Region 2, a swap will have a lower or equal cost and an efficacy greater than or equal to the current solution values, P_o. The swap in this region will be furthest away from the current decision. In Region 3, a swap will have a higher cost and a higher efficacy. The swap will have the highest possible change in efficacy per change in cost. Any solution that results in a higher cost for a lower efficacy is less efficient than current solution values, P_o, and, therefore, is not considered as a candidate for swapping.

Based on the searches for swaps in each of these regions, as many as three new solutions can replace the current operational solution. Searches for improvements beyond found solutions may continue until consecutive searches are equal or close enough or the allotted number of iterations for a search is reached. The following heuristic algorithm details this search process for efficient treatment protocols.

4.3.1.3 Heuristic process. Based on the solution characteristics of the SSD, a six step heuristic is used to provide a close estimate for the efficient frontier. The six steps are as follows:

1. Problem initiation. It is assumed the current feasible solution for operations or that a feasible solution of interest is known. The problem is initiated with this solution being a candidate member of the solution set and its objective function values being candidates for the efficient frontier.
2. Decision swap. One swap is made for an element of the solution set. The resulting solution is considered a candidate for the efficient solution set and must be compared to alternative solutions prior to being declared efficient.
3. Swap comparison. The proposed technique for estimating an alternative treatment cost and efficacy are used to determine possible improvements from a swap. The estimated changes in cost and efficacy are vetted for improvement upon solutions present in Region 1, 2, or 3 of the basic feasible region.
4. Solution and efficient frontier update. When candidates for efficiency are found to improve upon a compared solution, the compared solution is replaced the by the candidate solutions. The corresponding objective function values of the compared solutions are replaced as well.

Up to three candidate solutions may be selected to replace the compared solution based in searches in Regions 1, 2, and 3.

5. Efficient frontier concavity check. As a solution set is developed, it is important to ensure that the efficient frontier set maintains concavity. This check is done periodically, in declared intervals, during the heuristic search. Points that violate the requirement for concavity are removed along with the corresponding solutions.
6. Stopping criteria evaluation. The heuristic may be stopped based on at least one of two criterions: The distance between the efficient frontiers of consecutive searches is less than a desired epsilon or the maximum number of iterations is reached. If these criterions are not violated, the search for the efficient frontier continues.

A figure depicting the algorithm in detail is shown in Figure 4.4.

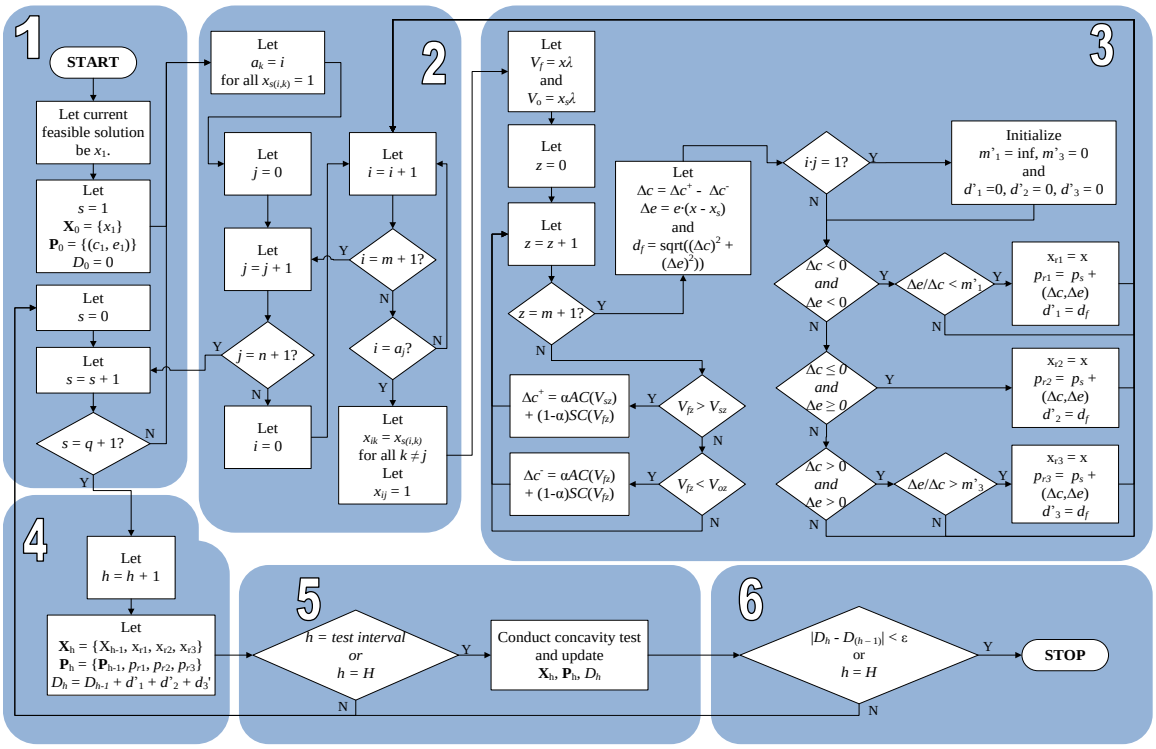

Figure 4.4. SSD heuristic procedure.

4.4 Integration Assessment

4.4.1 Selected factors. Based on the characterization in Section 4.1, the following factors are considered in developing a design of experiments.

4.4.1.1 Patient volume factors. For each problem, the demand volume of patients with disease j serviced by each provider k, V_{lk}, must be known. The variability in V_{lk} is one factor considered in the model. High variability indicates presence of treatment design by both experts and inexperienced providers. Low variability indicates that patient volumes are balanced across designers.

4.4.1.2 Efficacy factors. The parameters used to evaluate efficacy values in this study are e_{ijk}, and V_{lk}. The basis for evaluating total efficacy between problems is that the perceived efficacy may differ between providers and true efficacy may be estimated based on provider expertise. Values of efficacy are normalized to range from 0 to 1 with 0 being ineffective and 1 being completely effective. Given that provider k's perceived efficacy of treatment i for diagnosis j is e_{ijk}, the system calculated efficacy according to $\tilde{e}_{ij} = \left(\sum_k V_{jk}\tilde{e}_{ij}\right)/\left(\sum_k V_{jk}\right)$. The variability in perceived efficacy is a factor in the design of experiments. High variability indicates many different perceptions regarding treatment efficacy. Low variability indicates some consensus in terms of efficacy.

4.4.1.3 Cost factors. The parameters used to generate cost values in this research are d_i and a_i. The parameters for fixed costs and variable cost coefficients are varied from 0 to 100 and then added to form a unit cost. It is assumed that the variable cost of treatment i tends to be directly related to the relative efficacy of treatment i. Understanding that not all variable cost are correlated to treatment efficacies, a "noise" factor is added such that the variable cost of treatment i is $s_i = a_i(\sum_j \tilde{e}_{ij})/(\sum_j \tilde{e}_{ij}) \pm \sigma_i$. The cost of selecting treatment i is calculated as

$c_i = d_i y_i + s_i \left(\sum_j V_j x_{ij} \right)^{u_i}$. Variability in fixed costs and variability in variable costs are factors in the design of experiments.

4.4.2 Design of experiments. The objective of the experimental design is to understand the relationship between parameters (variance in provider's patient volumes, variance in provider's perceived efficacy, variance in entry cost, variance in variable cost, and different shapes) on the performance of treatment selection protocols of the SUD in comparison to the SSD. Parameters are set in a manner that allows for inferences to be made relative to a system progressing through the four policies of treatment protocols selection.

4.4.2.1 Generation of problem instances. In this study, a problem with three treatments, three providers, and three diagnoses is examined. There are five parameters used to generate the two problems in this research. The List of Symbols defines these parameters. These parameters are generated using custom programs in Matlab 7.12.0 (R2011a). This section discusses the design of experiments for comparing solutions of the two models. First, the methods used to generate their values for the design of experiments are provided.

4.4.2.2 Random variable generation. The values for d_i, e_{ijk}, a_i, and V_{lk}, are randomly generated from beta distributions as described in Chapter 3. The Matlab code for each of the random number generators is found in the APPENDIX. The problem parameters varied as a part of this research are summarized in Table 4.1. Ten replications are executed for each factor level combination in the design for a total of 320 observations.

Table 4.1

Study 2 experimental problem factors and levels.

Factor	Factor Description	Factor Levels
1	Variance in Fixed Costs between Treatments	1-High, 2-Low
2	Variance in Variable Costs between Treatments	1-High, 2-Low

Table 4.1.

Cont.

3	Degree of Volume Discounts	1-High, 2-Low
4	Variance in Patient Volumes between Providers	1-High, 2-Low
5	Variance in Perceived Efficacy between Providers	1-High, 2-Low

4.5 Integration Analysis

A total of 320 observations were used to assess model performances under varied experimental factors. In this section, efficient solutions of the SUD and SSD models are compared to the optimal treatment protocol solutions generated from total enumeration. First, a brief discussion of the SSD Heuristic is provided.

4.5.1 SSD heuristic performance. Objective function values of heuristic solutions and total enumeration solutions are compared to assess the effectiveness of the SSD Heuristic based on the stated design of experiments. The average execution time for the heuristic was 0.76 seconds. The solutions found using the heuristic were efficient solutions with an average change in cost of 19% and an average change in efficacy of 3%. Tuning of the heuristic is an opportunity for future research.

4.5.2 SUD-SSD comparison. Efficient frontier analyses help characterize some properties of optimal treatment protocols. The efficient frontiers compared used the decision variable output of each model and assessed it using the supply chain structure. Three measures were used to assess the performance of the SUD model relative to the SSD: average improvement in efficacy, average improvement in cost, and average error rate. The average improvement in efficacy and average improvement in cost are based on the area of opportunity between the frontier of the SUD and the SSD. No instances of "undefined" behavior were observed.

Table 4.2 lists the average performance results of the SUD relative to the SSD by experimental factor. The most outstanding property shown in comparing model solutions is that the SSD outperforms the SUD across all performance measures. Also, the SUD tends to perform closer to the SSD when there is low variance in problem factors. Improvements in efficacy from the SUD are minimal and appear to have no sensitivity with regard to changes in factor levels. However, improvements in cost are relatively substantial and do show sensitivity to study factors. Most of the error in the performance of the SUD is attributed to the difference in total cost calculations.

Table 4.2

SUD-SSD performance by experimental factor.

Factor Levels	Improvement in Cost	Improvement in Efficacy	Average Error Rate
Variance in Fixed Costs between Treatments			
High	17%	2%	34%
Low	7%	1%	24%
Variance in Variable Costs between Treatments			
High	15%	1%	25%
Low	9%	2%	33%
Degree of Volume Discounts			
High	14%	2%	33%
Low	10%	1%	25%
Variance in Patient Volumes between Providers			
High	12%	2%	31%
Low	11%	2%	27%
Variance in Perceived Efficacy between Providers			
High	11%	2%	27%
Low	12%	1%	31%

Overall, the average improvement in cost, improvement in efficacy, and error rate are 12%, 2%, and 29%, respectively. The following sections provide details of performance based on each experimental factor.

4.5.2.1 Provider variances. The total efficacies for both problems are calculated using the same factors. Therefore, it can be expected that there is little sensitivity to changes in provider variances when comparing the SUD and SSD. The average improvement in cost changes slightly from 12% to 11%, the average improvement in efficacy changes slightly from 1% to 2%, and the average error rate changes from 31% to 27% as the variance in perceived efficacy between providers is changed from low to high.

4.5.2.2 Patient volumes. Patient volumes were also used in the formulation of total cost and total efficacy values. Accordingly, this factor does show an impact on performance that is similar to the trends in variance in perceived efficacy between providers. The average improvement in cost is slightly increased from 11% to 12% as variance in patient volumes between providers is increased. The average improvement in efficacy is unchanged, 2%. However, the average error rate is increased from 27% to 31% as the variance in this factor is changed from low to high.

4.5.2.3 Fixed costs. All performance values are increased as the variance in fixed cost between treatments increased. The average change in cost is increased from 7% to 17%, the average change in efficacy is increased from 1% to 2%, and the average error rate is increased from 24% to 34% as variance in this factor is changed from low to high. Analyses of this factor provide the most support for heightened transparency of treatment costs. Over all factor levels, the greatest improvement in cost (17%) occurs when the variance in fixed cost between treatments is classified as high. Efficacy is not directly implied in the calculation of fixed costs

as in the case of variable cost (refer to cost factors). Fixed costs are random or unbiased to treatment efficacies. Without making decisions for treatment selection in lieu of fixed cost, performance in decreased.

4.5.2.4 Variable costs. Unlike fixed costs, variable costs are related to treatment efficacies. Therefore, if the efficacy of a treatment increases, the variable cost of the treatment is also expected to increase. When patient volumes are constant, the efficient frontiers for the SUD and SSD will make proportionate shifts as shown the first study of this research project. However, study results show greater improvements in cost (by 6 percentage points) when the variance in variable costs between treatments is classified as high. Contrarily, the error rate decreases from 33% to 25% when the factor level is changed from low to high.

4.5.2.5 Volume discounts. The SUD shows better performance across all measures when the degree of volume discounts is classified as low. The average improvement in cost is increased from 10% to 14%, the average improvement in efficacy is slightly increased from 1% to 2%, and the average error rate is increased from 25% to 33% when the system incorporates a higher level of volume discounts. For this study, a low level to volume discounts corresponds to a linear relationship between treatment cost and treatment volumes. For a system that desires to leverage economies of scale, this assumption is impractical. Such a system should expect to see greater improvements in performance after successful integration for supply chain management.

4.6 Integration Impact

4.6.1 Impact of cost variability. Small SSDs were assessed in preliminary studies to explore the impact of variable costs, fixed costs, and cost-volume relationships. This section discusses characteristics in formulary found in a sensitivity analysis of variable cost. The preliminary study considered a model with three treatments for the six diseases in the

preliminary model, a total of 729 feasible solutions were generated. The values of s_i were set at extremes 0.05 and 10.00 to assess the impact of unit costs as other variables were unchanged.

As a representation of economies of scale, it was assumed that the unit costs were a function of the square root of the sum of each treatment type times the supply chain cost coefficient to imply sublinear cost to volume relationships. That is, $u_i = 0.5$ for all i. The feasible solutions were derived using enumeration. The associated values for total efficacy and total cost per treatment design solution were then calculated using Microsoft Office Excel 2007. Figure 4.5 (a) and (b) exhibit the cost versus efficacy of feasible solutions for a system with high variable costs and low variable costs, respectively.

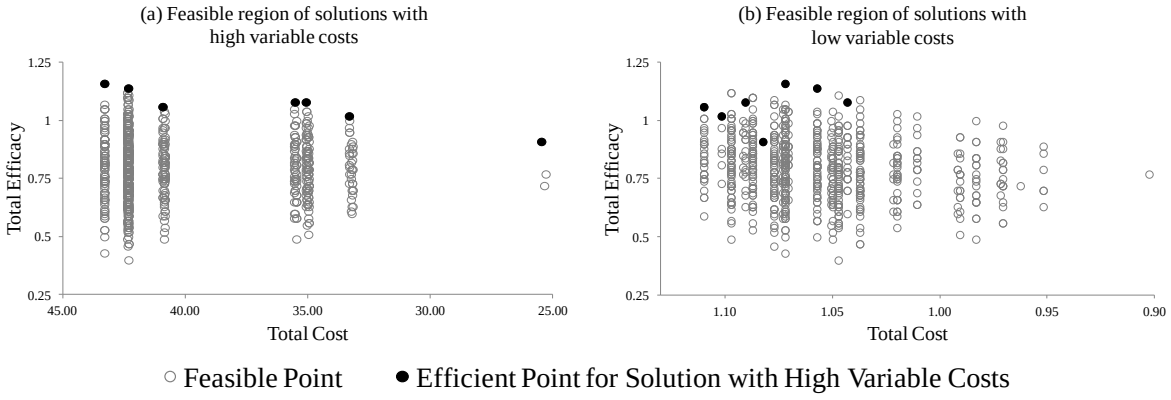

Figure 4.5. Total cost versus efficacy for (a) high variable cost and (b) low variable cost solutions.

Upon visual inspection of feasible points, it is possible to notice that solutions with high variable costs are more segregated. Intuitively, it was assumed that this occurred because the variances between total costs were higher when unit costs were higher. Another explanation was provided after viewing the structure of efficient solutions for each cost. As total costs increase, the product mixes of efficient points in this problem become increasingly diverse. As an example, consider the three feasible points in the lowest cost region (the far right) of Figure

4.5(a). Each of these points corresponds to solutions that select only one treatment for each of the diagnosis in this sample problem. Moving to the next region of points where total costs are an estimated 33 units, points correspond to selecting one treatment for five diagnoses and another for the sixth. This pattern of increasing treatment diversity continues through to the region where total costs are estimated at 44 units. In this final region, each treatment is assigned to exactly two diagnoses. This pattern in treatment diversity by total costs was not recognized for the low unit cost solutions or in the case of the SUD where treatment cost have a linear relationship to treatment volumes.

After observing the pattern in segregation for solutions with high variable costs, the efficient points for these solutions were further examined. The points forming the efficient frontier of Figure 4.5 (a) are black and filled to help discern these points from other feasible solutions. Figure 4.5 (b) illustrates that these points do not form the efficient frontier when variable costs are low. It is inferred that efficient solutions may not remain efficient as variable costs are changed. Most of the efficient solutions for the case with high variable cost are not efficient solutions when variable costs are low and vice versa. However, it is observed that the efficient solutions in the high efficacy region do remain the same as proposed in the first study of this research project. It is also observed that the number of solutions changes.

4.7 Chapter Summary

The study discussed in this chapter is unique in modeling and assessing a second study of integration for healthcare supply chain management. The study examined considers the implications of supply chain integration in informing treatment cost evaluations. A heuristic is proposed as a means of understanding system behaviors and the composition of efficient

treatment selection protocols. It is shown that heightened awareness of treatment costs leads to better overall system performance.

The final study of integration for this research project will explore the intricacies of treatment interactions and diagnosis severity levels. In the following study, a genetic algorithm (GA) is introduced as a solution method for treatment selection protocols of a complex problem. Properties of SSD efficient solutions are used to develop the GA.

CHAPTER 5

The Impact of Whole Patient Integration

The primary goal of this chapter is to determine the strategic implications of healthcare supply chain integration for a decentralized healthcare system that desires minimized cost and maximized efficacy. A genetic algorithm and a unique comparative analysis technique are also provided as a means to achieve the primary goal. The study examines a healthcare system before and after a decision support system enables holistic patient care. A complete design of experiments is conducted to determine the performance impact of provider variance, varying cost and efficacy structures, patient severity levels, and demand volumes on treatment selection protocols. It is determined that low variance between system providers, leveraging economies of scale, and full patient views (as opposed to per diagnosis) are among factors that lead to greater system efficiency. Recommendations are made to integrate treatment selection protocols in order to create more visibility and provide improved, holistic patient care.

5.1 System Segmentation and Integration Opportunity

The final segmentation challenge addressed is that created by the treatment of a patient by different physicians according to specialty. In considering the impact of integration in this case, a new representation of efficacy is needed.

5.1.1 Background. Strech et al. (2009) survey studies of physician acceptance of treatment rationing and find acceptance ranging from 9% to 94%. The study infers that previous researchers have conducted studies about healthcare coordination in in order to promote institutional agendas. Strech et al. (2010) discuss the ethics involved in evidence-based allocation and rationing policies and encourage training, reinforced consistency in decision making, and increasing transparency as a means to dispel adverse effects on healthcare delivery.

5.1.2 Characterization. With the whole patient view, treatment design is based on the patient type rather than a specific disease (where a patient may have one or more). Opportunities for synergy exist where a treatment might address multiple disease conditions. Risk may occur when separate physicians suggest treatments which in combination have negative side effects. As a result, a generalized view of efficacy as a function of treatment amount is constructed. A generalized view of the relationship between efficacy and utilization for a treatment under the three forms is given in Figure 5.1.

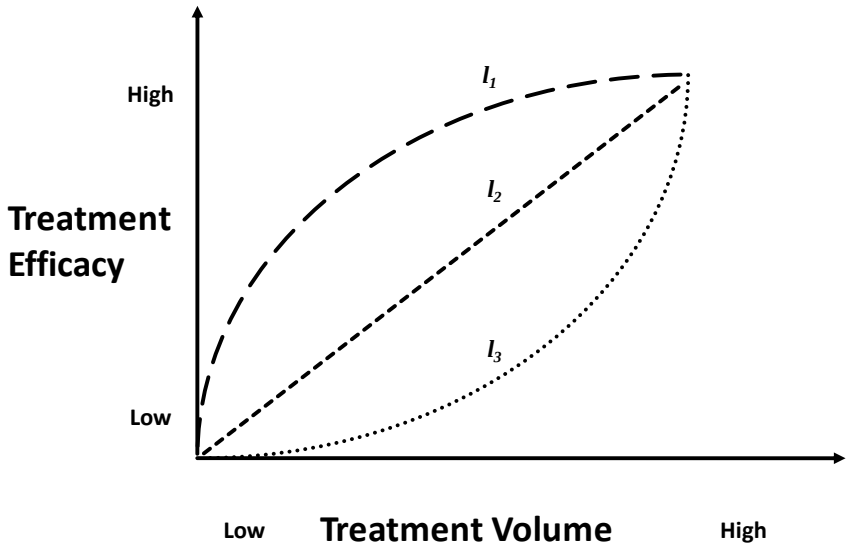

Figure 5.1. Superlinear, linear, and sublinear treatment and efficacies.

One form the treatment and efficacy relationship is assumed to result in a concave shape between efficacy and utilization as indicated by curve l_1. Efficacy has diminishing returns as treatment increases. In the second form, there is a direct relationship between treatment efficacy and utilization, implying linearity. In the third and final form, a treatment may provide minimal efficacy at low levels of utilization but show tremendous gains when approximating full "dosage". This form is implied when $w_{il} > 1$ as indicated by curve l_3. (Variable w_{il} is further defined in section 5.2.2 as a degree of efficacy function curvature.) This representation allows

consideration of issues of "rationing." This topic is highly sensitive in the public policy context, but a key consideration when balancing cost/efficacy impacting policies. When considering the efficacy assessment for a patient, the severity of their condition should also be considered. Two policy models are considered. The first models a pre-integration policy of treatment design based on system efficacy perceptions, system cost consideration, and disease specific / specialization (SSD model from Chapter 4). The second models a post-integration policy of treatment design based on system efficacy, system cost consideration, and whole patient consideration (SSW model). The second model includes representation of treatment volume to efficacy relationships and consideration of patient type risk/severity.

5.2 Policy Model Formulation

Two policy models are considered. The first models a pre-integration policy of treatment design based on system efficacy perceptions, system cost consideration, and disease specific / specialization (SSD model from Chapter 4). The second models a post-integration policy of treatment design based on system efficacy, system cost consideration, and whole patient consideration (SSW model). The second model includes representation of treatment volume to efficacy relationships and consideration of patient type risk/severity.

5.2.1 SSD model (pre-patient integration). With a consensus of treatment efficacy values, the goal of treatment selection decision makers at the system level is to find the optimal values of the decision variables x_{ij}, the binary decision for all providers in the system to select treatment i for diagnosis j to maximize the total system efficacy while minimizing total cost. The cost factors for treatment i are variable cost coefficient s_i, entry cost coefficient d_i, the degree of volume order discounts u_i where $0 < u_i < 1$, and the volume of demand for diagnosis j V_j.

Using the same decision variables from Chapter 4, x_{ij}, the problem is formally expressed in SSD model as follows:

$$\max \sum_i \sum_j \tilde{e}_{ij} x_{ij} \tag{5.1}$$

$$\min \sum_i \left[d_i y_i + s_i \left(\sum_j V_j x_{ij} \right)^{u_i} \right] \tag{5.2}$$

$$\text{s.t. } \sum_i x_{ij} = 1 \ \forall \ j \tag{5.3}$$

$$\sum_j x_{ij} - M y_i \leq 0 \ \forall \ i \tag{5.4}$$

$$x_{ij} = 0 \text{ or } 1 \ \forall \ i \text{ and } j \tag{5.5}$$

$$y_i = 0 \text{ or } 1 \ \forall \ i \tag{5.6}$$

Note that this model emphasizes the disease independence of the decision-making process. Decision variables are binary. Patient volume is considered. System cost information is considered visible by the designer.

5.2.2 SSW model (post-efficacy integration). With consideration for patient types, the goal of treatment selection decision makers at the system level is to find the optimal values of the decision variables x_{il} as selected portion of treatment i for patient type l to maximize the total system efficacy while minimizing total cost. The cost factors for treatment i are the same as in the SSD. The efficacy factors for this problem are the severity of disease/diagnosis g_j, the severity for patient type G_l, the volume of demand V_l, and the degree of efficacy function curvature w_{il}.

The SSD model can be modified and simplified into the following form. Maximize the total system efficacy value and minimize the cost of treatments, subject to the each of the following constraints:

- A portion of at least one treatment must be utilized for each diagnosis – recall, do nothing can be a treatment option.
- The sum of utilization for treatments must not exceed the capacity (maximum dosage) for a patient type.
- Treatment utilization decisions are normalized to be between zero and one.

Using the decision variables x_{il}, the problem are formally expressed in SSW model as follows:

$$\max \sum_i \left(G_l / \sum_l G_l\right)\left(V_l / \sum_l V_l\right) \sum_{j \in l} \left[\left(1 - \prod_i \left(1 - \left(\tilde{e}_{ij} x_{il}\right)^{w_{il}}\right)\right)\right] \tag{5.7}$$

$$\min \sum_i \left[d_i y_i + s_i \left(\sum_j V_j x_{ij} \right)^{u_i} \right] \tag{5.8}$$

$$\text{s.t. } \sum_j x_{ij} - M y_i \leq 0 \quad \forall \ i \tag{5.9}$$

$$\sum_i x_{il} \leq Cap_l \quad \forall \ l \tag{5.10}$$

$$x_{il} \leq \sum_i x_{il} \quad \forall \ i \text{ and } l \tag{5.11}$$

$$0 \leq x_{il} \leq 1 \quad \forall \ i \text{ and } l \tag{5.12}$$

$$y_i = 0 \text{ or } 1 \quad \forall \ i \tag{5.13}$$

Equation 5.7 is the efficacy objective function. The equation is weighed by the relative patient type severity and volume in addition to the weight of the individual diagnosis. These weights are calculated such that the severity of a patient type with multiple diagnoses will be higher than if the patient were to have a single diagnosis. Additionally, equation 5.7 considers that efficacy may be nonlinear with respect to a selected treatment volume. A capacity constraint is placed in the allotted treatment volume to ensure that treatments are not assigned beyond a tolerable threshold in equation 5.10.

5.3 Policy Model Solution

SSD policy model solution is shown in Chapter 4. SSW solution is performed by an enhanced genetic algorithm.

5.3.1 Integrated treatment selection genetic algorithm. The exact solution for SSW is infeasible due to the complexity and dimensionality of the problem. Therefore, previous models are studied to capture the essence of practical decision problems that are used to effectively solve the SSW using the integrated treatment selection genetic algorithm (ITSGA), an iterative solution procedure. This section discusses the formulation the ITSGA. The discussion is preceded with an overview of genetic algorithms (GAs) including important terminology and the use of GAs in efficient frontier analyses.

5.3.1.1 Genetic algorithm overview. The concept of genetic algorithms (GAs) was developed by Holland in the 1970s. The use of GAs is inspired by Darwin's theory of evolution and natural selection. According to natural selection, weaker members of a species will face extinction. Stronger members are more likely to be selected for reproduction and pass their genes on to future generations. As the population reproduces, random mutations may occur so that new genetic traits are presented. If these mutations are advantages, the species evolves to maintain the new genetic traits.

In artificial systems, a GA is an iterative procedure that uses a current population to create "children" that make up the next generation for the population. The algorithm selects "parents" as members from the current population that will contribute their "genes" or "chromosomes" to their children. According to natural selection, the algorithm usually selects parents that are strong or have the best "fitness value" among the population. The reproduction in a GA creates three different types of children for the next generation. "elite children" are the

members of the current generation that have the best "fitness value" among the population and, consequently, automatically survive to the next generation. "Crossover children" are created by combining the genes of a pair of parents. "Mutant children" are created by mutating a single parent.

The first multi-objective GA was not proposed until 1985 by Schaffer (1985), although the concept of GAs (Holland, 1975) and Pareto front analyses (Markowitz, 1959) were well developed beforehand. Since that time, evolutionary algorithms (GAs or derivatives) have become the most popular heuristic approach to multi-objective optimization problems (Jones, 2002). Konak et al. (2006) provide a thorough tutorial on multi-objective optimization using optimization and point out the benefits of using GAs in Pareto front analyses.

There are many other studies that introduce or compare performance of varying multi-objective optimization GAs (Coello, 1999; Fonseca and Fleming, 1998; Jensen, 2003; Xiujuan and Zhongke, 2004; Zitler and Thiele, 1999). Parameter tuning for GAs has been given extensive research (Villegas et al., 2006; Medaglia et al., 2009; Zandieh and Karimi, 2011). The Niched Pareto Genetic Algorithm (Horn and Goldberg, 1994), Nondominated Sorting Genetic Algorithm (NSGA) (Srinivas, N. and Deb, 1994), Fast Non-dominated Sorting Genetic Algorithm (NSGA-II) (Deb et al., 2002), Strength Pareto Evolutionary Algorithm (SPEA) (Zitzler, E. and Thiele, 1999), Pareto-Archived Evolution Strategy (PAES) (Knowles and Corne, 2000) are among the more common GAs for multi-objective optimization.

Early versions of multi-objective GAs do not include considerations for problem constraints. Anagnostopoulos and Mamanis (2009) conclude that the SPEA, PAES, and NSGAII all perform well for multi-objective portfolio selection problems with or without constraints.

The aim of this research is not to provide a comparison of varying GAs or to develop a methodology for GA parameter tuning. However, this study does survey the structures of popular multi-objective, multi-constraint optimization GAs that have been applied to Pareto front analyses in order to establish parameters for the ITSGA discussed in Section

Table 5.1 summarizes a survey of GA parameters used in ten relevant studies. Each of the listed studies applies a GA to a multi-objective optimization problem that has multiple constraints. The majority of the studies are for bi-objective problems. With the exception of Prakash et al. (2012), Anagnostopoulos and Mamanis (2011), Bowman et al. (2010) and Ko and Wang (2010), the studies have binary decision variables. Bowman et al. (2010), Goyal (2011), and Vlah Jeric and Figueria (2012) have 5, 24, and 4 objectives, respectively.

A variation of schemes is used to set population sizes. Many aim at setting the population size relative to the number of decisions or objectives. Prakash et al. (2012) propose a knowledge-based genetic algorithm (KBGA) that selects alternates between various crossover and mutation operators as the system acquires knew knowledge.

Single point crossover operators are used where gene member represent a set of resources such as surgical suites or time shifts for labor (Bowman et al., 2010; Medaglia et al. ,2009, and Gul et al., 2011). Single point crossover operators swap portions of parent genes at a single point. Other studies apply uniform operators that all the combination or switching of gene members among parents with equal probability in order to form crossover children Anagnostopoulos and Mamanis (2011), Goyal (2011), Alves, M. Almeida (2007), Ko and Wang (2010).

Table 5.1

GA parameters in applied multi-objective, multi-criteria optimization problems.

Reference	Number of objectives	Population size	Selection strategy	Crossover operator (rate)	Mutation rate	Elite children	Generations
Alves, M. Almeida (2007)	2,3,4	150, 350	Tournament	Uniform (80%)	10%	N/A	20× (2×#Objs+1)
Anagnostopoulos and Mamanis (2011)	2	500	Tournament	Uniform (100%)	10%	N/A	500
Bowman et al. (2010)	5	64×#Objs	Tournament	Single point (70%)	1/N	50%	100
Goyal (2011)	24	200	Tournament	Uniform (70%)	5%	N/A	500
[*]Gul et al. (2011)	2	40	Tournament	Single point (90%)	1/N	25%	50
Ko and Wang (2010)	2	20, 70	Roulette	Uniform (90%)	1/N	N/A	200,500
[*]Medaglia et al. (2009)	2	50	Tournament	Single point (70%)	5%	N/A	100
[*]Pato and Moz (2000)	2	400	Tournament	PMX (60%)	0.10%	2	≤ 2000
[ǂ]Prakash et al. (2012)	2	Custom	Custom	Custom	Custom	N/A	Custom
[*]Vlah Jeric and Figueira (2012)	4	50	Tournament	XVRA (90%)	10%	N/A	N/A

Mutation rates are set low across all studies in the literature, varying between 5% of the population size, 10% of the population size, and 1/N where N is the number of variables in a problem. More recently, multi-objective GAs have employed elitism strategies and proven to outperform traditional GAs that neglect elitism. Three of the listed studies employ elitism using one of two strategies: (1) maintaining elite children in the population throughout the generations (Pato and Moz, 2000) or (2) storing elite solutions in an external archive and reintroducing the elites to the population (Bowman et al., 2010 and Gul et al. 2011). The second strategy maintains a population that is usually 25 – 50% of the desired population size and therefore requires more memory and execution time. The number of generations range from 50 to 2000.

5.3.1.2 ITSGA methodology. The ITSGA is implemented using the multi-objective optimization functionality of the Matlab 7.12.0 (R2001a) GA toolbox. The following outline summarizes how the ITSGA works:

0. A user inputs population size P, termination condition X, and GA parameters e%, c%, and p%.
1. The algorithm begins by creating an initial population. Each gene in a population henceforth is the length of the number of variables in the SSW. Position $T\times(l - 1) + i$, in a gene takes the value of x_{il} where T is the number of treatments in the SSW. The initial population is often created at random in GAs. The specific initial population generation used in this study is detailed later in this section.
2. The algorithm creates a sequence of new populations. At each step, the algorithm uses the individuals of the current generation to create the next population. The create the new population, the algorithm performs the following steps:
    a. Scores each member of the population by computing its fitness value.

b. Scales the raw fitness scores to convert them into a more usable range of values.

c. Selects members called "parents" based on their fitness values.

d. Performs an elitist reproduction strategy on $P \times e\%$ members where $e\%$ of the best individuals (individuals with the best fitness values) in the current population are passed on to the next generation.

e. Produces $(100\% - e\%) \times P \times c\%$ crossover children by tournament selection. The tournament selection method is used to select parents for crossover children. The scattered crossover operator is used in this study and detailed in the next section

f. Produces $(100\% - c\%) \times (100\% - e\%) \times P$ mutant children. In mutation, a child is produced by making random changes to a single parent. The specific mutation operator used in this study is detailed in the next section.

g. Replaces current population with the next generation.

3. The algorithm stops when termination condition X is met.

To generate the initial population, the concept of low cost and high efficacy regions as discussed in Chapter 3 is applied to help ensure that GA generates a frontier that covers the full range of the actual frontier and possibly contains 0-1 solutions where they are desired. With regard to the properties of efficient solutions in the low cost region of the efficient frontier, The ITSGA considers P_C individuals of the initial population as solutions in the low cost region of the feasible solution set. The GA estimates the cost of treatment i as $C_i = (d_i + s_i(\sum_l V_i)^{u_i})/(\sum_l V_i)$. The value of $\min_i(C_i/\sum_i C_i)$ decreases as the range of treatment costs increases. Additionally, efficient points tend to be more segregated with fewer points within the low cost region as the range in treatment costs increases. Based on this tendency, the

ITSGA sets $x_{al} = 1$ and $x_{i\neq a,l} = 0$ for all l where $C_a/\sum_i C_i = \min_i(C_i/\sum_i C_i)$. The value $P_C = \lfloor \min_i(C_i/\sum_i C_i) \rfloor \times P$.

With regard to the properties of efficient solutions in the low cost region of the efficient frontier, The ITSGA considers P_E individuals of the initial population as solutions in the low cost region of the feasible solution set. The GA estimates the efficacy of treatment i as $E_i = \max_i(\sum_j e_{ij}/\sum_i \sum_j e_{ij})$. The value of $\max_i(E_i/\sum_i E_i)$ increases as the range of treatment efficacy increases. Additionally, efficient points tend to be more segregated, with fewer points within the high efficacy region, as the range in treatment efficacies increases. To avoid overlap with the members assigned to the low cost region, the number of candidates accepted as initial solutions in the high efficacy region is as follows:

$$P_E = \begin{cases} \left\lfloor \min_i\left(E_i/\sum_i E_i\right) \right\rfloor \times P, & P_C + \left\lfloor \min_i\left(E_i/\sum_i E_i\right) \right\rfloor \times P \le P \\ P - P_C, & \text{otherwise.} \end{cases} \tag{5.14}$$

Based on this tendencies of efficient solutions of the high efficacy region, the ITSGA sets $x_{bl} = 1$ and $x_{i\neq b,l} = 0$ for all l where $E_b/\sum_i E_i = \max_i(E_i/\sum_i E_i)$ for P_E members of the initial population. For the genes of any individual not assigned as a solution in the low cost region or high efficacy region of the initial feasible solution set, the ITSGA randomly assign values such that $\sum_i x_{il} = 1$ and $x_{il} = [0, 1]$.

As suggested, crossover children are created using the tournament selection strategy. The tournament proportion is p%. Therefore, $P \times p$% genes are selected at random from the population and the individual with the best fitness value is selected as a parent. The scattered crossover strategy is used. For scatter crossover, a random binary vector is created. The vector is used to select genes from the first parent where 1s are present. Otherwise, genes are selected from the second parent.

A custom mutation operator is used to ensure diversity and discourage convergence to a small region. For 50% of mutant children, gene elements are randomly set to 0 for each *l*. For another 25% of mutant children, genes are unchanged. For the remaining mutant children, gene elements are set to $\Sigma_{j \in l}(g_l e_{ij})/\Sigma_{j \in l} g_l$ for each element corresponding to x_{il} so that greater utility is assigned to a diagnosis with high efficacy and high severity values.

5.3.1.3 ITSGA parameters. The GA parameters used in this study are set according to suggestions made in literature in order to ensure the quality and timeliness of the ITSGA. In accordance to the Bowman et al. (2010) study, the population size for this study is set to 64 times the number of objectives. Stall generation limit is 100 to ensure that the GA terminates when there is no improvement in the objective function for a sequence of consecutive generations of length stall generations. This study also employs the more common tournament selection strategy. The crossover rate is set to 0.8 and the Matlab crossover heuristic operator is used with a weight of 1.2. The remainder of the population undergoes the custom mutation operator. Mutation occurs at a low rate as suggested in most literature. This study employs a 0.2 mutation rate (1 – crossover rate) using a custom mutation operator. For more detail on the Matlab crossover heuristic or other default values and operators, the reader is referred to the Genetic Algorithm and Direct Search Toolbox (2004). The parameters used for the ITSGA in this study are summarized as follows:

- Population Size = 128
- Initial Population = Custom
- Elite Count = 2
- Selection = Tournament
- Crossover Fraction = 0.8

- Crossover Operator = Matlab crossoverheuristic
- Mutation Operator = Custom
- Stall Generation Limit = 500

5.4 Integration Assessment

5.4.1 Selected factors. Based on the characterization in Section 4.1, the following factors are considered in developing a design of experiments.

5.4.1.1 Patient volume factors. For each problem, the demand volume of patients with disease j serviced by each provider k, V_{jk}, must be known. The variability in V_{jk} is one factor considered in the model. High variability indicates presence of treatment design by both experts and inexperienced providers. Low variability indicates that patient volumes are balanced across designers. As patient types are defined by disease combinations, the demand volume of patient type l is defined as $V_{lk} = \sum_{j \in l} V_{jk}$.

5.4.1.2 Efficacy factors. The parameters used to evaluate efficacy values in this study are e_{ijk}, and V_{lk}. The basis for evaluating total efficacy between problems is that the perceived efficacy may differ between providers and true efficacy may be estimated based on provider expertise. Values of efficacy are normalized to range from 0 to 1 with 0 being ineffective and 1 being completely effective. Given that provider k's perceived efficacy of treatment i for diagnosis j is e_{ijk}, the system calculated efficacy according to $\tilde{e}_{ij} = \left(\sum_k V_{jk}\tilde{e}_{ij}\right)/\left(\sum_k V_{jk}\right)$. The variability in perceived efficacy is a factor in the design of experiments. High variability indicates many different perceptions regarding treatment efficacy. Low variability indicates some consensus in terms of efficacy. Four levels are used for the efficacy curve shape: superlinear where $w_{il} = 2$, linear where $w_{il} = 1$ and sublinear where $w_{il} = 0.5$. The fourth level includes a random mix of shapes.

There are two variables considered for severity, the severity of an independent diagnosis, g_j, and the severity of a patient group, G_l. The value of g_j ranges from 1 to 10 with 1 being the lowest and 10 being the highest level of severity. The severity values of G_l follow the same ranking scale and have the following relationship to g_j:

$$G_l = \frac{10^x - \prod_{j \in l}\left(10 - g_j\right)}{10^{x-1}} \tag{5.15}$$

The factor has two levels based on the variability in patient condition severity.

5.4.1.3 Cost factors. The parameters used to generate cost values in this research are d_i and a_i. The parameters for fixed costs and variable cost coefficients are varied from 0 to 100 and then added to form a unit cost. It is assumed that the variable cost of treatment i tends to be directly related to the relative efficacy of treatment i. Understanding that not all variable cost are correlated to treatment efficacies, a "noise" factor is added such that the variable cost of treatment i is $s_i = a_i(\sum_j \tilde{e}_{ij})/(\sum_j \tilde{e}_{ij}) \pm \sigma_i$. The cost of selecting treatment i is calculated as $c_i = d_i y_i + s_i(\sum_j V_j x_{ij})^{u_i}$. Variability in fixed costs and variability in variable costs are factors in the design of experiments.

5.4.2 Design of experiments. The objective of the experimental design is to understand the relationship between parameters (variance in fixed cost, variance in provider's perceived efficacy, variance in disease severity, variance in variable cost coefficient, degree of volume discounts, variance in provider's patient volumes, and degree of efficacy curvature) on the performance of treatment selection protocols of the SSD in comparison to the SSW. This study does not intend to conduct an exhaustive study for parameter tuning. Parameters are set in a manner that allows for broad yet critical inferences to be made relative to a system evolving through the four policies of treatment protocols selection.

5.4.2.1 Generation of problem instances. In this study, a problem with three treatments, three providers, and three diagnoses (which implies seven patient types) is examined. There are seven parameters used to generate the four problems in this research. These parameters are generated using custom programs in Matlab 7.12.0 (R2011a).

5.4.2.2 Random value generation. The values for d_i, e_{ijk}, g_j, a_i, and V_{lk}, are randomly generated from beta distributions as described in Chapter 3. The Matlab code for each of the random number generators is found in the APPENDIX. The problem parameters varied as a part of this research are summarized in Table 5.2. Ten replications are executed for each factor level combination in the design for a total of 2560 observations.

Table 5.2

Study 3 experimental problem factors and levels.

Factor	Factor Description	Factor Levels
1	Variance in fixed cost	1-High, 2-Low
2	Variance in provider's perceived efficacy	1-High, 2-Low
3	Variance in disease severity	1-High, 2-Low
4	Variance in variable cost coefficient	1-High, 2-Low
5	Degree of volume discounts	1-Concave, 2-Linear
6	Variance in provider's patient volumes	1-High, 2-Low
7	Degree of efficacy curvature	1-Concave, 2-Linear, 3-Convex, 4-Random

5.5 Integration Analysis

A total of 2560 experiments were used to assess model performances under varied experimental factors. In this section, efficient solutions SSD models are compared to the optimal treatment protocol solutions generated from the ITSGA for the SSW model. First, a brief discussion of the performance of the ITSGA is provided.

5.5.1 ITSGA performance. The performance of the ITSGA is assessed based on the average distance between generational fitness values, termination condition, number of efficient solutions determined, number of generations required to terminate, spread, and run time. The

ITSGA always terminated due to a convergence to a relatively low difference between fitness values. The number of generations required to terminate ranged from 502 to 865 with a median of 502. The median run time was 69 seconds with a maximum of 118 seconds. The median distance and spread were 0.01 and 0.44, respectively. The median number of efficient solutions found by the ITSGA is 4 with the highest value of 120.

5.5.2 SSD-SSW comparison. Efficient frontier analyses are used to compare the performance of the problems generated help characterize some properties of optimal treatment protocols. For this analysis, solutions of the SSD were found using total enumeration and solutions of the SSW were estimated using the ITSGA.

A few steps were taken to transform the solutions of the SSD for comparison to the SSW. Recall that SSD solutions are represented as x_{ij} and take on binary values. In the instance where a patient group l contained diagnosis j and x_{ij} equaled one, the corresponding x_{il} was set to one for treatment i. Otherwise, x_{il} was set to zero for treatment i. Afterwards, the transformed variable was substituted into the objective function for the SSW to find the comparable objective function value.

Three measures were used to assess the performance of the SSD model relative to the SSW: average improvement in efficacy, average improvement in cost, and average error rate. The average improvement in efficacy and average improvement in cost are based on the area of opportunity between the frontier of the SSD and the SSW.

Table 5.3 lists the average performance results of the SSD relative to the SSW by experimental factor. The most outstanding property shown in comparing model solutions the SSW outperforms the SUD across all performance measures. Also, the SUD tends to perform closer to the SSD when there is low variance in problem factors, with the exception variable cost

between treatments. Improvements in efficacy from the SUD are minimal and appear to have little sensitivity with regard to changes in factor levels. However, improvements in cost are relatively substantial and do show sensitivity to most study factors. Most of the error in the performance of the SUD is attributed to the difference in total cost evaluation. Overall, the average improvement in cost, improvement in efficacy, and error rate are 17%, 3%, and 47%, respectively.

Table 5.3

Average performance results of the SSD relative to the SSW by experimental factor.

Factor Levels	Improvement in Cost	Improvement in Efficacy	Average Error Rate	Undefined for Cost	Undefined for Efficacy
Variance in Perceived Efficacy between Providers					
High	17%	3%	47%	3.2%	3.1%
Low	17%	3%	47%	5.2%	4.7%
Variance in Patient Volumes between Providers					
High	17%	3%	47%	4.7%	4.5%
Low	17%	3%	47%	3.8%	3.3%
Variance in Fixed Costs between Treatments					
High	17%	3%	47%	4.2%	3.8%
Low	17%	3%	47%	4.2%	4.1%
Variance in Variable Costs between Treatments					
High	19%	2%	48%	2.1%	1.6%
Low	15%	4%	46%	6.3%	6.2%
Volume Discounts					
High	13%	2%	43%	5.5%	4.8%
Low	21%	4%	51%	3.0%	3.0%
Variance in Severity Between Diseases					
High	18%	4%	48%	4.3%	4.0%
Low	16%	3%	46%	4.1%	3.8%
Efficacy Curvature					
Concave	20%	3%	62%	3.6%	3.6%
Linear	10%	1%	35%	4.2%	4.2%
Convex	10%	2%	33%	4.7%	4.7%
Random	27%	6%	59%	4.4%	3.1%

5.5.2.1 Provider variances. The results indicate that the performance of the SSD is not sensitive to the variance in perceived efficacy between providers. This outcome is intuitive because the efficacy variable for both problems is the same. However, the methods to evaluate the total efficacy for each problem differ. It is expected that performance will be sensitive to variables that differ between the two problems (i.e. variance in severity between diseases and degree of efficacy curvature). Across both levels of this factor, the average improvement in cost, improvement in efficacy, and error rate are the same as the overall values: 17%, 3%, and 47%.

5.5.2.2 Patient volumes. It was expected that there would be sensitivity to this factor, based on study 2 of cost integration. In the second study, there is a negligible increase for the average improvement in cost, average improvement in efficacy, and average error rate. The same is shown in this study. The average improvement in cost, improvement in efficacy, and error rate are 17%, 3%, and 47%, respectively, regardless of the factor level.

5.5.2.3 Fixed costs. The total fixed costs for both problems are calculated using the same factors. Therefore, it can be expected that there is little sensitivity to changes in fixed cost when comparing the SSD and SSW. The average improvement in cost, improvement in efficacy, and error rate are 17%, 3%, and 47% respectively.

5.5.2.4 Variable costs. Unlike fixed costs, variable costs are related to treatment efficacies. Therefore, if the efficacy of a treatment increases, the variable cost of the treatment is also expected to increase. When patient volumes are constant, the efficient frontiers for the SSD and SSW will make proportionate shifts as shown the first study of this research project. However, study results show greater improvements in cost (by 4 percentage points) when the variance in variable costs between treatments is classified as high. The average error rate increases slightly from 46% to 48% when variance in variable cost changes from low to high.

Contrarily, the average improvement in efficacy decreases slightly by 2 percentage points. To determine the cause for this decreased performance, the number of qualifying comparisons was evaluated. In the instances where the efficient frontiers for cost or efficacy do not lie within the same regions for integration frontiers are not compared improvements in cost or efficacy are considered undefined. The percentage in undefined comparisons for cost and efficacy is 6% and 2%, respectively, when the variance in the variable cost is low and high, a difference of 4 percentage points. This phenomenon implies that the ranges of cost and efficacy for the SSW are so far improved over the SSD that to two problems are more often incompatible when the variance in variable costs is low. Across all other factors, the average difference in the lowest and highest percentage of undefined comparisons is only 0.4 percentage points.

5.5.2.5 Volume discounts. The SSW outperforms the SSD across all measures and factor levels for volume discounts but the greatest improvement occurs when the degree of volume discounts is classified as low. The average improvement in cost is increased from 13% to 21%, the average improvement in efficacy is slightly increased from 2% to 4%, and the average error rate is increased from 43% to 51% when the system incorporates a lower level of volume discounts.

As discuss with variable costs, it was initially expected that the improvement at lower levels of discounts occurred because of undefined comparisons. However, for this factor, the number of undefined comparisons is higher when volume discounts are high, by 2.5 and 1.8 percentage points for cost and efficacy, respectively. Upon further exploration of this factor, it was found that 192 of 1210 defined comparisons (16%) with high volume discounts have one-point solutions where only 70 of 1242 defined comparisons (6%) with low volume discounts result in one-point solutions. This phenomenon indicates that there are more instances where the

system will select only one efficient treatment protocol when volume discounts are high. This conclusion confirms concepts presented in Section 3.3.1. As variable costs are increased due to volume discounts, the number of efficient points along an efficient frontier are decreased. Since there are fewer efficient solutions to pick from when volume discounts are high, it is more likely that efficient solutions are matched between the SSD and SSW.

5.5.2.6 Disease severity. The SSW shows greater improvement over the SSD for all measures when the variance in severity between diseases is classified as high. The average improvement in cost increases from 16% to 18%, the average improvement in efficacy increases from 3% to 4%, and the average error rate increases from 46% to 48% as the variance in severity between diseases changes from low to high. As previously mentioned, the SSD does not account for this factor. The SSW model is better informed and, therefore, can better manage system variance for this factor.

5.5.2.7 Efficacy curvature. Recall that concave efficacy curvature indicates that a treatment provides the greatest improvement in efficacy at low utility values. Therefore, minor changes for from low treatment utilities can result in larger changes in efficacy than otherwise. The results from this study indicate that the SSW has high improvement over the SSD when the efficacy curvature is classified as concave. The performance of the SSW is even higher when the curvature is random or a mixture of concave, linear, and convex. Over all of the factors in this analysis, improvements in cost efficacy are more prevalent in the SSW when efficacy curvature is random.

5.6 Integration Impact

There are many practical implications granted from the outcomes of this research. From the onset of this research, there were no assumptions about how a random relationship between

efficacy and volume may impact performance. It is understood that the relationship between treatment efficacy and treatment volumes can be more random than linear, concave, or convex and definitely vary between patient types in practice. The results support integration regardless of efficacy and volume relationships. However, it is shown that high performance gains exists for a system that experiences this type of random efficacy relationship. This outcome has promising implications for integration where treatment efficacies are more complex.

Studies in Chapters 3 and 4 show how the impact of integration can change with varying levels of perceived efficacy, patient volume, and fixed costs. Although the factor level impact is not shown here, given the progressive development of integration policies, previous outcomes are still implied here. This study advances those findings to exhibit integration advantages in a system that is much more complex than the IUD and SUD, as well as the SSD.

Here the use of evidence-based policies reinforces consistency in decision making such that cost and efficacy become more optimal. Strech et al. (2010) supports this finding. In general, regardless of factor levels, this study supports integration toward having a full view of patients across the system. It is further illustrated in the instances of high variance in variable costs as well as disease severities; consolidated and data-driven decision making processes lead to better performance than otherwise. This is particularly insightful and useful to an agency like the DLA that supports a system where the diagnoses of patients can range from minimal to fatal requiring supplies as inexpensive as a small bandage to capital expenditures like CT scan machines. Given the complexities of the defense healthcare system, study outcomes support efficacy integration.

5.7 Chapter Summary

The study discussed in this chapter is unique in modeling and assessing a third study of integration for healthcare supply chain management. The study examined considers the implications of a more holistic approach to patient care where visibility can inform cost and efficacy. A custom GA is proposed as a solution technique to the more complex problem of treatment protocol selection. Many of the premises noted in previous chapters are confirmed, illustrating that integration can improve of overall efficiency for simple and complex system structures.

CHAPTER 6

Conclusions and Future Research

6.1 Research Summary

Based on existing gaps in healthcare treatment selection protocols or the lack of, it was expected from the onset of this dissertation research that decision support system integration can improve healthcare systems performance. This research accessed various systems factors that can impact treatment selection performance in order to determine how modifying behaviors can improve efficiency. Specifically, physician bias, patient volumes, leveraging economies of scale or costing structures, and complex treatment efficacy calculations were evaluated by modeling three studies of integration. The results indicate that more integrated treatment selection protocols lead to decreases in cost alongside increases in efficacy. Additionally, this dissertation explained how more complex healthcare systems or systems with higher variability in performance factors have the highest opportunity for improvement.

Chapter 2 of this research explains provider integration where shared decision making is enabled among physicians to mitigate physician bias and exploit physician expertise. This form of integration can lead to greater cost improvements for systems with higher variance in perceive treatment efficacies between physicians. It has been previously determined that evidence based treatment selection protocols lead to improved efficacies but this study extends to illustrate how evidence based treatment selection also improves costs. Enabling cost awareness and consensus among providers on best practices can ensure that the most effective treatments are selected for a given cost.

Cost improvements are also greater in instances of high variance in patient volumes between providers. There are a few factors that influenced this outcome. The first study of

integration models system costs without regard for entry costs. Therefore, the system is less likely to avoid incurring entry costs in selecting treatments. This behavior led to lower performance where the variances in fixed costs between treatments were highest. In determining system efficacy values, there is also value added in weighing provider opinions based on the volume of patients treated for a particular diagnosis or diagnoses. Providers with relatively little experience in treating a diagnosis will have less influence on system efficiency as they encounter fewer patients. This aligns with practices of the Veterans Affairs (VA) Health Care system that incentivizes specialized care. Counter to this VA practice, newer legislation from the Affordable Care Act encourages civilian healthcare providers to become more inclusive and provide more generalized patient treatment. It is expected that generalized patient care will centralize practices so that patients can avoid transferring and issues associated with transferring between providers (such as incomplete transfers of patient medical history, returns, and travel cost). In either system, with specialized or generalized care, providing consensus for treatment efficacy can improve overall efficacy as well as cost.

Chapter 3 discusses cost integration. The study in this chapter extends on observations in Chapter 2 to explore the impact of leveraging economies of scale or exploiting opportunities for volume discounts. It is found that sub linear relationships between volume and cost (such that incremental costs decrease as patient demand volumes increase) encourage decision makers to avoid entry costs. As higher entry costs are most avoided, integration allows greatest cost improvements where the variance in fixed costs between treatments is high. Cost are also greatly improved in instances with high variance in variable cost between treatments. In systems with limited visibility, providers are not fully capable of recognizing opportunities for discounted cost where treatments have higher patient demands. Whereas, an integrated system can aid providers

in trending patient volumes and negotiating costs with suppliers. Overall, cost and efficacy is improved when treatment selection protocols consider volume discounts.

For the final study, discussed in Chapter 4, treatment selection behaviors of aforementioned chapters were incorporated in customized a genetic algorithm to solve the more complex treatment selection problem. Namely, the observation that providers with greater treatment volumes will have more influence on selection and that efficient treatment selections tend to avoid entry costs. The final model observed in this study extends on the previously studies MOOPs to account for a more complex treatment efficacy calculation. The observed treatment efficacy structures encourage the use of decision support systems with advance analytics especially where treatment efficacy and volumes random or super linear. The random relationship led to the highest improvements in cost and efficacy upon enhanced integration as treatment efficacies where now selected in a more informed manner. The super linear relationship led to the second highest improvements in cost and also had an impact on how selection decisions change with volume discounts. Treatment selection in instances with low volume discounts outperformed systems with high volume discounts for cost and efficacy (by eight and two percentage points, respectively). This phenomenon occurs according to a concept presented in Section 3.3.1. As variable costs are increased due to volume discounts, the number of efficient points along an efficient frontier are decreased. Since there are fewer efficient solutions to pick from when volume discounts are high, it is more likely that efficient solutions are matched.

6.2 Research Contributions

In all, this research provides contributions to industrial and systems engineering that can be adapted for various applications although the research focuses on healthcare and the design of

treatment selection protocols. Strategies for efficient treatment selection protocols and system evolution are investigated. Definition is provided for key parameters to model treatment protocols under varying operational strategies along with defining three key types of integration for holistic patient care. A novel comparative analysis method is employed to measure the impact of multi-objective decision making. Custom heuristics, including a genetic algorithm, are developed to solve for complex treatment selection problems.

6.3 Future Research

The three studies explored through chapters 2, 3, and 4 of this dissertation provide evidence in support of centralized or shared decision making in medical treatment protocol selection. Future research will provide evidence of system behaviors based on supply chain operations outside of the healthcare industry. In addition to applying real data, future studies will attempt to improve optimization routines and survey integration teams for best practices. The results of this study will be extended to create a requirements analysis for a supporting information technology infrastructure.

References

Ahuja, R. K., Kumar, A., Jha, K., & Orlin, J. B. (2003). *Exact and Heuristic Methods for the Weapon Target Assignment Problem*. Working Paper 4464-03. MIT Sloan School of Management. Retrieved from http://ssrn.com/abstract=489802.

Angst, C. M., Devaraj, S., Queenan, C. C., & Greenwood, B. (2011). Performance Effects Related to the Sequence of Integration of Healthcare Technologies. *Production and Operations Management, 20*(3), 319-333. doi: 10.1111/J.1937-5956.2011.01218.X.

Blum, A., & Miller, L. G. (1993). Physician awareness of prescription drug costs: a missing element of drug advertising and promotion. *Journal of Family Practice, 36*(1), 33-39.

Carlson, R. C., & Nemhauser, G. L. (1966). Scheduling to minimize cost. *Operations Research, 14*, 52-58.

Commander, C. W. (2005). A Survey of the Quadratic Assignment Problem, with Applications. *Morehead Electronic Journal of Applicable Mathematics*(4), 1-15.

Darling, M. M., & Wise, S. S. (2010). Not your father's supply chain. Following best practices to manage inventory can help you save big. *Materials management in health care, 19*(4), 30-30-33.

Day, R. H. (1985). Allocating weapons to target complexes by means of nonlinear programming. *Operations Research, 14*, 992-1013.

Eckler, A. R., & Burr, S. A. (1972). Mathematical models of target coverage and missile allocation. Alexandria, VA: Military Operations Research Society.

Eiselt, H. A., & Sandblom, C.-L. (2000). *Integer programming and network models*. New York, NY: Springer.

Elshafei, A. N. (1977). Hospital Layout as a Quadratic Assignment Problem. *Operations Research Quarterly, 28*, 167-179.

Ernst, M. E., Kelly, M. W., Hoehns, J. D., Swegle, J. M., Buys, L. M., Logemann, C. D., . . . Pretorius, R. W. (2000). Prescription Medication Costs: A Study of Physician Familiarity. *Archives of Family Medicine, 9*(NOV/DEC), 1002-1007.

Finke, G., Burkard, R. E., & Rendl, F. (1987). Quadratic Assignment Problems. *Annals of Discrete Mathematics, 31*, 61-82.

Fitzsimmons, J., & Fitzsimmons, M. (2008). *Service Management: Operations, Strategy, Information Technology* (6 ed.). New York, NY: McGraw-Hill/Irwin.

Fung, C. H., Lim, Y.-W., Mattke, S., Damberg, C., & Shekelle, P. (2008). Systematic Review: The Evidence That Publishing Patient Care Performance Data Improves Quality of Care. *Annals of Internal Medicine 148*, 111-123.

Geoffrion, A. M., & Graves, G. W. (1976). Scheduling Parallel Production Lines with Changeover Costs: Practical Applications of a Quadratic Assignment/LP Approach. *Operations Research, 24*, 595-610.

Goemans, M. X. (2009). Lecture notes on bipartite matching *18.433: Combinatorial Optimization*: Massachusetts Institute of Technology.

Gupta, M., & Orbe, M. (2009). Revise the supply chain to reduce costs, experts say Retrieved November 2, 2011, from http://news.medill.northwestern.edu/chicago/news.aspx?id=151419.

Hannan, E. L. E., O'Donnell, J. J., & Freedland, T. T. (1981). A priority assignment model for standards and conditions in a long term care survey. *Socio-economic planning sciences, 15*(6), 277-277-289.

Hollingsworth, B. (2003). Non-Parametric and Parametric Applications Measuring Efficiency in Health Care. *Health Care Management Science, 6*, 203-218.

. The Integration of Innovation and Clinical Need: ROi - the Mercy Supply Chain Story. (2002). Chesterfield, MO: Resource Optimization and Innovation.

Kerno, S. (2008). The efficient frontier [Electronic version]. *Mechanical Engineering Magazine*. Retrieved from http://www.memagazine.org/contents/current/features/efficient/efficient.html.

Koopmans, T. C., & Beckmann, M. (1957). Assignment Problems and the Location of Economic Activities. *Econometrica, 25*(1), 53-76.

Korn, L. M., Reichert, S., Simon, T., & Halm, E. A. (2003). Improving Physicians' Knowledge of the Costs of Common Medications and Willingness to Consider Costs When Prescribing. *Journal of General Internal Medicine, 18*, 31-37.

Levin, M. S. (2004) *"Design of Systems: Structural Approach"* (Lecture 30 compressed ed.): Moscow Institution of Physics and Technology.

Lusk, E. J. E. (1979). Analysis of hospital capital decision alternatives: a priority assignment model. *The Journal of the Operational Research Society, 30*(5), 439-439-448.

Markowitz, H. M. (1959). *Portfolio Selection: Efficient Diversification of Investments*: Basil Blackwell.

McDermott, C., & Stock, G. N. (2007). Hospital operations and length of stay performance. [Research paper]. *International Journal of Operations & Production Management, 27*(9), 1020 - 1042. doi: 10.1108/01443570710775847.

McGlynn, E. A., Asch, S. M., Adams, J., Keesey, J., Hicks, J., DeCristofaro, A., & Kerr, E. A. (2003). The Quality of Health Care Delivered to Adults in the United States. *New England Journal of Medicine, 348*(26), 2635-2645. doi: doi:10.1056/NEJMsa022615.

Mitton, C., & Donaldson, C. (2004). Health care priority setting: principles, practice and challenges. *Cost Effectiveness and Resource Allocation, 2*(3), 1-8.

Ozkarahan, I. I. (1991). A disaggregation model of a flexible nurse scheduling support system. *Socio-economic planning sciences, 25*(1), 9-9-26.

Pate, C., Chin, D., Lutz, S., Beard, N., Isgur, B., Carlos, R., . . . Young, B. (2008). The price of excess: Identifying waste in healthcare spending: PricewaterhouseCoopers' Health Research Institute.

Reichert, S., Simon, T., & Halm, E. A. (2000). Physicians' Attitudes About Prescribing and Knowledge of the Costs of Common Medications. *Arch Intern Med. , 160*, 2799-2803.

Sahni, S., & Gonzalez, T. (1976). P-Complete Aproximation Problems. *Journal of the ACM, 23*, 555-565.

Smith, P. C., Mossialos, E., & Papanicolas, I. (2008). *Performance measurement for health system improvement: experiences, challenges and prospects*. Paper presented at the WHO European Ministerial Conference on Health Systems, Tallinn, Estonia.

Spertus, J. A., Radford, M. J., Every, N. R., Ellerbeck, E. F., Peterson, E. D., & Krumholz, H. M. (2003). *Challenges and Opportunities in Quantifying the Quality of Care for Acute Myocardial Infarction: Summary From the Acute Myocardial Infarction Working Group of the American Heart Association/American College of Cardiology First Scientific Forum on Quality of Care and Outcomes Research in Cardiovascular Disease and*

Stroke. Paper presented at the American Heart Association/American College of Cardiology Conference. http://circ.ahajournals.org/content/107/12/1681.

Tan, P.-N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining*. Boston, MA: Pearson Education, Inc.

Terry, K. (2010). Health Spending Hits 17.3 Percent of GDP In Largest Annual Jump. Retrieved from http://www.cbsnews.com/8301-505123_162-43841117/health-spending-hits-173-percent-of-gdp-in-largest-annual-jump/#ixzz1fxEaPt4Q.

Wang-Rodriguez, J. J., Mannino, F. L. F., Liu, E. E., & Lane, T. A. T. (1996). A novel strategy to limit blood donor exposure and blood waste in multiply transfused premature infants. *Transfusion, 36*(1), 64-64-70.

Werner, R. M., & Bradlow, E. T. (2006). Relationship Between Medicare's Hospital Compare Performance Measures and Mortality Rates. *JAMA, 296*, 2694-2702.

Winston, W. L., & Goldberg, J. B. (2004). *Operations research: applications and algorithms* (4 ed.): Cengage Learning.

Appendix

Procedures to Generate Problem Parameters

```
function [eijk,eij,ci] = ...
   study1_Coefficients(number_of_treatments, number_of_diseases,...
   number_of_providers, design)

% The number of all possible patient types is generated
number_of_patient_types = 0;
for j=1:number_of_diseases
   number_of_patient_types = number_of_patient_types + ...
      factorial(number_of_diseases)/(factorial(number_of_diseases-j)...
      *factorial(j));
end

%% ------------------------- Parameters -------------------------------
% This section of code defines and initiates values of all parameters. The
% following functions generate appropriate random values for parameters
% based on parameter indices and level in the design of the experiment.
% --------------------------------------------------------------------------

% di    fixed cost for selecting treatment i
di = generate_di(number_of_treatments, design(2));

% eijk      the perceived value of efficacy for treatment i for diagnosis j...
      % by provider k
eijk = generate_eijk(number_of_treatments, number_of_diseases,...
   number_of_providers, design(3));

% vlk       volume of patients group l by provider k
vlk = generate_vlk(number_of_patient_types,...
   number_of_providers, design(5));

%% ----------------------- End Parameters -------------------------------

%% ---------------------- Dependent Variables ---------------------------
% This section of code defines and initiates values that are dependent on
% the problem parameters based on parameter indices.
% --------------------------------------------------------------------------

%Volume of patient groups
   % vjk    volume of patient with disease j for provider k
   vjk = zeros(number_of_diseases, number_of_providers);
   for k = 1:number_of_providers
      vjk(:,k)=diseasecombos(number_of_diseases,vlk(:,k));
   end

   %vk volume of patients per provider
   vk = sum(vjk,1);

%Evidence based efficacy
   % eij     estimated efficacy for treatment i and diagnosis j
   eij=zeros(number_of_treatments,number_of_diseases);
   for i = 1: number_of_treatments
      for j = 1:number_of_diseases
         for k = 1:number_of_providers
            eij(i,j)=eij(i,j)+vjk(j,k)*eijk(i,j,k);
         end
         x=sum(vjk,2);
         eij(i,j)=eij(i,j)/x(j);
      end
   end

%Efficacy related cost coefficient
   % si efficacy related variable cost coefficient for treatment i
si = generate_si(eij,number_of_treatments, design(4));
```

```
%Unit cost coefficient without consideration for volume or discounts
   ci = di + si;
%% ---------- End Dependent Variables ----------

function [ui,vj,eij,di,si] = ...
   study2_Coefficients(number_of_treatments, number_of_diseases,...
   number_of_providers, design)


number_of_patient_types = 0;
for j=1:number_of_diseases
   number_of_patient_types = number_of_patient_types + ...
      factorial(number_of_diseases)/(factorial(number_of_diseases-j)...
      *factorial(j));
end

%% ---------- Parameters ----------
% This section of code defines and initiates values of all parameters
% --------------------------------------------------------------------

% di    fixed cost for selecting treatment i
di = generate_di(number_of_treatments, design(2));

% eijk      the perceived value of efficacy for treatment i for diagnosis j...
      % by provider k
eijk = generate_eijk(number_of_treatments, number_of_diseases,...
   number_of_providers, design(3));

% ui        degree of volume discount for the cost of treatment i
ui = generate_ui(number_of_treatments, design(5));

% vlk       volume of patients group l by provider k
vlk = generate_vlk(number_of_patient_types,...
   number_of_providers, design(6));

%% ---------- End Parameters ----------

%% ---------- Dependent Variables ----------
% This section of code defines and initiates values that are dependent on
% the problem parameters
% --------------------------------------------------------------------

%Volume of patient groups
   % vjk    volume of patient with disease j for provider k
   vjk = zeros(number_of_diseases, number_of_providers);
   for k = 1:number_of_providers
      vjk(:,k)=diseasecombos(number_of_diseases,vlk(:,k));
   end

   vj = sum(vjk,2);

%Evidence based efficacy
   % eij     estimated efficacy for treatment i and diagnosis j
   eij=zeros(number_of_treatments,number_of_diseases);
   for i = 1: number_of_treatments
      for j = 1:number_of_diseases
         for k = 1:number_of_providers
            eij(i,j)=eij(i,j)+vjk(j,k)*eijk(i,j,k);
         end
         x=sum(vjk,2);
         eij(i,j)=eij(i,j)/x(j);
      end
   end

%Efficacy related cost coefficient
   % si efficacy related variable cost coefficient for treatment i
   si = generate_si(eij,number_of_treatments, design(4));

%% ---------- End Dependent Variables ----------
```

```
function [eijk,gj,ui,vlk,wil,vjk,eij,ci,di,si,Gl,d] = ...
    study3_Coefficients(number_of_treatments, number_of_diseases,...
    number_of_providers, design)

number_of_patient_types = 0;
for j=1:number_of_diseases
    number_of_patient_types = number_of_patient_types + ...
        factorial(number_of_diseases)/(factorial(number_of_diseases-j)...
        *factorial(j));
end

%% ---------------------------- Parameters --------------------------------
% This section of code defines and initiates values of all parameters
% ------------------------------------------------------------------------

% di    fixed cost for selecting treatment i
di = generate_di(number_of_treatments, design(2));

% eijk      the perceived value of efficacy for treatment i for diagnosis j...
        % by provider k
eijk = generate_eijk(number_of_treatments, number_of_diseases,...
    number_of_providers, design(3));

% gj        severity diagnosis j for patient group
gj = generate_gj(number_of_diseases, design(4));

% ui        degree of supply chain cost function for treatment i
ui = generate_ui(number_of_treatments, design(6));

% vlk       volume of patients group l by provider k
vlk = generate_vlk(number_of_patient_types,...
    number_of_providers, design(7));

% wil       degree of efficacy function treatment i for patient group l
wil = generate_wil(number_of_treatments,...
    number_of_patient_types, design(8));
%% ----------------------- End Parameters --------------------------------

%% ---------------------- Dependent Variables ----------------------------
% This section of code defines and initiates values that are dependent on
% the problem parameters
% ------------------------------------------------------------------------

%Volume of patient groups
    % vjk    volume of patient with disease j for provider k
    vjk = zeros(number_of_diseases, number_of_providers);
    for k = 1:number_of_providers
        vjk(:,k)=diseasecombos(number_of_diseases,vlk(:,k));
    end

%Evidence based efficacy
    % eij    estimated efficacy for treatment i and diagnosis j
    eij=zeros(number_of_treatments,number_of_diseases);
    for i = 1: number_of_treatments
        for j = 1:number_of_diseases
            for k = 1:number_of_providers
                eij(i,j)=eij(i,j)+vjk(j,k)*eijk(i,j,k);
            end
            x=sum(vjk,2);
            eij(i,j)=eij(i,j)/x(j);
        end
    end


%Efficacy related cost coefficient
    % si efficacy related variable cost coefficient for treatment i
    si = generate_si(eij,number_of_treatments, design(5));

    ci = di + si;
```

```
%Patient groups and severity
   Gl=ones(number_of_patient_types,1);
   d=zeros(number_of_patient_types,number_of_diseases);
   r=0;
   for j=1:number_of_diseases
      A=combinator(number_of_diseases,j,'c');
      [rows,columns]=size(A);
      for l=1:factorial(number_of_diseases)/...
            (factorial(number_of_diseases-j)*factorial(j))
         for m=1:rows
            for k=1:columns
               d(r+m,k)=A(m,k);
            end
         end
      end
      A=[];
      r=r+rows;
   end

   countjl=zeros(number_of_diseases,number_of_patient_types);
   for l=1:number_of_patient_types
      for j=1:number_of_diseases
         if find(d(l,:)==j)>0
            Gl(l) = Gl(l)*(10-gj(j));
            countjl(j,l)=1;
         end
      end
      x=sum(countjl,1);
      Gl(l) = (10^x(l)-Gl(l))/(10^(x(l)-1));
   end
%% -------------------- End Dependent Variables ---------------------------

function di = generate_di(number_of_treatments, factor_level)

di = zeros(number_of_treatments,1);

if factor_level == 1
   A = 1;
   B = 1;
   [M,V] = betastat(A,B);
   for r = 1:1000
      x =  betarnd(A,B,number_of_treatments,1);
      if var(100*x) >= V*100^2
         di(:) = 100*x;
         break
      end
   end
else
   A = 6;
   B = 6;
   [M,V] = betastat(A,B);
   for r = 1:1000
      x =  betarnd(A,B,number_of_treatments,1);
      if var(100*x) <= V*100^2
         di(:) = 100*x;
         break
      end
   end
end

function eijk = generate_eijk(number_of_treatments, number_of_diseases,...
   number_of_providers, factor_level)

eijk = zeros(number_of_treatments, number_of_diseases,...
   number_of_providers);

for i = 1:number_of_treatments
   for j = 1:number_of_diseases
      if factor_level == 1
         A = 1;
```

```
            B = 1;
            [M,V] = betastat(A,B);
            for r = 1:1000
                x = betarnd(A,B,1,number_of_providers);
                if var(x) >= V
                    eijk(i,j,:) = x;
                    break
                end
            end
        else
            A = 6;
            B = 6;
            [M,V] = betastat(A,B);
            for r = 1:1000
                x = betarnd(A,B,1,number_of_providers);
                if var(x) <= V
                    eijk(i,j,:) = x;
                    break
                end
            end
        end
    end
end

function gj = generate_gj(number_of_diseases, factor_level)

gj = zeros(1,number_of_diseases);

if factor_level == 1
    A = 1;
    B = 1;
    [M,V] = betastat(A,B);
    for r = 1:1000
        x =  betarnd(A,B,1,number_of_diseases);
        if var(10*x) >= V*10^2
            gj(:) = 10*x;
            break
        end
    end
else
    A = 6;
    B = 6;
    [M,V] = betastat(A,B);
    for r = 1:1000
        x =  betarnd(A,B,1,number_of_diseases);
        if var(10*x) <= V*10^2
            gj(:) = 10*x;
            break
        end
    end
end

function si = generate_si(eij,number_of_treatments, factor_level)

si = zeros(number_of_treatments,1);
rgn = zeros(number_of_treatments,1);
ei = sum(eij,2);

if factor_level == 1
    A = 1;
    B = 1;
    [M,V] = betastat(A,B);
    for r = 1:1000
        x =  betarnd(A,B,number_of_treatments,1);
        for i = 1:number_of_treatments
            rgn(i) = 200*x(i)*ei(i)/sum(ei)...
                +(-25+50.*rand())*x(i)*ei(i)/sum(ei);
        end
        if var(rgn) > V*(100)^2
            si = rgn;
```

```
            break
        end
    end
else
    A = 6;
    B = 6;
    [M,V] = betastat(A,B);
    for r = 1:1000
        x =  betarnd(A,B,number_of_treatments,1);
        for i = 1:number_of_treatments
            rgn(i) = 200*x(i)*ei(i)/sum(ei)...
                +(-10+20.*rand())*x(i)*ei(i)/sum(ei);
        end
        if var(rgn) <= V*(100)^2
            si = rgn;
            break
        end
    end
end

function ui = generate_ui(number_of_treatments, factor_level)

if factor_level == 2
    ui = ones(number_of_treatments,1);
else
    ui = 0.5*ones(number_of_treatments,1);
end

function vlk = generate_vlk(number_of_patient_types,...
    number_of_providers, factor_level)

vlk = zeros(number_of_patient_types,number_of_providers);

for j = 1:number_of_patient_types
    if factor_level == 1
        A = 1;
        B = 1;
        [M,V] = betastat(A,B);
        for r = 1:1000
            x = betarnd(A,B,1,number_of_providers);
            if var(100*x) >= V*100^2
                vlk(j,:) = 100*x;
                break
            end
        end
    else
        A = 6;
        B = 6;
        [M,V] = betastat(A,B);
        for r = 1:1000
            x = betarnd(A,B,1,number_of_providers);
            if var(100*x) <= V*100^2
                vlk(j,:) = 100*x;
                break
            end
        end
    end
end

function wil = generate_wil(number_of_treatments,...
    number_of_patient_types, factor_level)

for l = 1:number_of_patient_types
    if factor_level == 1
        wil = 0.5*ones(number_of_treatments,number_of_patient_types);
    elseif factor_level == 2
        wil = ones(number_of_treatments,number_of_patient_types);
    elseif factor_level == 3
        wil = 2*ones(number_of_treatments,number_of_patient_types);
    else
```

```
      wil = 2*rand(number_of_treatments,number_of_patient_types);
   end
end
```

Procedures to generate A matrix for binary problems

```
function A = generateA(Number_of_treatments, Number_of_diseases)

A = zeros(Number_of_diseases, Number_of_diseases*Number_of_treatments);
r=0;

for j = 1:Number_of_diseases
   for i = 1:Number_of_treatments
      A(j,i+r)=1;
   end
   r=r+Number_of_treatments;
end
```

Procedures to generate Objective Function Value of Model 1

```
function z = ofv_Model1(xijk, eijk, ci)
%This function calculations the objective function value (ofv) for a given
%feasible solutions of Model 1 based on eijk and ci.

[Number_of_treatments,Number_of_diseases, Number_of_providers]=size(eijk);

% Initialize for two objectives
z = zeros(Number_of_providers,2);

for k = 1:Number_of_providers
   for i = 1:Number_of_treatments
      for j = 1:Number_of_diseases
         z(k,1) = z(k,1) + ci(i)*xijk(i,j,k);        %total cost
         z(k,2) = z(k,2) + eijk(i,j,k)*xijk(i,j,k);  %total efficacy
      end
   end
end
```

Procedures to generate Objective Function Value of Model 2

```
function z = ofv_Model2(xij, eij, ci)
%This function calculations the objective function value (ofv) for a given
%feasible solutions of Model 2 based on eij and ci.

% Initialize for two objectives
z = zeros(1,2);

[Number_of_treatments,Number_of_diseases]=size(eij);

for i = 1:Number_of_treatments
   for j = 1:Number_of_diseases
      z(1) = z(1) + ci(i)*xij(i,j);     %total cost
      z(2) = z(2) + eij(i,j)*xij(i,j);  %total efficacy
   end
end
```

Procedures to generate Objective Function Value of Model 3

```
function z = ofv_Model3(xij, di, eij, si, ui, vj)
%This function calculates the objective function value of Model one as
%cost z(1) and efficacy z(2) for decision xij given coefficients ci and eij
```

```
%Determing the number of treatments, diseases, and volume per treatment
Number_of_treatments = size(eij,1);
vi = xij*vj;

%Initialize objective function value as 1x2 vector
z=zeros(1,2);

%Sum total cost
for i = 1:Number_of_treatments
   if vi(i) > 0
      z(1) = z(1) + di(i) + si(i)*(vi(i))^ui(i);
   end
end

%Sum total efficacy
z(2) = sum(dot(eij,xij));
```

Procedures to generate Objective Function Value of Model 4

```
function y = ofv_Model4(xij, eij, di, si, ui, Vl, Gl, wil, d, gj)

[number_of_treatments, number_of_diseases] = size(eij);
number_of_patient_types = size(Gl);

% Initialize for two objectives
y = zeros(2,1);
% Convert xij to xil
xil = zeros(number_of_treatments,number_of_patient_types);
for l=1:number_of_patient_types
   for j=1:number_of_diseases
      if find(d(l,:)==j)>0
         for i=1:number_of_treatments
            if xij(i,j) == 1
               xil(i,l) = 1;
            end
         end
      end
   end
end

% Compute efficacy
trtinefficacy = ones(number_of_diseases,number_of_patient_types);
group_efficacy = zeros(1,number_of_patient_types);
Total_Efficacy = 0;
for l=1:number_of_patient_types
      for j=1:number_of_diseases
         if find(d(l,:)==j)>0
            for i=1:number_of_treatments
               trtinefficacy(j,l)=trtinefficacy(j,l)...
                  *(1-(eij(i,j)*xil(i,l))^wil(i,l));
            end
         group_efficacy(l)=group_efficacy(l)+gj(j)*(1-trtinefficacy(j,l))/sum(gj);
         end
      end
      Total_Efficacy = Total_Efficacy +(Gl(l)/sum(Gl))*(Vl(l)/sum(Vl))*group_efficacy(l);
end
y(2) = Total_Efficacy;

% Compute cost objective to minimize
FC = 0;
VC = 0;
trtvolume = zeros(1,number_of_treatments);
for i=1:number_of_treatments
   for l = 1:number_of_patient_types
      trtvolume(i) = trtvolume(i) + Vl(l)*xil(i,l);
   end
```

```
    if trtvolume(i) > 0
        FC = FC + di(i);
    end
    VC = VC + si(i)*(trtvolume(i)^ui(i));
end
y(1) = FC + VC;
```

Procedures to generate Study 1

```
function [dx,dy,ndomStartx,ndomEndx,domStartx,domEndx,...
    ndomStarty,ndomEndy,domStarty,domEndy,hits, misses, nglobalpts] = ...
    dissertation_Study1v2(number_of_treatments, number_of_diseases,...
    number_of_providers, number_of_replications)
%This program is designed to conduct a full factor experiment comparing the
%efficient frontier of model 1 to model 2 given the number of treatments,
%number of diseases, number of providers, and number of replications.

%Declare no. of levels for the four factors used in the experimental design
Factor1Levels = 2; %No. of levels for fixed cost variance
Factor2Levels = 2; %No. of levels for variance in percieved efficacy
Factor3Levels = 2; %No. of levels for variable cost variance
Factor4Levels = 2; %No. of levels for variance in provider patient volume

%Generate full factorial design matrix
dFF = fullfact([number_of_replications,Factor1Levels,Factor2Levels,...
    Factor3Levels,Factor4Levels]);

%Count the number of experiments included in design for loop performance
%measurements
number_of_experiments = size(dFF,1);

%Initialize output variable eMargin to store the errors of each
%provider's perceived efficient frontier.
dx = zeros(number_of_experiments,number_of_providers);
dy = zeros(number_of_experiments,number_of_providers);
ndomStartx = zeros(number_of_experiments,number_of_providers,2);
ndomEndx = zeros(number_of_experiments,number_of_providers,2);
domStartx = zeros(number_of_experiments,number_of_providers,2);
domEndx = zeros(number_of_experiments,number_of_providers,2);
ndomStarty = zeros(number_of_experiments,number_of_providers,2);
ndomEndy = zeros(number_of_experiments,number_of_providers,2);
domStarty = zeros(number_of_experiments,number_of_providers,2);
domEndy = zeros(number_of_experiments,number_of_providers,2);
hits = zeros(number_of_experiments,number_of_providers);
misses  = zeros(number_of_experiments,number_of_providers);
nglobalpts = zeros(number_of_experiments,1);

%Loop is used to determine providers' performance in each experiment
for experiment_number = 1:number_of_experiments

    %Generate coefficients for model 1/2 based on parameters and designID
    [eijk,eij,ci] = ...
        study1_Coefficients(number_of_treatments, number_of_diseases,...
        number_of_providers, dFF(experiment_number,:));

    %Generate all feasible solutions for model 2
    A = generateA(number_of_treatments, number_of_diseases);
    b = ones(1,number_of_diseases);
    model2BFSs = feassol(A,b);

    %Initialize decision variable
    xij = zeros(number_of_treatments, number_of_diseases);

    %Loop to determine objective function values of all feasible solutions
    number_of_BFSs = size(model2BFSs,2); %Number of basic feasible solns

    objectiveValue_Model1 = zeros(number_of_BFSs,2,number_of_providers);
```

```
   objectiveValue_Model2 = zeros(number_of_BFSs,2);

   for n = 1:number_of_BFSs

      for j = 1:number_of_diseases
         for i = 1:number_of_treatments
            xij(i,j) = model2BFSs(number_of_diseases*(j-1)+i,n);
         end
      end

      %Calculate the objective function value of an indexed solution for
      %model 2
      objectiveValue_Model2(n,:) = ofv_Model2(xij,eij,ci);


      %Calculate the objective function value of an indexed solution
      %using model 2 as if model 1 is a submodel for each provider.
      for k = 1:number_of_providers
         objectiveValue_Model1(n,:,k) = ofv_Model2(xij,eijk(:,:,k),ci);
      end

   end

   %Use objective funtion values of all feasible solutions to determine
   %efficient frontiers of providers in model 1 and model 2
   efficient_frontier1 = concaveFront(objectiveValue_Model2);

   for k = 1:number_of_providers

         [sval,solutionID] = concaveFront(objectiveValue_Model1(:,:,k));
         number_of_solutions = size(solutionID,1);
         provider_frontier = zeros(number_of_solutions,2);

         %determine provider k frontier in model 2
         for r = 1:number_of_solutions
            for j = 1:number_of_diseases
               for i = 1:number_of_treatments
                  xij(i,j) = model2BFSs(number_of_diseases*(j-1)+i,solutionID(r));
               end
            end
            provider_frontier(r,:) = ofv_Model2(xij,eij,ci);
         end

         %sort the provider frontier
         [d1,d2] = sort(provider_frontier(:,1));
         provider_frontier = provider_frontier(d2,:);

         [dx(experiment_number,k),...
         dy(experiment_number,k),...
         ndomStartx(experiment_number,k,:),...
         ndomEndx(experiment_number,k,:),...
         domStartx(experiment_number,k,:),...
         domEndx(experiment_number,k,:),...
         ndomStarty(experiment_number,k,:),...
         ndomEndy(experiment_number,k,:),...
         domStarty(experiment_number,k,:),...
         domEndy(experiment_number,k,:),...
         hits(experiment_number,k),...
         misses(experiment_number,k)]...
         = measure3(provider_frontier,efficient_frontier1);
   end
   nglobalpts(experiment_number) = size(efficient_frontier1,1);
end
```

Procedures to generate Study 2

```
function [dx,dy,ndomStartx,ndomEndx,domStartx,domEndx,...
```

```
    ndomStarty,ndomEndy,domStarty,domEndy,hits, misses, nglobalpts,...
    runtimes, enum_runtimes] = ...
    dissertation_Study2a_v2(number_of_treatments, number_of_diseases,...
    number_of_providers, number_of_replications)
% error = dissertation_Study2(number_of_treatments, number_of_diseases,...
%     number_of_providers, number_of_replications)
%This program is designed to conduct a full factor experiment comparing the
%efficient frontier from the heuristic of model 3 to the actual
%efficient frontier of model 3 given the number of treatments,
%number of diseases, number of providers, and number of replications.
format long g
%Declare no. of levels for the five factors used in the experimental design
Factor1Levels = 2; %No. of levels for fixed cost variance
Factor2Levels = 2; %No. of levels for variance in percieved efficacy
Factor3Levels = 2;  %No. of levels for variable cost variance
Factor4Levels = 2; %No. of levels for degree of volume discounts
Factor5Levels = 2; %No. of levels for variance in provider patient volume

%Generate full factorial design matrix
dFF = fullfact([number_of_replications,Factor1Levels,Factor2Levels,...
    Factor3Levels,Factor4Levels,Factor5Levels]);

%Count the number of experiments included in design for loop performance
%measurements
number_of_experiments = size(dFF,1);

%Initialize output variable eMargin to store the error in the heuristic
%efficient frontier.
% error = zeros(number_of_experiments,2);
dx = zeros(number_of_experiments,1);
dy = zeros(number_of_experiments,1);
ndomStartx = zeros(number_of_experiments,2);
ndomEndx = zeros(number_of_experiments,2);
domStartx = zeros(number_of_experiments,2);
domEndx = zeros(number_of_experiments,2);
ndomStarty = zeros(number_of_experiments,2);
ndomEndy = zeros(number_of_experiments,2);
domStarty = zeros(number_of_experiments,2);
domEndy = zeros(number_of_experiments,2);
hits = zeros(number_of_experiments,1);
misses  = zeros(number_of_experiments,1);
nglobalpts = zeros(number_of_experiments,1);
runtimes = zeros(number_of_experiments,1);
enum_runtimes = zeros(number_of_experiments,1);
% %Additional programming for scatterplot of Frontier
% figure('Name', 'HOLD ON approach');
% hold on

%Loop is used to determine heuristic performance in each experiment
for experiment_number = 1:number_of_experiments

    efficient_frontier3 = [];
    heuristic_frontier = [];

    %Generate coefficients for model 3 based on parameters and designID
    [ui,vj,eij,di,si] = ...
        study2_Coefficients(number_of_treatments, number_of_diseases,...
        number_of_providers, dFF(experiment_number,:));
    tic
    %Generate all feasible solutions for model 2
    A=generateA(number_of_treatments, number_of_diseases);
    b=ones(1,number_of_diseases);
    model3BFSs=feassol(A,b);

    %Initialize decision variable
    xij = zeros(number_of_treatments, number_of_diseases);

    %Loop to determine objective function values of all feasible solutions
    objectiveValue_Model3 = zeros(size(model3BFSs,2),2);
```

```
    for n = 1:size(model3BFSs,2)
        for j = 1:number_of_diseases
            for i = 1:number_of_treatments
                xij(i,j) = model3BFSs(number_of_diseases*(j-1)+i,n);
            end
        end

        %Calculate the objective function value of an indexed solution for
        %model 3
        objectiveValue_Model3(n,:) = ofv_Model3(xij, di, eij, si, ui, vj);

    end

    %Use objective funtion values of all feasible solutions to determine
    %efficient frontier of model 3
    efficient_frontier3 = concaveFront(objectiveValue_Model3);
    enum_runtimes(experiment_number) = toc;

    %Use heuristic to estimate the efficient frontier of model 3
    tic
    [heuristic3_frontier,heuristic3_solution] = ...
        model3_Hueristic(di, eij, si, ui, vj);
    runtimes(experiment_number) = toc;

    front1 = zeros(size(heuristic3_solution,1),2);
    for n = 1:size(heuristic3_solution,1)
        xij = xijStrtoMatrix(heuristic3_solution(n,:),number_of_treatments, number_of_diseases);
        front1(n,:)=ofv_Model3(xij, di, eij, si, ui, vj);
    end
    [d1,d2] = sort(front1(:,1));
    front1 = front1(d2,:);

    %Compare model 1 to model 2 and store error per provider
%       error(experiment_number,:) = eMargin(front1,efficient_frontier3);
    [dx(experiment_number),...
    dy(experiment_number),...
    ndomStartx(experiment_number,:),...
    ndomEndx(experiment_number,:),...
    domStartx(experiment_number,:),...
    domEndx(experiment_number,:),...
    ndomStarty(experiment_number,:),...
    ndomEndy(experiment_number,:),...
    domStarty(experiment_number,:),...
    domEndy(experiment_number,:),...
    hits(experiment_number),...
    misses(experiment_number)]...
    = measure3(front1,efficient_frontier3);
    nglobalpts(experiment_number) = size(efficient_frontier3,1);

end
```

Procedures to generate Model 3 Heuristic

```
function [Front, Solution, Distance, iteration] = ...
    model3_Hueristic(di, eij, si, ui, vj)

%STEP 0: Initialize heuristic parameters and solution
[number_of_treatments, number_of_diseases] = size(eij);
xijString0 = initializeXij(number_of_treatments, number_of_diseases);
Solution = xijString0;

%STEP 3: Repeat STEP 1-2 until the number of iterations is 100, previous
%solutions are the same a solutions of current iteration, or the distance
%between the previous front and the current front is less than epsilon.
iteration = 1;
Distance = 0;
N = number_of_treatments*number_of_diseases;
```

```
nVars = N;
interval = 0;

while iteration < 100
   %STEP 1: Find one-swap solutions with furthest distance from each
   %point in the current set of solutions that fit in regions 1, 2, and 3.
   swapFront = [];
   swapSolution = [];
   swapDistance = 0;

   for r = 1:size(Solution,1)
      [swapFront0, swapSolution0, swapDistance0] = ...
         model3_findSwaps(Solution(r,:), di, eij, si, ui, vj);
      swapFront = [swapFront; swapFront0];
      swapSolution = [swapSolution; swapSolution0];
      swapDistance = swapDistance + swapDistance0;
   end

   Front = swapFront;
   Solution = swapSolution;
   Distance = swapDistance;

   %Condition for stopping based on solution or distance
   if swapDistance > .5
      iteration = iteration + 1;
   else
      iteration = 100;
   end

   %STEP 2: Ensure that front is concave after N solutions have been
   %evaluated
   interval = interval + r;

   if interval >= nVars*5
      [concaveFront, concaveSolns] = evalFrontSlope(Front, Solution);
      interval = 0;
%        Front = [];
%        Solution = [];
      Front = concaveFront;
      Solution = concaveSolns;
   end

end

%STEP 4: Ensure that the final front is concave then STOP
if isempty(Front)== 0
   [concaveFront, concaveSolns] = evalFrontSlope(Front, Solution);
%           Front = [];
%        Solution = [];
   Front = concaveFront;
   Solution = concaveSolns;
end

function [swapFront, swapSolution, swapDistance] = ...
   model3_findSwaps(xijString, di, eij, si, ui, vj)
%The program determines if more optimal points exists for the objective
%function value of xijString or P0 and returns these points in swapFront
%and their solutions in swapSolution. Scalar swapDistance totals distances
%from P0 to the most optimal points.

%STEP 0: Initialize the objective function value of xijString and other
%code variables
[Number_of_treatments, Number_of_diseases] = size(eij);
swapFront = []; %Front for more improving swap decisions
swapSolution = []; %Solutions for more improving swap decisions
N = size(xijString,2);  %Number of solutions evaluated
xij0 = xijStrtoMatrix(xijString, Number_of_treatments, Number_of_diseases);
P0 = ofv_Model3(xij0, di, eij, si, ui, vj);
V0 = xij0*vj;
point = zeros(N,2); %Matrix to store ofv for the solutions being evaluated
```

```
trialXij = zeros(N);
for r = 1:N
   trialXij(r,:) = xijString ;%Matrix to store solutions being evaluated
end

%STEP1: Determine all possible swaps from P0
for i = 1:Number_of_treatments
   for j = 1:Number_of_diseases
      r = Number_of_diseases*(i-1)+j;
      xij = xijStrtoMatrix(trialXij(r,:),...
         Number_of_treatments, Number_of_diseases);
      xij(:,j) = 0;
      xij(i,j) = 1;
      trialXij(r,:) = xijMatrixtoStr(xij);
      Vi = xij*vj;
      P1 = swapIncrement(xij0, xij, di, eij, ...
         si, ui, V0, Vi);
      point(r,:)= P0 + P1;
   end
end

%STEP 2: Calculate the distances from P0 to all possible swaps from P0
alldistances = pdist([P0; point]); %Distances between all points
d = alldistances(1:N); %Distances from P0 to all possible swaps from P0

%STEP 3: Define points of furthest distance for swaps in regions one, two,
%and three
dtemp1 = 0; %Stores previous best distance evaluated in region 1, R1
dtemp2 = 0; %Stores previous best distance evaluated in region 2, R2
dtemp3 = 0; %Stores previous best distance evaluated in region 3, R3
mtemp1 = inf; %Stores previous best distance evaluated in region 1, R1
mtemp3 = 0; %Stores previous best distance evaluated in region 3, R3
region1Front = [];
region2Front = [];
region3Front = [];
for index = 1:N
   if P0(1) > point(index,1) && P0(2) > point(index,2) %R1 criteria
      m1 = (P0(2) - point(index,2))/(P0(1) - point(index,1));
      if m1 < mtemp1
         mtemp1 = m1;
         dtemp1 = d(index);
         region1Front = point(index,:);
         region1Solution = trialXij(index,:);
      end
   elseif  P0(1) >= point(index,1) && P0(2) <= point(index,2) %R2 criteria
      if d(index)> dtemp2
         dtemp2 = d(index);
         region2Front = point(index,:);
         region2Solution = trialXij(index,:);
      end
   elseif P0(1) < point(index,1) && P0(2) < point(index,2) %R3 criteria
      m3 = (point(index,2) - P0(2))/(point(index,1) - P0(1));
      if m3 > mtemp3
         mtemp3 = m3;
         dtemp3 = d(index);
         region3Front = point(index,:);
         region3Solution = trialXij(index,:);
      end
   end
end
% dtemp1 = 0; %Stores previous best distance evaluated in region 1, R1
% dtemp2 = 0; %Stores previous best distance evaluated in region 2, R2
% dtemp3 = 0; %Stores previous best distance evaluated in region 3, R3
% region1Front = [];
% region2Front = [];
% region3Front = [];
% for index = 1:N
%    if P0(1) > point(index,1) && P0(2) > point(index,2) %R1 criteria
%       if d(index)> dtemp1
```

```
%            dtemp1 = d(index);
%            region1Front = point(index,:);
%            region1Solution = trialXij(index,:);
%        end
%    elseif  P0(1) >= point(index,1) && P0(2) <= point(index,2) %R2 criteria
%        if d(index)> dtemp2
%            dtemp2 = d(index);
%            region2Front = point(index,:);
%            region2Solution = trialXij(index,:);
%        end
%    elseif P0(1) < point(index,1) && P0(2) < point(index,2) %R3 criteria
%        if d(index)> dtemp3
%            dtemp3 = d(index);
%            region3Front = point(index,:);
%            region3Solution = trialXij(index,:);
%        end
%    end
% end

%STEP 4: Define swapFront as best points that may exist in region one, two,
%and three which correspond to the points in swapSolution and sum their
%distances to P0.
if isempty(region1Front) == 0
   swapFront = [swapFront; region1Front];
   swapSolution = [swapSolution; region1Solution];
end

if isempty(region2Front) == 0
   swapFront = [swapFront; region2Front];
   swapSolution = [swapSolution; region2Solution];
end

if isempty(region3Front) == 0
   swapFront = [swapFront; region3Front];
   swapSolution = [swapSolution; region3Solution];
end

swapDistance = dtemp1 + dtemp2 + dtemp3; %Sums furthest distances
```

Procedures to generate Study 3

```
function [dx,dy,ndomStartx,ndomEndx,domStartx,domEndx,...
   ndomStarty,ndomEndy,domStarty,domEndy,run_time,...
   run_gens, run_exit, run_avgdist, run_spread, run_frontsize] =...
   dissertation_Study3(number_of_treatments, number_of_diseases,...
   number_of_providers, number_of_replications)
%This program is designed to conduct a full factor experiment comparing the
%efficient frontier of model 3 to model 4 given the number of
%treatments, number of diseases, number of providers, and number of
%replications.  measure3 is used to assess performance. Initial population
%of the GA is solution set from Model 3.

number_of_patient_types = 0;
for j=1:number_of_diseases
   number_of_patient_types = number_of_patient_types + ...
      factorial(number_of_diseases)/(factorial(number_of_diseases-j)...
      *factorial(j));
end

%Declare no. of levels for the five factors used in the experimental design
Factor1Levels = 2; %No. of levels for fixed cost variance
Factor2Levels = 2; %No. of levels for variance in percieved efficacy
Factor3Levels = 2; %No. of levels for variance disease severity
Factor4Levels = 2; %No. of levels for variable cost variance
Factor5Levels = 2; %No. of levels for degree of volume discounts
Factor6Levels = 2; %No. of levels for variance in provider patient volume
```

```
Factor7Levels = 4; %No. of levels for efficacy curvature

dFF = fullfact([number_of_replications,Factor1Levels,Factor2Levels,...
   Factor3Levels,Factor4Levels,Factor5Levels,Factor6Levels,...
   Factor7Levels]);

number_of_experiments = size(dFF,1);

%Initialize output variables to store the model distance measures
dx = zeros(number_of_experiments,number_of_providers,3);
dy = zeros(number_of_experiments,number_of_providers,3);
ndomStartx = zeros(number_of_experiments,number_of_providers,2,3);
ndomEndx = zeros(number_of_experiments,number_of_providers,2,3);
domStartx = zeros(number_of_experiments,number_of_providers,2,3);
domEndx = zeros(number_of_experiments,number_of_providers,2,3);
ndomStarty = zeros(number_of_experiments,number_of_providers,2,3);
ndomEndy = zeros(number_of_experiments,number_of_providers,2,3);
domStarty = zeros(number_of_experiments,number_of_providers,2,3);
domEndy = zeros(number_of_experiments,number_of_providers,2,3);

%Initialize output variables to store GA performance measures
run_time = zeros(number_of_experiments,1);
run_gens = zeros(number_of_experiments,1);
run_exit = zeros(number_of_experiments,1);
run_avgdist = zeros(number_of_experiments,1);
run_spread = zeros(number_of_experiments,1);
run_frontsize = zeros(number_of_experiments,1);
run_frontPA = zeros(number_of_experiments,2);
run_frontPB = zeros(number_of_experiments,2);

for experiment_number = 1:number_of_experiments
   efficient_frontier =[];
   [eijk,gj,ui,vlk,wil,vjk,eij,ci,di,si,Gl,d] = ...
      study4_Coefficients(number_of_treatments, number_of_diseases,...
      number_of_providers, dFF(experiment_number,:));
   Vl = sum(vlk,2);
   Vj = sum(vjk,2);

   %Generate all feasible solutions for model 2
   A=generateA(number_of_treatments, number_of_diseases);
   b=ones(1,number_of_diseases);
   model2BFSs=feassol(A,b);

   %Initialize decision variable
   xij = zeros(number_of_treatments, number_of_diseases);

   %Loop to determine objective function values of all feasible solutions
   objectiveValue_Model1 = zeros(size(model2BFSs,2),2,number_of_providers);
   objectiveValue_Model2 = zeros(size(model2BFSs,2),2);
   objectiveValue_Model3 = zeros(size(model2BFSs,2),2);

   for n = 1:size(model2BFSs,2)
      for i = 1:number_of_diseases
         for j = 1:number_of_treatments
            xij(i,j) = model2BFSs(number_of_treatments*(j-1)+i,n);
         end
      end

      %Calculate the objective function value of an indexed solution for
      %model 2
      objectiveValue_Model2(n,:) = ofv_Model2(xij,eij,ci);

      %Calculate the objective function value of an indexed solution
      %using model 2 as if model 1 is a submodel for each provider.
      for k = 1:number_of_providers
         objectiveValue_Model1(n,:,k) = ofv_Model2(xij,eijk(:,:,k),ci);
      end

      %Calculate the objective function value of an indexed solution for
      %model 3
```

```
        objectiveValue_Model3(n,:) = ofv_Model3(xij, di, eij, si, ui, sum(vjk,2));
    end

    %Use objective funtion values of all feasible solutions to determine
    %efficient frontiers model 1 to 3
    [~, solutionID2] = concaveFront(objectiveValue_Model2);
    [~, solutionID3] = concaveFront(objectiveValue_Model3);

    tic;

    % Genetic Algorithm for model4 returns Fval4 as the efficient frontier
%Code to estimate percentiles of members in low cost and high efficacy
%regions
avgCi = zeros(number_of_treatments,1); %average cost of i
avgEi = ones(number_of_treatments,1); %average efficacy of i
for i = 1:number_of_treatments
    avgCi(i)=(di(i)+si(i)*sum(Vl)^ui(i))/(sum(Vl));
    for j = 1:number_of_diseases
        avgEi(i) = avgEi(i)*(1-eij(i,j));
    end
    avgEi(i) = 1 - avgEi(i);
end

[minCRatio,minCIndex] = min(avgCi/sum(avgCi));
[maxCRatio,~] = max(avgCi/sum(avgCi));
[minERatio,minEIndex] = min(avgEi/sum(avgEi));
[maxERatio,maxEIndex] = max(avgEi/sum(avgEi));

% Genetic Algorithm for model4 returns Fval4 as the efficient frontier
NVARS4 = number_of_treatments*number_of_patient_types; %number of independent variables for the fitness function
popSize= 64*2;
popRand = randperm(popSize);
mypop = zeros(popSize,NVARS4);
M3frontSize = size(solutionID3,1);

for p = 1:popSize
    if popRand(p) <= round(popSize*(minCRatio))
        for l = 1:number_of_patient_types %original used mypop(p,...)..
            mypop(popRand(p), number_of_treatments*(l-1)+minCIndex) = 1;
        end
    elseif popRand(p) >= min(round(popSize*minERatio),popSize-round(popSize*(minCRatio)))
        for l = 1:number_of_patient_types
            mypop(popRand(p), number_of_treatments*(l-1)+maxEIndex) = 1;
        end
    else
        index = randi([1 M3frontSize]);
        %index = popRand(p) - divisor*(floor((popRand(p) - 1)/divisor));
        for j = 1:number_of_diseases
            for i = 1:number_of_treatments
                xij(i,j) = model2BFSs(number_of_diseases*(j-1)+i,...
                    solutionID3(index));
            end
        end
        for l=1:number_of_patient_types
            for j=1:number_of_diseases
                if find(d(l,:)==j)>0
                    for i=1:number_of_treatments
                        if xij(i,j) == 1
                            mypop(popRand(p), number_of_treatments*(l-1)+i) = 1;
                        end
                    end
                end
            end
        end
    end
end

%     mypop = CreationFcn1(popSize, solutionID3, model2BFSs, d, ...
%     number_of_treatments, number_of_diseases, number_of_patient_types);
```

```
LB4 = zeros(1, NVARS4);
UB4 = ones(1, NVARS4);
A4top = generateA(number_of_treatments, number_of_patient_types);
A4bottom = 0*A4top;
b4top = ones(number_of_patient_types,1);
b4bottom = zeros(number_of_patient_types,1);
Aeq4 = [];
beq4 = [];
options = gaoptimset(...
   'InitialPopulation',mypop,...
   'EliteCount', round(popSize*0.10), ...
   'CrossoverFraction', .80,...
   'CrossoverFcn',{@crossoverheuristic,1.2},...
   'PopulationSize', popSize, ...
   'StallGenLimit',100,...
   'MutationFcn',{@mutationbyDerivatives,d, eij, ci, gj, si, ui, Vl, Gl, wil},...
   'PlotFcns',@gaplotpareto);
   %'PlotFcns',@gaplotpareto

A4 = [A4top;A4bottom];
b4 = [b4top;b4bottom];

% Genetic Algorithm for model4 returns Fval4 as the efficient frontier
FITNESSFCN4 = @(x)m4_multiobjective(x,eij, di, si, ui, Vl, Gl, wil, d, gj);
[x, Fval4, exitflag, output, population] = gamultiobj(FITNESSFCN4,NVARS4,A4,b4,...
   Aeq4,beq4,LB4,UB4,options);
Fval4(:,2)=-1*Fval4(:,2);
concaveefficient_frontier = concaveFront(Fval4);

run_time(experiment_number) = toc;
run_gens(experiment_number) = output.generations;
run_exit(experiment_number) = exitflag;
run_avgdist(experiment_number) = output.averagedistance;
run_spread(experiment_number) = output.spread;
run_frontsize(experiment_number) = size(concaveefficient_frontier,1);
run_frontPA(experiment_number,:) = concaveefficient_frontier(1,:);
run_frontPB(experiment_number,:) = concaveefficient_frontier(size(concaveefficient_frontier,1),:);


 %determine provider k frontier in model 4
 for k = 1:number_of_providers

   [~,solutionID1] = concaveFront(objectiveValue_Model1(:,:,k));
   number_of_solutions = size(solutionID1,1);
   provider_frontier = [];
   provider_frontier = zeros(number_of_solutions,2);


   for r = 1:number_of_solutions
      for j = 1:number_of_diseases
         for i = 1:number_of_treatments
            xij(i,j) = model2BFSs(number_of_diseases*(j-1)+i,solutionID1(r));
         end
      end
   provider_frontier(r,:) = ofv_Model4(xij, eij, di, si, ui, Vl, Gl, wil, d, gj);
   end
   %sort the provider frontier
   [~,d2] = sort(provider_frontier(:,1));
   provider_frontier = provider_frontier(d2,:);

   [dx(experiment_number,k,1),...
      dy(experiment_number,k,1),...
      ndomStartx(experiment_number,k,:,1),...
      ndomEndx(experiment_number,k,:,1),...
      domStartx(experiment_number,k,:,1),...
      domEndx(experiment_number,k,:,1),...
      ndomStarty(experiment_number,k,:,1),...
      ndomEndy(experiment_number,k,:,1),...
      domStarty(experiment_number,k,:,1),...
      domEndy(experiment_number,k,:,1)]...
```

```
        = measure3(provider_frontier,concaveefficient_frontier);
    end

    %determine model2 frontier in model 4
    number_of_solutions = size(solutionID2,1);
    model2_frontier = [];
    model2_frontier = zeros(number_of_solutions,2);

    for r = 1:number_of_solutions
        for j = 1:number_of_diseases
            for i = 1:number_of_treatments
                xij(i,j) = model2BFSs(number_of_diseases*(j-1)+i,solutionID2(r));
            end
        end
    model2_frontier(r,:) = ofv_Model4(xij, eij, di, si, ui, Vl, Gl, wil, d, gj);
    end

    %sort model2frontier
    [~,d2] = sort(model2_frontier(:,1));
    model2_frontier = model2_frontier(d2,:);

    [dx(experiment_number,1,2),...
        dy(experiment_number,1,2),...
        ndomStartx(experiment_number,1,:,2),...
        ndomEndx(experiment_number,1,:,2),...
        domStartx(experiment_number,1,:,2),...
        domEndx(experiment_number,1,:,2),...
        ndomStarty(experiment_number,1,:,2),...
        ndomEndy(experiment_number,1,:,2),...
        domStarty(experiment_number,1,:,2),...
        domEndy(experiment_number,1,:,2)]...
        = measure3(model2_frontier,concaveefficient_frontier);

    %determine model3 frontier in model 4
    number_of_solutions = size(solutionID3,1);
    model3_frontier = [];
    model3_frontier = zeros(number_of_solutions,2);

    for r = 1:number_of_solutions
        for j = 1:number_of_diseases
            for i = 1:number_of_treatments
                xij(i,j) = model2BFSs(number_of_diseases*(j-1)+i,solutionID3(r));
            end
        end
    model3_frontier(r,:) = ofv_Model4(xij, eij, di, si, ui, Vl, Gl, wil, d, gj);
    end

    %sort model3frontier
    [~,d2] = sort(model3_frontier(:,1));
    model3_frontier = model3_frontier(d2,:);

    [dx(experiment_number,1,3),...
        dy(experiment_number,1,3),...
        ndomStartx(experiment_number,1,:,3),...
        ndomEndx(experiment_number,1,:,3),...
        domStartx(experiment_number,1,:,3),...
        domEndx(experiment_number,1,:,3),...
        ndomStarty(experiment_number,1,:,3),...
        ndomEndy(experiment_number,1,:,3),...
        domStarty(experiment_number,1,:,3),...
        domEndy(experiment_number,1,:,3)]...
        = measure3(model3_frontier,concaveefficient_frontier);
end

function [EfficientFront,FrontIndex] = concaveFront(f)
format long

[w,d2] = sort(f(:,1));
P = f(d2,:);
Q = d2;
```

```
N = length(P);
mab = ones(1,N);
keep = ones(1,N);
for a = 1:(N - 1)
    for b = (a + 1):N
        if (P(a,1) == P(b,1))*(P(a,2) <= P(b,2))*keep(b) == 1
            keep(a) = 0;
            break
        end
        mab(b) = (P(b,2) - P(a,2)) / (P(b,1) - P(a,1));
    end
    if keep(a) == 1
        for b = (a + 1):N
            if b < N
                if (mab(b) > max(mab((b + 1):N)))*(mab(b) > 0) == 0
                    keep(b) = 0;
                end
            else
                 if (mab(b) > 0) == 0
                    keep(b) = 0;
                end
            end
        end
    end
end

EfficientFront = zeros(sum(keep),2);
FrontIndex = zeros(sum(keep),1);
c = 1;
for i = 1:N
    if keep(i) == 1
    EfficientFront(c,:) = P(i,:);
    FrontIndex(c,:) = Q(i);
    c = c + 1;
    end
end
```