4-9-2012

Extensions of Multistage Stochastic Optimization with Applications in Energy and Healthcare

Ludwig Charlemagne Kuznia
University of South Florida, lkuznia@mail.usf.edu

Extensions of Multistage Stochastic Optimization with Applications in Energy and Healthcare

by

Ludwig Kuznia

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Industrial and Management Systems Engineering
College of Engineering
University of South Florida

Major Professor: Grisselle Centeno, Ph.D.
Tapas Das, Ph.D.
David Decker, M.D.
Sherwin Kouchekian, Ph.D.
Bo Zeng, Ph.D.

Date of Approval:
April 9, 2012

Keywords: Markov decision processes, chemotherapy, random processes, renewable energy systems, Benders' decomposition, probabilistic programming

Acknowledgements

I have learned during the process of earning my Ph.D. that it is more of a team effort than most people realize. I will therefore thank a number of people who I feel played a crucial role, big or small, in making this journey possible. I will start with those people who helped put me in a position to be successful. This must begin with my parents, Raymond and Maria Kuznia, without whom I would not have gone to college and therefore would certainly not be where I am today. Next, Pete Buczkowski, who graduated from Western Michigan University (my alma mater) and returned to give a talk about Operations Research during my junior year. Although it took four years, this talk inspired me to enter the Industrial Engineering program at the University of South Florida to study the subject.

It is natural to turn now to those that are currently aiding in my success. First, for taking a chance on a student of mathematics and agreeing to be my advisor, I owe a great deal to Dr. Grisselle Centeno. Next, to Dr. Bo Zeng for his many hours of discussion regarding research and life, and to Dr. David Decker, Veronica Decker, and Brian Decker, without whom Chapter 4 would not exist, I am very grateful. Also, Dr. Tapas Das and Dr. Sherwin Kouchekian for graciously serving on my committee. Additionally, I would like to thank Dr. Alex Savachkin and Dr. Jose Zayas-Castro for the great deal of support they provided me while at USF.

Finally, I cannot put into words the amount I wish to thank my wife, Lindsay Kuznia. She has been an unwavering source of support and motivation. Though I cannot say I would have been unable to finish this degree without her, I do know it would have been exponentially more challenging and much longer.

Table of Contents

List of Tables

List of Figures

Abstract

This dissertation focuses on extending solution methods in the area of stochastic optimization. Attention is focused to three specific problems in the field. First, a solution method for mixed integer programs subject to chance constraints is discussed. This class of problems serves as an effective modeling framework for a wide variety of applied problems. Unfortunately, chance constrained mixed integer programs tend to be very challenging to solve. Thus, the aim of this work is to address some of these challenges by exploiting the structure of the deterministic reformulation for the problem. Second, a stochastic program for integrating renewable energy sources into traditional energy systems is developed. As the global push for higher utilization of such green resources increases, such models will prove invaluable to energy system designers. Finally, a process for transforming clinical medical data into a model to assist decision making during the treatment planning phase for palliative chemotherapy is outlined. This work will likely provide decision support tools for oncologists. Moreover, given the new requirements for the usage electronic medical records, such techniques will have applicability to other treatment planning applications in the future.

1 Introduction

Humans are decision makers. The number and complexity of these decisions is perhaps one thing that sets us apart from the other animals of the world. Many of the decisions we make are simple to answer: "Do I want cream in my coffee today?" Others are much more complicated: "Given the current economic situation in the United States and Europe, how should I invest my assets so that I can retire by the age of 60?" The latter captures two elements that are the focus of this dissertation: stochasticity and extended planning. Making decision subject to uncertainty often falls under the purview of stochastic programming, which provides the framework for solving optimization problems subject to randomness. Although this is a widely applicable methodology, there is no archetypical example due to the large variety of problems to which this framework applies [72]. There are however, certain classes of problems that commonly are solved with stochastic optimization techniques such as two-stage models and sequential decision (multistage) models. These models are discussed as an introduction to modeling techniques and solution methods in stochastic programming.

1.1 Two-Stage Stochastic Programs

In two-stage stochastic programs decisions in the problem can be classified into two categories. First, those decisions that must be made prior to any random information being realized, and second, the decisions that are made after random information is realized. Generally, the decisions from the first category will impact the decisions that can be made from the second category. With this in mind, if ξ is a random variable, then the general form for two-stage stochastic (linear) program depending

on ξ is formulated as [13]:

$$\min \quad cx + \mathbb{E}\{F(x,\xi)\} \tag{1.1}$$

$$\text{subject to} \quad Ax \geq b \tag{1.2}$$

$$x \in \mathbb{R}^n_+ \tag{1.3}$$

where $F(x,\xi)$ is given by

$$\min \quad fy \tag{1.4}$$

$$\text{subject to} \quad H_\xi y \geq h_\xi - G_\xi x \tag{1.5}$$

$$y \in \mathbb{R}^m_+. \tag{1.6}$$

In the formulation of $F(x,\xi)$, h_ξ is a random vector depending on ξ and H_ξ and G_ξ are random matrices depending on ξ. Note that there is no restriction that x or y need to be continuous. Each realization of ξ is referred to as a *scenario*. The problem described by (1.1)-(1.3) is referred to as the first-stage problem and x is the first-stage decision vector, while (1.4)-(1.6) is called the second-stage problem and y is called the second-stage decision vector or the recourse decision vector. In problems such as these, the first-stage decisions, x in this case, must be determined prior to the realization of the random variable ξ while the second-stage decisions, y in this case, are made given both a realization of ξ and also a fixed value of the first-stage decision. If ξ has a finite distribution, i.e., $\mathbb{P}\{\xi = k\} = \pi_k$ for $k = 1, \ldots, K$ with $\sum_{k=1}^K \pi_k = 1$,

then (1.1)-(1.3) can be reformulated as a large-scale mixed integer program [72]:

$$\min \ cx + \sum_{k=1}^{K} \pi_k f y_k \tag{1.7}$$

$$\text{subject to } \ Ax \geq b \tag{1.8}$$

$$G_k x + H_k y_k \geq h_k \tag{1.9}$$

$$y_k \in \mathbb{R}^m_+ \tag{1.10}$$

$$x \in \mathbb{R}^n_+. \tag{1.11}$$

Note that notation has switched from ξ to k since the distribution is completely characterized by k. This formulation ensures that x is chosen to minimize the original objective while maintaining feasibility of recourse decisions with the trade off that the number of variables and constraints each grow like $O(K)$.

Two-stage stochastic problems are in general $\mathcal{NP}$-hard [13]. Coupling this with the fact that their size rapidly becomes large as the number of scenarios increases, it is not surprising that techniques have been developed in an attempt to solve two-stage problems more effectively. Perhaps the most famous is Benders' decomposition [11]. This solution method capitalizes on features common to all stochastic two-stage problems such as decomposability of the second-stage problem over scenarios once first-stage decisions are fixed. Benders' decomposition is discussed in detail in Chapter 3.

1.2 Multistage Stochastic Programs (MSSPs)

Multistage stochastic programs could be considered an extension to the two-stage programs discussed in the previous section. In this case, there are a series of decisions to be made sequentially with only past information available; the realization of future random variables impacting the system is not known during the current decision epoch. Some notation is now given to describe MSSPs.

Let $\{t \mid t = 1, \ldots, T\}$ be the set of decision epochs, Ξ_t be the set of instances of a random variable, ξ_t, where the subscript t denotes the possible dependence on a particular decision epoch, $\mathcal{X}_t(x_{t-1}, \xi_t)$ the feasible space of decision vectors, x_t, depending on the decisions from the previous decision epoch and the current realization of the random vector ξ_t, and $f_t : \mathcal{X}_t(-, \Xi_t) \times \Xi_t \to \mathbb{R}$ the cost function in epoch t. The problem can be visualized in Figure 1.

Figure 1: Depiction of a Multistage Decision Process

With this notation, the MSSP can be modeled as the following problem

$$\min_{x_1 \in \mathcal{X}_1(\xi_1)} f_1(x_1, \xi_1) + \mathbb{E}\left\{ \min_{x_2 \in \mathcal{X}_2(x_1, \hat{\xi}_2)} f_2(x_2, \hat{\xi}_2) + \mathbb{E}\Big\{ \cdots + \mathbb{E}\big\{ \min_{x_T \in \mathcal{X}_T(x_{T-1}, \hat{\xi}_T)} f_T(x_T, \hat{\xi}_T) \big\} \Big\} \right\}$$

where $\hat{\xi}_t = (\xi_1, \ldots, \xi_t)$. It is noted here that an assumption is made in using minimum rather than infimum in the above formulation. Specifically, the minimum for each problem is obtained over the feasible space. This is not an overly restrictive assumption, for instance requiring that $\mathcal{X}_t$ be compact and f_t be continuous is sufficient. Although the MSSP can look daunting to solve for even one-dimensional decisions, it does not need to be solved as one single problem; it can be solved through the use of Bellman's Equation, which serves as a commonly employed solution method for such dynamic programs. The strategy, developed by Bellman [9] in the 1950's, allows complex problems to be decomposed into a sequence of smaller problems. Let

$$V_T(x_{T-1}, \xi_T) = \min_{x_T \in \mathcal{X}_T(x_{T-1}, \xi_T)} f_T(x_T, \xi_T)$$

then for $1 \leq t \leq T-1$ write

$$V_t(x_{t-1}, \xi_t) = \min_{x_t \in \mathcal{X}_t(x_{t-1}, \xi_t)} \Big\{ f_t(x_t, \xi_t) + \mathbb{E}\left\{ V_{t+1}(x_{t+1}, \xi_{t+1}) \middle| \xi_{t-1} \right\} \Big\}.$$

Bellman's Principal of Optimality ensures that solving these recursive equations yields the optimal solution to a multistage stochastic problem. One case of the MSSP of particular interest is when $(\xi_1, \ldots, \xi_T)$ is Markovian. For discrete distributions Markovian means

$$\mathbb{P}\left\{ \xi_t = j \mid \xi_{t-1} = i, \xi_{t-2} = i_{t-2}, \ldots, \xi_1 = i_1 \right\} = P_{ij}$$

for all $i_1, \ldots, i_{t-2}, i, j$, and t [64]. In this case, the problem described is a Markov decision process [62]. The additional structure imposed on the probability distribution of ξ allows for efficient solution algorithms such as policy and value iteration [58, 62]. Policy iteration will be presented in detail in Chapter 4.

1.3 Additional Comments Regarding Stochastic Optimization

Two-stage programs and multistage programs provide seemingly different frameworks for modeling stochastic optimization problems. The reality is that these modeling techniques in essence are the same; just presented in differing formats for ease of solution. A problem modeled as a two-stage program can notationally be transformed into a multistage program and vice a versa. A slightly different language is used for each, but those fortunate enough to work with both modeling techniques can often develop a sort of Rosetta Stone for the terminology associated with them. Although this digression has no direct impact on the research presented in this dissertation, it is certainly worth mentioning. By recognizing the threads that unite the area of stochastic optimization, academicians and practitioners open many more avenues for attacking problems they face. Returning to the task at hand, the next section ex-

plains the impact of the work to be presented on the field and provides an outline of the remainder of the document.

1.4 Research Contributions

This dissertation will contribute to the body of knowledge in stochastic programming specifically pertaining to two-stage models and multistage models. Although the contributions will be discussed in detail in the next three chapters, they are first mentioned here. For two-stage models, challenges relating to solution methods for chance constrained mixed integer programs are addressed followed by a new application of the two-stage model to energy system design. The latter problem resulted in the development and implementation of an improved Benders' decomposition algorithm. The research relating to multistage models is more applied. It addresses deficiencies in modeling chemotherapy treatment planning. This work outlines the methodology for applying the stochastic multistage framework to this yet unexplored area. Moreover, it promises to provide oncologists with a more robust decision making tool that could impact the quality of care for cancer patients.

The remainder of this document is organized as follows. Chapter 2 discusses the extension of the classic two-stage stochastic model to include probabilistic constraints (also known as chance constraints) as well as a solution method for such problems. Chapter 3 presents a new stochastic mixed integer programming model for the design of an energy system consisting of traditional and renewable energies. Additionally, an improved implementation of Benders' decomposition is described as a solution method for this challenging problem. Chapter 4 outlines the process of modeling palliative chemotherapy as a Markov decision process. This includes the determination of the model's state space and reward function. Finally, an overall summary of work and future research directions is given in Chapter 5.

2 A Primal Heuristic for Chance Constrained Mixed Integer Programs

This chapter presents a solution method for mixed integer programs (MIPs) subject to probabilistic constraints. First, some background regarding probabilistic programming is presented. This is followed by a description of a novel heuristic approach developed for solving chance constrained MIPs. The chapter closes with some final remarks and future research directions.

2.1 Introduction to Chance Constrained Mixed Integer Programs

Chance constrained mathematical programs (CCMPs) are a class of probabilistic programs that incorporate uncertainty in the feasible space of the problem. CCMPs can be formulated as

$$\min \{f(x) \mid \mathbb{P}\{x \in P(\xi)\} \geq 1 - \epsilon, x \in X \subseteq \mathbb{R}^n\}$$

where ξ is a random vector, $P(\xi) \subseteq \mathbb{R}^n$, and ϵ is a confidence parameter chosen prior to solving to the problem and is typically small, e.g., $\epsilon = 0.05$. Although in general there are no restrictions on f or X, this work will focus on problems where these are linear. Specifically, attention is given to problems of the form

$$\min \{cx + fy \mid \mathbb{P}\{T_\xi x + G_\xi y \geq h_\xi\} \geq 1 - \epsilon, x \in X\}, \tag{2.1}$$

where x is a vector of integer variables and y is a vector of continuous variables. Such problems are referred to as chance constrained mixed integer programs (CCMIPs).

CCMIPs are a versatile modeling framework, being utilized to model problems relating to supply chain management [40], production planning [46], surface water quality management [76], air quality management [6], and chemical processes [28].

2.2 Previous Research in Probabilistic Programming

Chance constrained mathematical programs were introduced by Charnes et. al in the 1950's and were applied to individual constraints [17]. Miller and Wagner extended this idea to include joint probabilistic constraints with independent right hand side distributions [45]. Probabilistic programming with joint probabilistic constraints and dependent right hand side distributions was pioneered by Prékopa beginning in 1970 [59]. Although this class of problems has been well studied, see for example [61], there are two major challenges in solving problems with probabilistic constraints. First, calculating $\mathbb{P}\{T_\xi x + G_\xi y \geq h_\xi\} \geq 1 - \epsilon$ can be computationally burdensome, for instance it may require multidimensional integration. The second major challenge is that the feasible region defined by the probabilistic constraint is generally not convex. One notable exception to this is when x and y are continuous, T_ξ and G_ξ are deterministic, and the distribution for h_ξ is log-concave, in which case the feasible region is convex [59–61]. A non-convex feasible space is often the result of ξ having discrete support. For this case, various methods have been developed to solve chance constrained mathematical programs given certain assumptions. Assuming deterministic coefficient matrix and a linear program without the chance constraint, a branch and cut strategy was developed using strengthened star inequalities [41, 42]. The methodology applies to problems that are MIPs without chance constraints, though it proves far less effective. Tanner and Ntaimo also implemented a branch and cut strategy, however their technique is based on combinatorial type Benders cuts generated from irreducibly infeasible subsystems. Their work allowed for a random coefficient matrix [77].

In this work, an algorithm is developed for solving CCMIPs with the assumption that ξ has discrete finite support, i.e., $\mathbb{P}\{\xi = k\} = \pi_k$ for $k = 1, \ldots, K$. This assumption allows (2.1) to be re-formulated as a deterministic MIP using "big-M" constraints, provided $F_\xi = \{x \mid G_\xi x + H_\xi y_\xi \geq h_\xi\}$ is "nice." "Nice" here can mean

that F_ξ is a compact set, though this stringent condition is not generally required. However, given this provision, the problem can be formulated as

$$\min \quad cx + \mathbb{E}\{fy_k\} \tag{2.2}$$

$$\text{subject to} \quad G_k x + H_k y_k + M_k z_k \geq h_k \qquad k = 1, \ldots, K \tag{2.3}$$

$$Ax \geq b \tag{2.4}$$

$$\sum_{k=1}^{K} \pi_k z_k \leq \epsilon \tag{2.5}$$

$$x \in \mathbb{R}^n_+ \tag{2.6}$$

$$x_i \in \mathbb{Z}_+ \qquad i \in N \tag{2.7}$$

$$z_k \in \{0, 1\} \qquad k = 1, \ldots, K \tag{2.8}$$

$$y_k \in \mathbb{R}^m_+ \qquad k = 1, \ldots, K \tag{2.9}$$

where M_k is large enough that if $z_k = 1$ then constraint k of the form (2.3) is satisfied for all $(x, y) \in \left\{(x, y) \mid Ax \geq b,\ x \in \mathbb{R}^n_+,\ x_i \in \mathbb{Z}_+ \text{ for } i \in N,\ y \in \mathbb{R}^+_n\right\}$ [65]. Constraint (2.5) ensures that any feasible solution to (2.2)-(2.9) fulfills the requirement expressed by $\mathbb{P}\{G_k x + H_k y_k \geq h_k\} \geq 1 - \epsilon$.

Even given this MIP formulation of a CCMIP, it is clear that in general this can be a difficult problem to solve due in part to the presence of the the big-M constraints; see [42] for a proof that the problem is in general $\mathcal{NP}$-hard. In the remainder of this chapter, a primal heuristic, dubbed the Probe, Explore, Rank, and Cut Heuristic (PERC Heuristic), for solving the deterministic formulation of a CCMIP is described. This is a significant step toward developing solution methods for MIPs subject to chance constraints, since most existing work is aimed at linear programs subject to chance constraints. The Heuristic is presented in detail in Section 2.3 and computational results are provided in Section 2.4.

2.3 Probe, Explore, Rank, and Cut Heuristic

This section begins with an assumption and some notation. Next, a general outline for the PERC Heuristic is given followed by the pseudocode for the Heuristic. For this Heuristic, it is assumed that $\pi_k = 1/K$ for all k. If this is not the case, then it can be coerced with techniques such as increasing the number scenarios through over sampling or the use of sample average approximation [54]. Next, let $P = (P_1, \ldots, P_K)$ and set $P_k = \delta$, $0 < \delta \leq 1$, for all k. Let $M(P)$ be the problem described by (2.2)-(2.9) with M_k replaced with $P_k M_k$. That is,

$$
\begin{aligned}
\min \quad & cx + \mathbb{E}_k \{ f y_k \} && (M(P)) \\
\text{subject to} \quad & G_k x + H_k y_k + P_k M_k z_k \geq h_k \quad k = 1, \ldots, n \\
& Ax \geq b \\
& \sum_{k=1}^{K} z_k \leq \lfloor K\epsilon \rfloor \\
& x \in \mathbb{R}^n_+ \\
& x_i \in \mathbb{Z}_+ \text{ for } i \in N \\
& y_k \in \mathbb{R}^m_+ \\
& z_k \in \{0, 1\}.
\end{aligned}
$$

Notice that a solution for $M(P)$ is feasible for (2.2)-(2.9), so $M(P)$ is a restriction of (2.2)-(2.9). Next, let $M(P, m)$ be the problem $M(P)$ with the constraint that the objective function should be at least as good as m. Notice that $M(P, \infty) = M(P)$. Let $V^*(M(P, m))$ denote the optimal value of the problem $M(P, m)$. Finally, let $M_{\overline{z}}$ be the problem (2.2)-(2.9) subject to $z = \overline{z}$, where $\overline{z}$ is a vector of 1's and 0's.

The general idea behind the PERC Heuristic is to start with a restricted problem that can be efficiently solved then intelligently expand the feasible space. The algorithm starts by solving the restricted problem $M(P, m)$. The optimal solution for

this problem is then used to expand the search space by increasing the value of some P_k's. This newly expanded region is explored by solving $M_{\overline{z}}$ where $\overline{z}$ is the optimal solution from $M(P, m)$. Recall that there is an assumption that $M_{\overline{z}}$ can be solved efficiently, so there is little computational burden associated with solving this problem. Once the optimal solution is found in this expanded region, a cardinality constraint is added to remove the extreme points in this region. This process is depicted in Figure 2 below. The large outer hexagon represents the feasible scenario space of $M(1)$ while the smaller inner hexagon represents $M(P)$. The black dots indicate the region currently being explored and the grey squares are the scenarios with $z_k = 1$.

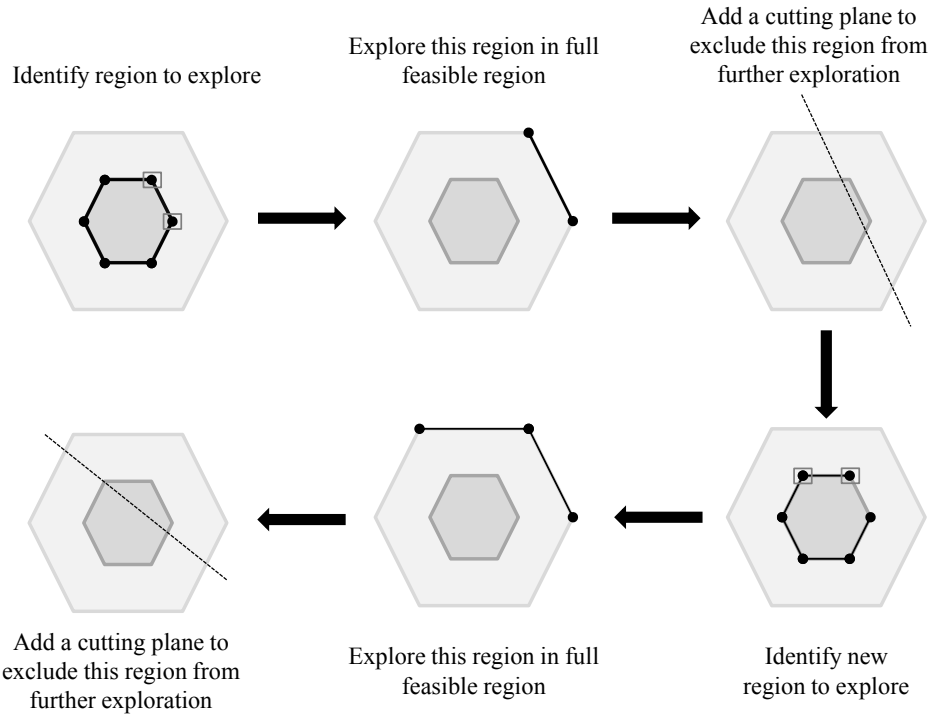


Figure 2: Visual Representation of PERC Heuristic

Table 1 presents the pseudocode for the PERC Heuristic. In the actual implementation, additional steps can be taken to improve performance. In this work, such steps included the addition of precedence constraints and variable fixing. These are presented in greater detail in Sections 2.3.1 and 2.4.1, respectively.

Table 1: Probe, Explore, Rank, and Cut Heuristic for CCMIPs

Line	Code Description
1	Set $t = 0$, $v = m = +\infty$, Value = `True`
2	**while** Value
3	Optimize $M(P, m)$
4	**if** $M(P, m)$ is infeasible
5	Set Value = `False`
6	**else**
7	**Step 1**
8	Let (x^t, y^t, z^t) be the optimal solution to $M(P, m)$
9	Set $m = \min\{m, V^*(M(P, m))\}$
10	**for** $k = 1, \ldots, K$
11	**if** $z_k^t = 1$ and $P_k < 1$ **:** Set $P_k = 1$
12	**end for**
13	**Step 2**
14	Optimize M_{z^t}
15	Let $(\hat{x}^t, \hat{y}^t, \hat{z}^t)$ be the optimal solution
16	**if** $V^*(M_{z^t}) < v$
17	Set $v = V^*(M_{z^t})$ and $(x^*, y^*, z^*) = (\hat{x}^t, \hat{y}^t, \hat{z}^t)$
18	Add the constraint $\sum_{j\,:\,P_j=1} z_j \leq \lfloor K\epsilon \rfloor - 1$ to $M(P, -)$
19	**end if**
20	t=t+1
21	**end while**
22	At termination, (x^*, y^*, z^*) is the best solution found.

It is important to note that the problem M_{z^t} solved in line 14 of Table 1 is a two-stage MIP. There is an assumption that this problem can be solved efficiently through the use of a commercial solver or an existing algorithm. Before moving on, a proposition regarding the finite termination of this algorithm is presented.

Proposition 1. *The PERC Heuristic terminates in at most $K + 1$ iterations.*

Proof. Notice that after at most K iterations the constraint $\sum_k z_k \leq \lfloor K\epsilon \rfloor - 1$ will be added to $M(P, m)$. Since in the best optimal solution $M(P, m)$ thus far has $\sum_k \hat{z}_k = \lfloor K\epsilon \rfloor$, the best optimal solution is no longer feasible. Therefore, $M(P, m)$ will be infeasible in iteration $K + 1$ and the algorithm will terminate. □

2.3.1 Precedence Constraints

Precedence constraints are those of the form $z_j \geq z_i$ for some i and j. Such constraints can be very useful to improve performance, the challenge is effectively identifying them. The following theorem provides some insights for identifying such constraints for this class of problems.

Theorem 2. *Let $S_k(\hat{x})$ denote the second-stage cost for scenario k given a feasible $\hat{x}$. If $S_j(\hat{x}) \geq S_i(\hat{x})$ for every feasible $\hat{x}$, then $z_j \geq z_i$.*

This is a very strict condition to satisfy. However, if the inequality holds for some set of feasible solutions, then it is reasonable to include the associated precedence constraint in the PERC Heuristic. Given this, the process of generating precedence constraints to be added to $M(P, m)$ during the PERC Heuristic is now presented. Let $\hat{S}_k^t = (S_k^1, S_k^2, \ldots, S_k^t)$ where S_k^t represents the second stage costs of scenario k in iteration t, then consider a pair $\hat{S}_j^t$ and $\hat{S}_i^t$ such that $\hat{S}_j^t \geq \hat{S}_i^t$, where the inequality holds for each component. This means across t iterations of the PERC Heuristic the costs associated with scenario j are higher than those of scenario i. Therefore, it is reasonable to assume that scenario j should be ignored (i.e., $z_j = 1$) before scenario i. Mathematically, this can be written as the following precedence constraint: $z_j \geq z_i$, which says if scenario i is ignored ($z_i = 1$) then scenario j is ignored as well ($z_j = 1$). Thus, beginning with the second iteration these constraints are generated after solving M_{z^t} in line 14 of Table 1. In this work, the inequality $\hat{S}_j^t \geq \hat{S}_i^t$ is checked for all pairs (i, j) in the implementation of the PERC Heuristic. Ignoring for a moment these additional inequalities, an example is presented to provide further insight to the mechanics of the PERC Heuristic.

2.3.2 Example of PERC Heuristic

Consider a CCMIP with ten scenarios ($K = 10$) and $\epsilon = 0.2$. In such a problem, (2.5) implies at most two z_k are non-zero. A hypothetical implementation of the

algorithm is presented in Table 2. As mentioned in the previous section, this example does not incorporate precedence constraints since the objective is to provide the reader with an overview of the process by which the Heuristic works.

Table 2: Example Implementation of the PERC Heuristic

Line	Explanation
1	**Step 0**
2	Let $P_k = \delta < 1$ for all k and $t = 1$
3	*Iteration 1*
4	**Step 1**
5	Suppose that $z^1 = (z_1^1, z_2^1, z_3^1, \ldots, z_{10}^1) = (1, 1, 0, \ldots, 0)$
6	Update $P = (1, 1, \delta, \ldots, \delta)$
7	**Step 2**
8	Solve M_{z^1} (i.e., (2.2)-(2.9) subject to $z = z^1$)
9	Add the constraint $z_1 + z_2 \leq 1$ to $M(P, m)$
10	*Iteration 2*
11	**Step 1**
12	Suppose that $z^2 = (z_1^2, z_2^2, z_3^2, z_4^2, \ldots, z_{10}^1) = (1, 0, 1, 0 \ldots, 0)$
13	Update $P = (1, 1, 1, \delta, \ldots, \delta)$
14	**Step 2**
15	Solve M_{z^2} (i.e., (2.2)-(2.9) subject to $z = z^2$)
16	Add the constraint $z_1 + z_2 + z_3 \leq 1$ to $M(P, m)$
17	$\vdots$
18	*Iteration 10*
19	**Step 1**
20	Suppose that $z^{10} = (z_1^{10}, z_2^{10}, z_3^{10}, z_4^{10}, \ldots, z_{10}^{10}) = (0, 1, 0, \ldots, 1)$
21	Update $P = (1, 1, 1, 1, \ldots, 1)$
22	**Step 2**
23	Solve $M_{z^{10}}$ (i.e., (2.2)-(2.9) subject to $z = z^{10}$)
24	Add the constraint $z_1 + z_2 + z_3 + \cdots + z_{10} \leq 1$ to $M(P, m)$
25	*Iteration 11*
26	$M(P, m)$ is now infeasible so the algorithm terminates.

Generally, it will not be required to explore all scenarios as outlined in the example above, though such an exploration does ensure the finite termination of the algorithm as pointed out in Proposition 1. With the mechanics of the PERC Heuristic presented, the computational experiments are now explained along with additional methods utilized for improving the performance of the Heuristic.

2.4 Computational Experiments

2.4.1 Variable Fixing and Other Improvements

Variable fixing is conducted prior to the implementation of the algorithm; it can be thought of as a preprocessing step. The purpose of variable fixing is to reduce the search space of $M(P, m)$ and thereby decrease computation time in each iteration. Two distinct methods were employed to perform variable fixing. All attention was focused on finding z_k's to fix at zero. The first method is the following: Solve (2.2)-(2.9) with constraint (2.8) relaxed, referred to as the relaxed CCMIP. If $z_k = 0$ in the relaxed CCMIP, then the constraint $z_k = 0$ is added to $M(P, m)$ in the algorithm.

The second method is slightly more complicated, but can be thought of as is a "greedy" tactic that provides an initial relaxed region to explore. After the initial exploration of this region, some z_k's can be fixed at zero. The first step in the process is to construct two sets, one based on the ranking of second stage costs and the other based on the values of z_k in the optimal solution of the relaxed CCMIP. Let $\{i_1, i_2, \ldots, i_K\}$ be the rank of the second stage costs in increasing order and $U_1 = \{i_1, i_2, \ldots, i_{\lfloor K\epsilon \rfloor}\}$. Also, let $\{j_1, j_2, \ldots, j_K\}$ be such that $z_{j_1} \geq z_{j_2} \geq \cdots \geq z_{j_K}$ and $U_2 = \{j_1, j_2, \ldots, j_{\lfloor K\epsilon \rfloor}\}$. Next, set $U = U_1 \cup U_2$ and for $j \in U$ let $P_j = 1$. Now, in the first iteration when the problem $M(P, \infty)$ is solved if $z_k = 0$ and $k \in U$ then z_k is fixed at zero for the duration of the algorithm. This process provides a method for greatly reducing the search space through variable fixing.

Additional techniques were utilized for improving the Heuristic performance. Since a high quality solution will provide insights regarding which scenarios to allow to be violated completely, the problem solved in line 5 of Table 1 does not need to be solved to optimality. Since Gurobi [53], a professional solver, was used to solve this problem, we were able to adjust the default settings to decrease solution time. Specifically, the relative optimality gap was relaxed from 10^{-4} to 10^{-3}. Another tactic employed was restricting the number of consecutive iterations allowed without an improvement to

the best solution found. That is, if v does not decrease for r iterations, where r is a (termination) threshold selected prior to implementation, then the heuristic terminates. With these improvements, the Heuristic was tested on various randomly generated two-stage stochastic mixed integer programs of the form (2.2)-(2.9). All first stage variables are assumed to be integer valued. The process of generating the test sets for the numerical experiments is now described.

2.4.2 Data Generation

The models tested all had the following format:

$$\min \quad cx + \mathbb{E}\{fy_k\} \tag{2.10}$$

$$\text{subject to} \quad Gx - H_k^1 y_k + M_k^1 z_k \geq 0 \qquad k = 1, \ldots, K \tag{2.11}$$

$$H_k^2 y_k + M_k^2 z_k \geq h_k \qquad k = 1, \ldots, K \tag{2.12}$$

$$Ax \geq b \tag{2.13}$$

$$\sum_{k=1}^{K} z_k \leq \lfloor K\epsilon \rfloor \tag{2.14}$$

$$x \in \mathbb{Z}_+^n, z_k \in \{0,1\}, y_k \in \mathbb{R}_+^m. \tag{2.15}$$

The entries of c, f, G, H_k^1, H_k^2, and h_k were all nonnegative integers, while the entires of A and b were general integer values. In total, two classes of datasets with various numbers of scenarios were generated to test the performance of the PERC Heuristic. Before providing the properties of the datasets, some notation is given. Let N_c the number of continuous variables, N_i be the number of integer valued variables, M_s the total number of second-stage constraints, and M_f the number of first-stage constraints, then the tuple (M_f, M_s, N_i, N_c) describes the general characteristics of a class. The properties of the classes are provided in Table 3, where (a, b) indicates a uniform distribution for integers between a and b inclusive. Five instances of each class were generated with 250, 500, and 750 scenarios. Using this data, the PERC

Heuristic was tested against the commercial solver Gurobi, with solution value and time being of primary interest.

Table 3: Properties of the Datasets for Computational Experiments

	Rows/Distribution				Rows/Distribution			
Characteristics	G	H_k^1	H_k^2	h_k	A	b	c	f
(5,15,10,20)	6	6	9	9	5	5	10	20
	(0,10)	(0,3)	(0,3)	(10,100)	(-25,25)	(-50,50)	(100,300)	(5,10)
(10,25,15,35)	10	10	15	15	10	10	15	35
	(0,10)	(0,3)	(0,3)	(10,100)	(-25,25)	(-50,50)	(100,300)	(5,10)

2.4.3 Computational Results

Tables 4 and 5 provide the results for the numerical experiments conducted with $\epsilon = 0.05$ and $\epsilon = 0.1$, respectively. A time limit of one hour was set for all trials. Within the tables, "NA" indicates that the instance was proven infeasible. The results in both tables utilized the improvements discussed Section 2.4.1. Also, P_k was initialized with the value $1/25$ for all k.

Table 4: Computational Comparison of PERC Heuristic to Gurobi: $\epsilon = 0.05$

			Heuristic		Solver		
Scenarios	Characteristics	ID	Value	Time	UB	% Gap	Time
250	5,15,10,20	1	3046.0	26.3	3045.9	0.0	47.0
		2	2938.8	39.0	2938.8	0.0	64.8
		3	2847.4	15.0	2847.4	0.0	24.0
		4	2887.7	22.2	2887.7	0.0	49.4
		5	3598.7	21.1	3598.0	0.0	33.9
500	5,15,10,20	1	3879.0	107.5	3877.9	0.0	1245.3
		2	2832.5	71.3	2812.8	0.0	922.5
		3	2440.4	56.8	2402.9	0.0	736.5
		4	2750.0	68.8	2688.8	0.0	363.6
		5	8739.3	6.4	8739.3	0.0	1.5
750	5,15,10,20	1	2261.0	136.3	2261.0	0.0	921.9
		2	2891.5	74.3	2756.8	0.0	769.4
		3	2644.6	130.7	2657.3	14.7	3600.1*
		4	1937.4	139.3	1891.2	0.0	888.7
		5	NA	12.2	NA	NA	14.4
250	10, 25,15,35	1	3485.9	171.8	3485.5	0.0	1050.6
		2	3740.0	128.5	3739.7	0.0	540.6
		3	3028.1	145.3	3028.1	0.0	360.3
		4	3916.8	83.2	3916.0	0.0	3600.0*
		5	5724.5	67.5	5723.5	0.0	3600.3*
500	10, 25,15,35	1	3606.0	1028.9	3606.0	9.9	3600.0*
		2	5498.6	643.5	5498.4	0.0	3600.2*
		3	2719.5	445.4	2774.1	3.0	3600.0*
		4	2549.2	672.5	2684.1	22.1	3600.0*
		5	8184.9	50.4	8184.8	0.0	59.6
750	10, 25,15,35	1	NA	33.5	NA	NA	33.5
		2	9503.4	110.6	9503.0	0.0	141.3
		3	3438.4	1877.0	3631.3	31.5	3600.0*
		4	2623.2	1315.5	2717.8	22.1	3600.0*
		5	NA	61.5	NA	NA	60.5

* indicates time limit met

With the exception of 6 instances out of 60, the PERC Heuristic terminates faster than the commercial solver. For some small instances, the heuristic produces a solution of lower quality than solver (e.g., 500 scenarios with $\epsilon = 0.05$). Interestingly, as the problem becomes harder for the commercial solver, the PERC Heuristic tends produce better solutions in significantly less time. In particular, if the commercial solver terminates due to the one hour time limit with a gap of at least 10%, then the PERC Heuristic produces a better solution. Moreover, the time required is often significantly less for these instances.

Table 5: Computational Comparison of PERC Heuristic to Gurobi: $\epsilon = 0.1$

			Heuristic		Solver		
Scenarios	Characteristics	ID	Value	Time	UB	% Gap	Time
250	5,15,10,20	1	2785.1	26.0	2783.8	0.0	86.1
		2	2722.0	36.4	2717.5	0.0	140.4
		3	2674.7	53.7	2674.7	0.0	152.2
		4	2650.7	18.2	2650.7	0.0	81.0
		5	3414.3	22.7	3412.7	0.0	228.9
500	5,15,10,20	1	3706.3	145.8	3726.3	8.1	3600.0*
		2	2648.2	105.5	2673.2	4.3	3600.0*
		3	2220.3	128.8	2205.1	0.0	1340.6
		4	2668.3	50.8	2570.5	0.1	3600.1*
		5	8723.2	6.3	8723.2	0.0	1.5
750	5,15,10,20	1	2104.6	259.1	2204.8	10.5	3600.0*
		2	2605.3	292.2	2604.0	0.3	3600.1*
		3	2447.9	467.9	2516.1	24.3	3600.0*
		4	1816.5	270.4	1771.3	0.1	3600.0*
		5	NA	13.2	NA	NA	12.1
250	10, 25,15,35	1	3323.1	257.9	3322.6	0.0	2078.6
		2	3500.4	138.0	3500.1	0.0	1685.2
		3	2966.7	119.3	2880.4	0.0	1328.9
		4	3900.1	123.3	3898.0	0.1	3600.4*
		5	5707.8	85.2	5429.1	0.0	106.2
500	10, 25,15,35	1	3482.6	1275.8	3618.0	27.4	3600.1*
		2	5141.3	678.9	5595.9	21.8	3600.5*
		3	2702.5	661.2	2701.0	9.4	3600.0*
		4	2507.9	519.8	2592.7	28.7	3600.0*
		5	8170.3	60.9	8170.0	0.0	15.8
750	10, 25,15,35	1	NA	2.8	NA	NA	2.7
		2	9487.5	92.8	9487.1	0.0	129.1
		3	3285.2	3031.1	3753.4	48.2	3600.0*
		4	2543.2	2858.7	2850.1	40.1	3600.0*
		5	NA	64.7	NA	NA	64.8

* indicates time limit met

These results are quite promising. This algorithm was tested with problems having no specific structure beyond being two-stage models. Given more insight to the nature of the problem, addition techniques, such as variable fixing described in [70], could be implemented within the algorithm (as opposed to only prior to the algorithm as in this work). The implementation of such techniques would significantly improve computational performance.

Another set of experiments was conducted to provide a more accurate comparison of the performance of the algorithm to the commercial solver. After solving a test instance with the PERC Heuristic, the instance was solved with the commercial solver with a time limit equal to the time taken for the PERC Heuristic to terminate. The results of these trials are given in Tables 6 and 7 with $\epsilon = 0.05$ and $\epsilon = 0.1$, respectively.

The column Relative Change in % Gap is computed in the following manner. Given a lower bound on the optimal solution as determined in Tables 4 and 5, the optimality gap is computed for the PERC Heuristic solution, G_H, and the best feasible solution found by solver, G_S. The Relative Change in % Gap is then given by $1 - G_H/G_S$, with the convention that $0/0 = 1$. Thus, a value close to 1 indicates the PERC Heuristic found a much better solution than the commercial solver and a value close to 0 indicates similar qualities produced.

Table 6: Comparison of PERC Heuristic to Gurobi with Time Limit: $\epsilon = 0.05$

Scenarios	Characteristics	ID	Heuristic Value	Solver Value	Relative Change in % Gap	Time
250	5,15,10,20	1	3046.0	3115.1	1.00	26.3
		2	2938.8	2938.8	0.00	39.0
		3	2847.4	2865.0	1.00	15.0
		4	2887.7	2908.6	1.00	22.2
		5	3598.7	3598.1	0.00	21.1
500	5,15,10,20	1	3879.0	4140.0	0.99	107.5
		2	2832.5	3213.8	0.95	71.3
		3	2440.4	2492.2	0.58	56.8
		4	2750.0	2894.9	0.70	68.8
		5	8739.3	8739.3	0.00	6.4
750	5,15,10,20	1	2261.0	2386.0	1.00	136.3
		2	2891.5	3032.6	0.51	74.3
		3	2644.6	4037.6	0.81	130.7
		4	1937.4	1947.0	0.17	139.3
		5	NA	NA	NA	12.2
250	10, 25,15,35	1	3485.9	4238.9	1.00	171.8
		2	3740.0	4088.4	1.00	128.5
		3	3028.1	3156.2	1.00	145.3
		4	3916.8	4066.3	0.99	83.2
		5	5724.5	5724.0	0.00	67.5
500	10, 25,15,35	1	3606.0	3767.9	0.33	1028.9
		2	5498.6	5865.6	1.00	643.5
		3	2719.5	2899.4	0.88	445.4
		4	2549.2	2992.0	0.56	672.5
		5	8184.9	8184.8	0.00	50.4
750	10, 25,15,35	1	NA	NA	NA	33.5
		2	9503.4	9506.4	0.74	110.6
		3	3438.4	4254.5	0.55	1877.0
		4	2623.2	2876.6	0.39	1315.5
		5	NA	NA	NA	61.5

From these trials, the computational improvement of the PERC Heuristic over the commercial solver is more evident. First, the solution quality of the PERC Heuristic is always at least as good as the commercial solver. Second, if the solver terminates due to the time limit imposed from the PERC Heuristic, which is the case for all

Table 7: Comparison of PERC Heuristic to Gurobi with Time Limit: $\epsilon = 0.1$

Scenarios	Characteristics	ID	Heuristic Value	Solver Value	Relative Change in % Gap	Time
250	5,15,10,20	1	2785.1	2821.2	0.96	26.0
		2	2722.0	2828.3	0.96	36.4
		3	2674.7	2719.1	1.00	53.7
		4	2650.7	2814.9	1.00	18.2
		5	3414.3	3540.7	0.99	22.7
500	5,15,10,20	1	3706.3	4317.2	0.70	145.8
		2	2648.2	3212.7	0.87	105.5
		3	2220.3	2403.8	0.92	128.8
		4	2668.3	6709.3	0.98	50.8
		5	8723.2	8723.2	0.00	6.3
750	5,15,10,20	1	2104.6	2230.5	0.54	259.1
		2	2605.3	2818.7	0.96	292.2
		3	2447.9	2563.8	0.21	467.9
		4	1816.5	1966.2	0.76	270.4
		5	NA	NA	NA	13.2
250	10, 25,15,35	1	3323.1	3869.2	1.00	257.9
		2	3500.4	3888.3	1.00	138.0
		3	2966.7	3203.5	0.73	119.3
		4	3900.1	4129.9	0.97	123.3
		5	5707.8	5707.3	0.00	85.2
500	10, 25,15,35	1	3482.6	4030.6	0.46	1275.8
		2	5141.3	7016.0	0.77	678.9
		3	2702.5	2802.0	0.30	661.2
		4	2507.9	3395.0	0.64	519.8
		5	8170.3	8170.0	0.00	60.9
750	10, 25,15,35	1	NA	NA	NA	2.8
		2	9487.5	9487.5	0.06	92.8
		3	3285.2	3753.4	0.38	3031.1
		4	2543.2	2850.1	0.38	2858.7
		5	NA	NA	NA	64.7

but 3 instances, then the solution from the PERC Heuristic is in general significantly better.

2.5 Conclusions and Future Research in Chance Constrained Programming

Chance constrainted MIPs are a widely applicable class of problems that are in general quite challenging to solve. As such, it is important to identify/develop efficient solution strategies that are independent of the particular application. With this research, an attempt has been made to do just that. In this chapter, a novel approach for solving CCMIPs was presented that is application independent. Although the PERC Heuristic is not an exact solution method, it does tend to produce near optimal solutions more quickly than a commercial solver. These results indicate that this new solution strategy may be the right approach for this class of problems. For future

research, we will look to convert the PERC Heuristic into an exact solution method. This could be accomplished by coupling the Heuristic with a relaxation technique, such as Lagrangian relaxation.

3 Hybrid Energy System Design

This chapter, based on [38], relates to the integration of renewable energy into existing energy systems in remote areas. The challenges of modeling and solving such a problem are discussed. More specifically, the chapter is organized as follows: in Section 3.2 the problem description and the mathematical formulation are provided. Section 3.3 presents the details of the Benders' decomposition algorithm developed for solving the problem. Computational results for a set of test instances are given in Section 3.4. Conclusions and future research directions are discussed in Section 3.5.

3.1 Introduction to Hybrid Energy Systems

Providing energy to remote or isolated areas can be extremely costly due to the investment associated with transmission networks, land acquisition, control towers, and construction materials. The issue of energy independence is also a concern given that traditional configurations of power delivery depend heavily on energy or fuel generated outside the target area. There are two typical ways to deliver energy to remote areas, through transmission lines from places with excess capacity or large-scale generating facilities, or by utilizing a local thermal generating facility. Establishing a line to an offshore island is very expensive due to the complications associated with laying underwater transmission lines. Power generation using local thermal systems can be even more expensive due to high transportation and inventory holding costs [21, 57, 71]. Moreover, both options unavoidably lead to environmental concerns, especially the latter which produces a large volume of green house gases and pollutants. Technology advancements, especially relating to wind and solar energy, provide remote areas with the option of renewable energy systems for reliable energy supply.

Compared with traditional systems, renewable energy systems have a clear advantage environmentally as well as on the energy independence aspect.

In 2009, the American Clean Energy and Security Act of 2009 [26], was passed by the U.S. House of Representatives requiring the deployment of clean energy resources and a reduction in pollutants that contribute to global warming. This act will aid in the transition to a clean energy economy. However, the implementation of a reliable clean energy system may be limited by the nature of wind and solar energy which tends to be intermittent and highly variable [8, 57, 71]. To handle this challenge, energy storage equipment including pump water [14], battery [49], and hydrogen and a fuel cell [21], can be integrated into the system. Also, traditional thermal generators and transmission lines can be utilized, in a less frequent fashion, to deal with the random generation aspect of renewable energy/storage device systems, see [47] and references therein for a recent review.

Various heuristics have been applied to this problem, including simulated annealing [23], genetic algorithm [71], and tabu search [37], to derive good solutions. To capture the impact of randomness in system design, simulation based optimization methods are among the most popular approaches due to their ability to evaluate the system design in random environments, see [12] for a review on simulation optimization study. One obvious drawback of heuristics and simulation based optimization is that they cannot guarantee the quality of the solution. Therefore, these methods yield little insight into optimal system configuration.

Stochastic programming models have also been applied to model the energy system design problem and to derive the optimal configuration considering various probabilistic scenarios. In [1], a stochastic mixed integer programming (SMIP) model for optimal sizing of storage system is developed for an existing isolated wind-diesel power system. The randomness for wind and load is described by a set of scenarios where one scenario represents a 24-hour instance of wind and load. The model is

solved using a mixed integer programming (MIP) solver through GAMS [75]. In [14], a stochastic linear programming model is developed for optimal capacity design of a pumped storage device in a hybrid system where thermal generators' generation level is known. Nevertheless, with the exception of storage capacity, little work has been done using stochastic programming as a tool to study the larger scope system design. This is most likely due to the limitation of solvers to efficiently deal with the complexity of SMIP models for hybrid system design.

Clearly, if SMIP is employed to model a more comprehensive hybrid system design, instead of depending on professional MIP solvers, it is necessary to develop customized solution procedures that can efficiently compute an optimal system configuration. Currently, algorithms based on Benders' decomposition method [11] have been deemed most effective ones to solve various SMIP applications. Also, many enhancement strategies have been designed and implemented to make further improvement. Among them, the idea of Pareto-optimal cuts by Magnanti and Wong [43] is widely adopted. This method was recently modified and improved by Papadakos in [55]. The motivation behind the Magnanti-Wong and modified Magnanti-Wong methods is to introduce high quality optimality cuts in an efficient manner. This allows the algorithm to converge more quickly. Another modification that is worth mentioning is introduced by Saharidis and Ierapetritou [67], which generates an optimality cut in iterations where a feasibility cut is added to the Benders' master problem. This strategy, known as maximum feasible subsystem cut generation, can be highly effective when there are a relatively large number of feasibility cuts generated by the Benders' method. Obviously, the aforementioned methodological development provides a solid basis for us to employ this analytical tool to study more comprehensive hybrid system design problems, which should generalize or extend existing work that only computes storage capacity.

Specifically, in this chapter, we consider a hybrid system design problem for an isolated area, which depends on a local thermal generator and expects an increase in electric demand in the near future. The set of options considered for the configuration include building a long distance transmission network or constructing renewable energy facilities and a storage system as shown in Figure 3. For this problem, we build a SMIP model to derive the optimal configuration with random renewable energy generation and demand. To address the computational challenges, we develop an algorithm using Benders' decomposition method with both maximum feasible subsystem cut generation and the modified Magnanti-Wong strategies. We also improve on the application of maximum feasible subsystem cut generation by taking advantage of the scenario structure of decomposed subproblems in our SMIP formulation. To our knowledge, this is the first time a stochastic discrete optimization model for comprehensive hybrid system design has been developed as well as the first effective computational tool.

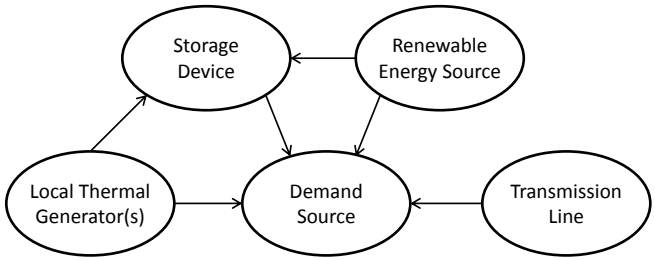


Figure 3: Transfer of Energy in a Hybrid System

3.2 Problem Description and Mathematical Formulation

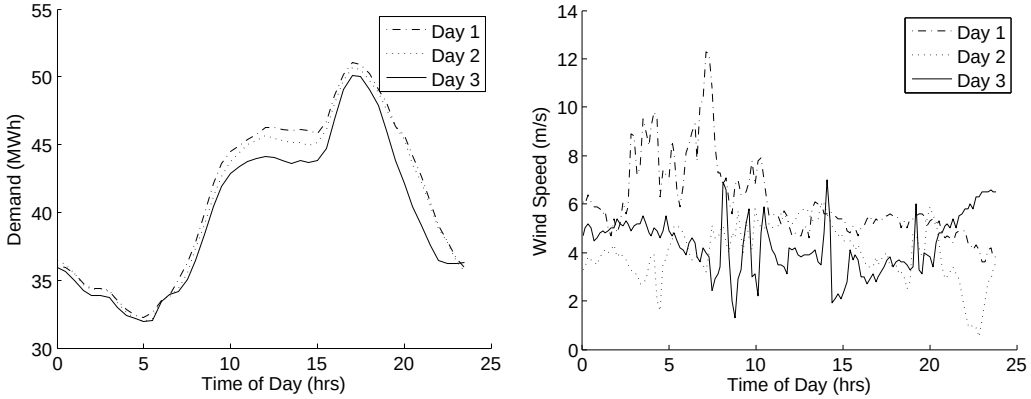
In this section, we first describe the modeling background in the hybrid system design. We then present the stochastic mixed integer programming model. Also, by observing this model is equivalent to capacitated lot-sizing problem under special circumstances, we show that this model is in general $\mathcal{NP}$-hard.

We consider the system design problem for a remote area which currently has a local thermal generator and must accommodate an increase in demand in the near future. Thus, installation and capacity decisions for the renewable energy generation (we will focus on wind generated energy, though the model is applicable to other forms of renewable energy as well), the storage device, and/or the transmission network must be made. In our study, as we focus on the long-term system configuration, we do not model the local generator's operations using the *unit commitment* model, which could drastically increase the complexity of the problem. In fact, as the remote areas generally may have large difference between their peak and minimum demands, they may install low-load diesel generators or simply force thermal generators to work with light loads [1], for which it is not necessary to incorporate the unit commitment model. Under this situation, we use a binary variable to model the working status of the local generator and to capture its fixed cost; a continuous variable is used to capture the variable generation cost.

Note that the role of a storage device is similar to that of a warehouse. It is used to store excess energy generated to meet future demands, along with just-in-time generation from the renewable energy and the local generator. One difference from classical inventory systems is that the energy loss from storage devices is often quite significant, as energy efficiency ranges from 60% (hydrogen storage) to 90% (battery storage). Although the energy loss can be high in storage devices, they are extremely useful to deal with randomness in renewable energy generation and demand.

In Figure 4(a), three daily demand curves are shown for three consecutive days (data from [50]). Figure 4(b) provides three daily wind speed curves for three consecutive days (data from [16]). From these figures, we see that the variability in hourly demand and wind speed cannot be ignored (particularly for wind speed), doing so could result in unmet demand or increased operating costs. Following the typical strategy in stochastic programming, every possible random situation is represented

by a scenario with the associated probability. Specifically, because demand (or generation) could change dramatically over seasons, one year is decomposed into a set of "seasons," denoted by I, during which daily demands (or wind speed) are reasonably consistent. Then, each season can be represented by a single day. One possible demand curve (or wind curve) of a single day is represented by a scenario. The randomness of daily demand (or generation, respectively) is represented by S, a set of scenarios, and their associated discrete probabilities. Furthermore, for modeling operations of each energy component, a single day is divided into a set of time slots T (typically 24 slots for 24 hours). Typically, the system is designed such that for every season and in every scenario: i) demand in all time slots must be met by the sum of energy from various sources; ii) the "inventory level" in the storage device is balanced with respect to inflow and outflow and energy efficiency; iii) the production level of each component of the system must be less than or equal to the capacity of that component. Tables 8 and 9 summarize the variables and parameters used in the model to meet the objective.



(a) Daily Demand Curves (b) Daily Wind Speed Curves

Figure 4: Daily Curves for Demand and Wind Speed

Table 8: Decision Variables for Model

w	number of storage capacity units purchased
g	number of wind turbines purchased
ℓ	number of transmission line capacity units purchased
x	1 if storage device is constructed
y	1 if wind turbines are erected
z	1 if transmission network is installed
$FO^s_{i,t}$	1 if local energy is used in (s,i,t)
$O^s_{i,t}$	amount of energy used to supply demand from local sources in (s,i,t)
$OS^s_{i,t}$	amount of energy put into storage from local sources in (s,i,t)
$S^s_{i,t}$	storage level in (s,i,t)
$SD^s_{i,t}$	amount of energy used to supply demand from storage in (s,i,t)
$L^s_{i,t}$	amount of energy used to supply demand from transmission line in (s,i,t)
$GS^s_{i,t}$	amount of energy put into storage from wind sources in (s,i,t)
$GD^s_{i,t}$	amount of energy used to supply demand from wind sources in (s,i,t)

Table 9: Parameters for Model

C_w	cost of units of storage capacity
C_g	cost of each wind turbine
C_ℓ	cost of units of transmission line capacity
F_w	fixed cost associated with construction of a storage device
F_g	fixed cost associated with construction of a wind farm
F_ℓ	fixed cost associated with construction of a transmission line
C_k	cost associated with storage of energy per MW per time unit
C_p	cost per MWh of energy from transmission line
C_O	operating cost of local generation facility
K_w	size of capacity units of storage
$K_g(W)$	generation capacity of each wind turbine as a function of wind speed
K_ℓ	size of capacity units of transmission line
K_O	capacity of local generation facility
M_w	big-M constant for storage
M_g	big-M constant for wind turbines
M_ℓ	big-M constant for transmission line
P	length of planning horizon
s	element of S, the set of scenarios
i	element of I, the set of seasons
t	element of T, the set of time slots
γ	conversion factor for energy put into storage
β	conversion factor from cost of energy from transmission line to cost of energy from local generation

Using these parameters and variables, the stochastic optimization model is formulated as follows:

$$\begin{aligned}\min \quad & C_w w + C_g g + C_\ell \ell + F_w x + F_g y + F_\ell z + \\ & + P\left[\mathbb{E}_s\left\{\sum_{i,t} C_k S^s_{i,t} + C_p L^s_{i,t} + C_O FO^s_{i,t} + \beta C_p\left(O^s_{i,t} + OS^s_{i,t}\right)\right\}\right] \qquad (3.1)\end{aligned}$$

$$\begin{aligned}\text{subject to} \quad & D^s_{i,t} = L^s_{i,t} + GD^s_{i,t} + SD^s_{i,t} + O^s_{i,t} && \forall s,i,t \qquad (3.2)\\ & S^s_{i,t} = S^s_{i,t-1} + \gamma GS^s_{i,t} + \gamma OS^s_{i,t} - SD^s_{i,t} && \forall s,i,t \qquad (3.3)\\ & GS^s_{i,t} + GD^s_{i,t} \le K_g\left(W^s_{i,t}\right)\, g && \forall s,i,t \qquad (3.4)\\ & S^s_{i,t} \le K_w\, w && \forall s,i,t \qquad (3.5)\\ & L^s_{i,t} \le K_\ell\, \ell && \forall s,i,t \qquad (3.6)\\ & O^s_{i,t} + OS^s_{i,t} \le K_O\, FO^s_{i,t} && \forall s,i,t \qquad (3.7)\\ & w \le M_w\, x && \qquad (3.8)\\ & g \le M_g\, y && \qquad (3.9)\\ & \ell \le M_\ell\, z && \qquad (3.10)\\ & O^s_{i,t},\ OS^s_{i,t},\ S^s_{i,t},\ SD^s_{i,t},\ L^s_{i,t}\, GS^s_{i,t}\, GD^s_{i,t} \in \mathbb{R}_+ && \forall s,i,t \qquad (3.11)\\ & FO^s_{i,t} \in \{0,1\} && \forall s,i,t \qquad (3.12)\\ & x,y,z \in \{0,1\} && \qquad (3.13)\\ & w,g,\ell \in \mathbb{Z}_+ && \qquad (3.14)\end{aligned}$$

The objective of the model is to minimize the sum of the building costs and expected operating costs over a given planning horizon P. There is a fixed cost associated with construction of a transmission line F_ℓ, a storage device F_w, and a wind farm F_g as well as an incremental cost per unit of each purchased C_ℓ, C_w, and C_g, respectively. The operating costs are the cost of storage per MW per time unit

in the storage device C_k, the cost per MWh of energy from the transmission line C_p, the fixed cost of operating the local energy source C_O, and the cost per MWh of energy from the local energy source which is captured as C_p multiplied by a factor β (typically $\beta \geq 1$). The expectation of these costs is taken over all scenarios in S. Adding the fixed costs with the expected value of the variable costs gives the objective function (3.1).

The first constraint (3.2) ensures that the demand $D^s_{i,t}$ is met by the sum of the energy from the transmission line $L^s_{i,t}$, renewable energy $GD^s_{i,t}$, energy from storage $SD^s_{i,t}$, and energy from local sources $O^s_{i,t}$. The balance of the storage level $S^s_{i,t}$ with inflow, $\gamma GS^s_{i,t}$ and $\gamma OS^s_{i,t}$ and outflow relative to the previous level is guaranteed by (3.3). Note that two assumptions are made here: i) the level in storage is already expressed in terms of dispatchable energy so that there is no loss in energy transfer from storage to the grid; ii) the discharge capacity of the storage device is equal to the capacity of the device. Constraint (3.4) gives the capacity for the total renewable energy used as a function of wind speed $K_g\left(W^s_{i,t}\right)$ and the number of wind turbines purchased g (in general, $K_g(-)$ is a function describing the output of the renewable energy system used). Figure 5 shows a typical wind power curve that defines the function $K_g(W)$. Notice that for a given wind speed, $K_g(W)$ is a constant. The storage level cannot exceed the number of storage units constructed w multiplied by the size of a storage capacity unit K_w, by (3.5). The amount of energy from the transmission line cannot exceed the number of transmission line units constructed ℓ multiplied by the capacity of a transmission line unit K_ℓ, by (3.6). $FO^s_{i,t}$ is a binary variable equal to 1 if the local generating facility is used in (s, i, t) and (3.7) ensures that local energy can be used only if the local energy production facility is operating at that time and the total usage cannot exceed the facility's capacity K_O. Additionally, x, y, and z are binary variables equal to 1 if construction of a storage device, a wind

farm, or a transmission line occurs, respectively, and zero otherwise. These are linked to w, g, and ℓ, respectively, through big-M constraints (3.8) - (3.10).

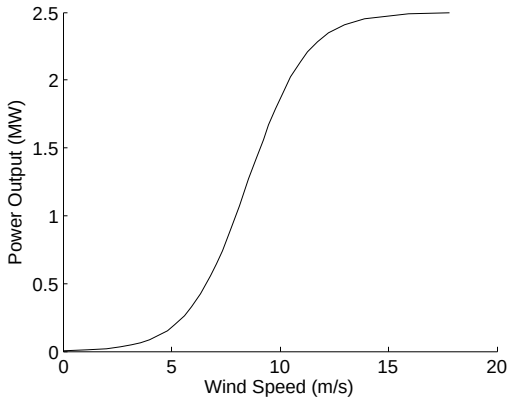


Figure 5: Wind Power Curve

Based on the role of the storage device in the whole system and the system flow balance dynamics, we observe that the whole model has a strong connection to the classical capacitated lot-sizing model [36]. In fact, we next show that the model, in its general form, has a structure of the capacitated lot-sizing model and therefore is difficult to solve.

Proposition 3. *The problem described by* (3.1)-(3.14) *is* $\mathcal{NP}$*-hard.*

Proof. We consider its simplest case where $|S| = 1$ and $|I| = 1$, i.e. a single scenario and a single season. When (i) the fixed cost of transmission line is very high, (ii) little renewable energy is available, (iii) both the fixed cost and unit cost of storage capacity are very low, it is clear that neither the transmission line will be built nor the green energy generation facility will be constructed while a sufficient storage device will be established. Therefore, the local generator will be the only generating source

to meet demand. The original model is then reduced to

$$\min \quad C_w w + F_w x + P\left[\mathbb{E}_s\left\{\sum_{i,t} C_k S^s_{i,t} + C_O F O^s_{i,t} + \beta\left(C_p O^s_{i,t} + O S^s_{i,t}\right)\right\}\right]$$

$$\text{subject to} \quad D^s_{i,t} = S D^s_{i,t} + O^s_{i,t} \tag{3.15}$$

$$S^s_{i,t} = S^s_{i,t-1} + \gamma O S^s_{i,t} - S D^s_{i,t} \tag{3.16}$$

$$O^s_{i,t} + O S^s_{i,t} \leq K_O\, F O^s_{i,t} \tag{3.17}$$

$$w \leq M_w\, x \tag{3.18}$$

$$S^s_{i,t},\ S D^s_{i,t},\ O^s_{i,t},\ O S^s_{i,t} \in \mathbb{R}_+ \tag{3.19}$$

$$w \in \mathbb{Z}^+, x \in \{0,1\} \tag{3.20}$$

Combining (3.15) and (3.16), we obtain

$$S^s_{i,t} + D^s_{i,t} = S^s_{i,t-1} + \gamma O S^s_{i,t} + O^s_{i,t},$$

which is exactly the flow balance equation in lot-sizing model if $\gamma = 1$. With the finite production capacity (3.17), this model is equivalent to the capacitated lot-sizing problem, which is known to be $\mathcal{NP}$-hard [36]. □

3.3 Solution Method

Given the structure of the problem defined by (3.1)-(3.14), we develop an algorithm using Benders' decomposition method [11], which is the most popular computation method in solving stochastic programming problems [13]. We first describe the decomposition scheme of our problem. Then, we introduce Pareto-optimal cuts using a modified Magnanti-Wong method and maximum feasible subsystem generated cuts along with their integration into Benders' decomposition. Finally, we discuss heuristic based performance improvement strategies and outline the algorithm implementation. To simplify our exposition, we do not give the scenario-wise primal and

dual subproblem formulations until sections 3.3.2 and 3.3.3, although they can easily be obtained from the aggregated subproblem formulation.

3.3.1 Benders' Decomposition

Note that our model has a clear two stage structure: i) in the system design stage, a small number of discrete decisions must be made, which are common to all seasons and stochastic scenarios; ii) in the daily operation stage, many binary decisions for local generator and continuous flow variables must be determined given the realization of renewable energy generation. To take advantage of Benders' decomposition method and to balance the decomposed subsystems, we propose to incorporate all the local generator decisions as well as capacity variables into the first stage in the solution algorithm. Specifically, given values of w^*, g^*, ℓ^*, x^*, y^*, z^*, $\left(FO^s_{i,t}\right)^*$, $\left(O^s_{i,t}\right)^*$, and $\left(OS^s_{i,t}\right)^*$ satisfying (3.7)-(3.14) we can reduce the model defined by (3.1)-(3.14) to the following primal subproblem:

$$\min \quad P\left[\mathbb{E}_s\left\{\sum_{i,t} C_k S^s_{i,t} + C_p L^s_{i,t}\right\}\right] \tag{3.21}$$

$$\text{subject to} \quad D^s_{i,t} \le L^s_{i,t} + GD^s_{i,t} + SD^s_{i,t} + \left(O^s_{i,t}\right)^* \qquad \forall s,i,t \tag{3.22}$$

$$S^s_{i,t} \le S^s_{i,t-1} + \gamma GS^s_{i,t} + \gamma\left(OS^s_{i,t}\right)^* - SD^s_{i,t} \qquad \forall s,i,t \tag{3.23}$$

$$GS^s_{i,t} + GD^s_{i,t} \le K_g\left(W^s_{i,t}\right)\, g^* \qquad \forall s,i,t \tag{3.24}$$

$$S^s_{i,t} \le K_w\, w^* \qquad \forall s,i,t \tag{3.25}$$

$$L^s_{i,t} \le K_\ell\, \ell^* \qquad \forall s,i,t \tag{3.26}$$

$$S^s_{i,t},\ SD^s_{i,t},\ L^s_{i,t}\ GS^s_{i,t}\ GD^s_{i,t} \in \mathbb{R}_+ \qquad \forall s,i,t \tag{3.27}$$

Let $\boldsymbol{\eta} = \{\eta^s_{i,t} \in \mathbb{R}_+ \ : \ \forall s,i,t\}$, $\boldsymbol{\lambda} = \{\lambda^s_{i,t} \in \mathbb{R}_+ \ : \ \forall s,i,t\}$, $\boldsymbol{\theta} = \{\theta^s_{i,t} \in \mathbb{R}_+ \ : \ \forall s,i,t\}$, $\boldsymbol{\delta} = \{\delta^s_{i,t} \in \mathbb{R}_+ \ : \ \forall s,i,t\}$, and $\boldsymbol{\alpha} = \{\alpha^s_{i,t} \in \mathbb{R}_+ \ : \ \forall s,i,t\}$ be the dual variables for constraints (3.22)-(3.26), respectively (assuming each constraint is rearranged to be of the form $\ge$, which can always be done). Then the dual of the primal subproblem,

aptly named the dual subproblem, is formulated as:

$$\begin{aligned}
\max \quad & \sum_{i,t,s} \left[-\ell^* K_\ell \alpha^s_{i,t} - w^* K_w \delta^s_{i,t} - g^* K_g \left(W^s_{i,t} \right) \theta^s_{i,t} + \right. && (3.28) \\
& \left. + \left(D^s_{i,t} - \left(O^s_{i,t} \right)^* \right) \eta^s_{i,t} - \gamma \left(OS^s_{i,t} \right)^* \lambda^s_{i,t} \right] \\
\text{subject to} \quad & \eta^s_{i,t} - \theta^s_{i,t} \leq 0 && \forall s, i, t \quad (3.29) \\
& \gamma \lambda^s_{i,t} - \theta^s_{i,t} \leq 0 && \forall s, i, t \quad (3.30) \\
& \lambda^s_{i,t+1} - \lambda^s_{i,t} - \delta^s_{i,t} \leq P\, C_k \mathbb{P}\{s\} && \forall s, i, t \quad (3.31) \\
& \eta^s_{i,t} - \lambda^s_{i,t} \leq 0 && \forall s, i, t \quad (3.32) \\
& \eta^s_{i,t} - \alpha^s_{i,t} \leq P\, C_p \mathbb{P}\{s\} && \forall s, i, t \quad (3.33) \\
& \alpha^s_{i,t}, \delta^s_{i,t}, \theta^s_{i,t}, \eta^s_{i,t}, \lambda^s_{i,t} \in \mathbb{R}_+ && \forall s, i, t \quad (3.34)
\end{aligned}$$

Notice that the feasible set for the dual subproblem, Ω, does not depend on the values of w^*, g^*, ℓ^*, x^*, y^*, z^*, $\left(FO^s_{i,t}\right)^*$, $\left(O^s_{i,t}\right)^*$, and $\left(OS^s_{i,t}\right)^*$. These values only affect the coefficients in the objective function of the dual subproblem.

Now the zero vector is in Ω, so the dual subproblem is always feasible though it may be unbounded. Therefore, the primal problem is either infeasible or feasible and bounded. Let P_Ω and R_Ω be the sets of extreme points and extreme rays of Ω, respectively. Then, the dual subproblem is bounded (and hence the primal subproblem is feasible and bounded) if

$$\begin{aligned}
\sum_{i,t,s} \Big[& - \ell^* K_\ell \alpha^s_{i,t} - w^* K_w \delta^s_{i,t} - g^* K_g \left(W^s_{i,t} \right) \theta^s_{i,t} + \\
& + \left(D^s_{i,t} - \left(O^s_{i,t} \right)^* \right) \eta^s_{i,t} - \gamma \left(OS^s_{i,t} \right)^* \lambda^s_{i,t} \Big] \leq 0
\end{aligned}$$

for all $(\boldsymbol{\eta}, \boldsymbol{\lambda}, \boldsymbol{\theta}, \boldsymbol{\delta}, \boldsymbol{\alpha}) \in R_\Omega$. Moreover, if both problems are feasible and bounded,

then they have a common value of

$$\max_{(\eta,\lambda,\theta,\delta,\alpha)\in P_\Omega} \left\{ \sum_{i,t,s} \left[-\ell^* K_\ell \alpha^s_{i,t} - w^* K_w \delta^s_{i,t} - g^* K_g \left(W^s_{i,t}\right) \theta^s_{i,t} + \right.\right.$$
$$\left.\left. + \left(D^s_{i,t} - \left(O^s_{i,t}\right)^*\right) \eta^s_{i,t} - \gamma \left(OS^s_{i,t}\right)^* \lambda^s_{i,t} \right] \right\}.$$

Let ζ be a free variable, then the Benders' Master problem is formulated as:

$$\begin{aligned}
\min \quad & \zeta + C_w w + C_g g + C_\ell \ell + F_w x + F_g y + F_\ell z + && (3.35)\\
& + P\left[\mathbb{E}_s \left\{ \sum_{i,t} C_O FO^s_{i,t} + \beta C_p \left(O^s_{i,t} + OS^s_{i,t}\right) \right\}\right] \\
\text{subject to} \quad & \sum_{i,t,s} \left[-\ell K_\ell \alpha^s_{i,t} - w K_w \delta^s_{i,t} - g K_g \left(W^s_{i,t}\right) \theta^s_{i,t} + \right. \\
& \left. + \left(D^s_{i,t} - O^s_{i,t}\right) \eta^s_{i,t} - \gamma OS^s_{i,t} \lambda^s_{i,t} \right] \leq 0 \qquad \hat{r} \in R_\Omega && (3.36)\\
& \zeta \geq \sum_{i,t,s} \left[-\ell K_\ell \alpha^s_{i,t} - w K_w \delta^s_{i,t} - g K_g \left(W^s_{i,t}\right) \theta^s_{i,t} + \right. \\
& \left. + \left(D^s_{i,t} - O^s_{i,t}\right) \eta^s_{i,t} - \gamma OS^s_{i,t} \lambda^s_{i,t} \right] \qquad \hat{e} \in P_\Omega && (3.37)\\
& O^s_{i,t} + OS^s_{i,t} \leq K_O \, FO^s_{i,t} \qquad \forall s,i,t && (3.38)\\
& w \leq M_w \, x && (3.39)\\
& g \leq M_g \, y && (3.40)\\
& \ell \leq M_\ell \, z && (3.41)\\
& O^s_{i,t},\ OS^s_{i,t} \in \mathbb{R}_+ \qquad \forall s,i,t && (3.42)\\
& FO^s_{i,t} \in \{0,1\} \qquad \forall s,i,t && (3.43)\\
& x,y,z \in \{0,1\} && (3.44)\\
& w,g,\ell \in \mathbb{Z}_+ && (3.45)\\
& \zeta \in \mathbb{R}, && (3.46)
\end{aligned}$$

where $\hat{r} = (\eta, \lambda, \theta, \delta, \alpha)$ is an extreme ray of Ω and $\hat{e} = (\eta, \lambda, \theta, \delta, \alpha)$ is an extreme

point of Ω. In general, this formulation can have a large number of constraints of the form (3.36) and (3.37), known as feasibility cuts and optimality cuts, respectively. A relaxed Benders' reformulation consists of replacing P_Ω and R_Ω with sets P'_Ω and R'_Ω in the Benders' Master problem such that $P'_\Omega \subset P_\Omega$ and $Q'_\Omega \subset Q_\Omega$. In [11], Benders describes an algorithm to iteratively add feasibility and optimality cuts to a relaxed Benders' reformulation for a mixed integer programming problem. The key initial observations for this algorithm are that an optimal solution to the relaxed Benders' reformulation provides a lower bound (LB) on the optimal value of the original problem and a feasible solution to the relaxed Benders' reformulation provides values of w^*, g^*, ℓ^*, x^*, y^*, z^*, $\left(FO^s_{i,t}\right)^*$, $\left(O^s_{i,t}\right)^*$, and $\left(OS^s_{i,t}\right)^*$ satisfying (3.7) - (3.14) which serves as the input for the dual subproblem. Next, the dual subproblem is solved. If it has a finite optimal value, then an optimality cut can be added to the relaxed Benders' reformulation and an upper bound (UB) for the original problem can be found. If the dual subproblem is unbounded, then a feasibility cut can be added to the relaxed Benders' reformulation. Resolving the relaxed Benders' reformulation restarts the process. This continues until $LB = UB$, at which point an optimal solution has been found.

There are a number of methods to improve the performance of Benders' decomposition. One commonly employed tactic for two-stage stochastic models is to decompose the dual subproblem over scenarios, since these problems are independent once the first stage variables are fixed. Thus for each $s \in S$, we have a dual subproblem, denoted DSP_s. In addition to this, there are two technical enhancements and a combination of heuristics we utilize. These are described in the next three sections along with pseudocode for our algorithm. Before this, we digress to comment on some tactics not employed in this work that are never the less still worth mentioning. The first two are cut generation strategies for the relaxed master problem. Saharidis et. al reviewed existing developments in Benders decomposition and

proposed to strengthen the master problem [66] by adding valid inequalities for applications with the fixed-charge network structure, of which unfortunately our problem lacks. Another method presented in Saharidis et. al [68] involves generating a bundle of low-density cuts rather than a single low-density cut. For the application in this paper, this effect was achieved for feasibility cuts through use of the decomposition technique mentioned at the beginning of this section as well as noting that optimality cuts were already high-density. Another set of strategies employed to improve the performance of Benders is the use of heuristic techniques. Côté and Laughton developed a set of such techniques [19]. We were able to achieve similar effects through the use of other heuristic based methods which are outlined in Section 3.3.4.

3.3.2 Generating Pareto-Optimal Cuts

Pareto-optimal cuts, introduced by Magnanti and Wong [43], are cutting planes added to the Master problem that are not dominated by any other optimality cuts. The process developed in [43], referred to as the MW method, is the following: the dual subproblem is solved, then a core point from the Master problem replaces the objective function coefficients in the subproblem and a constraint is added forcing this new objective function to equal the optimal value from the subproblem. The optimal solution to this problem generates a Pareto-optimal cut. The reliance on the solution from the subproblem to formulate the problem used to generate the Pareto-optimal cuts can be a major computational drawback. In [55], Papadakos develops a method for generating Pareto-optimal cuts that does not rely on the solution to the subproblem. In particular, he shows that any core point generates a Pareto-optimal cut and the additional constraint forcing equality of the new objective function to the optimal value of the subproblem is not necessary. Moreover, a method for updating the core point is given that is computationally very efficient. We refer to this as the modified Magnanti-Wong (MMW) method. For a single scenario in our application,

the problem for the MMW method is formulated as

$$
\begin{aligned}
\max \quad & \sum_{i,t} \left[-\ell^0 K_\ell \alpha^s_{i,t} - w^0 K_w \delta^s_{i,t} - g^0 K_g \left(W^s_{i,t} \right) \theta^s_{i,t} + \right. && (PDSP_s) \\
& \left. + \left(D^s_{i,t} - \left(O^s_{i,t} \right)^0 \right) \eta^s_{i,t} - \gamma \left(OS^s_{i,t} \right)^0 \lambda^s_{i,t} \right] \\
\text{subject to} \quad & \eta^s_{i,t} - \theta^s_{i,t} \leq 0 && \forall i, t \\
& \gamma \lambda^s_{i,t} - \theta^s_{i,t} \leq 0 && \forall i, t \\
& \lambda^s_{i,t+1} - \lambda^s_{i,t} - \delta^s_{i,t} \leq P\, C_k \mathbb{P}\{s\} && \forall i, t \\
& \eta^s_{i,t} - \lambda^s_{i,t} \leq 0 && \forall i, t \\
& \eta^s_{i,t} - \alpha^s_{i,t} \leq P\, C_p \mathbb{P}\{s\} && \forall i, t \\
& \alpha^s_{i,t}, \delta^s_{i,t}, \theta^s_{i,t}, \eta^s_{i,t}, \lambda^s_{i,t} \in \mathbb{R}_+ && \forall i, t
\end{aligned}
$$

where w^0, g^0, ℓ^0, x^0, y^0, z^0, $\left(FO^s_{i,t}\right)^0$, $\left(O^s_{i,t}\right)^0$, and $\left(OS^s_{i,t}\right)^0$ is a core point of the Master problem. It is clear in the formulation of $PDSP_s$ that this problem is independent of the dual subproblems and the solution from the Master problem. It relies only on finding a core point from the feasible space of the Master problem. Finally, an optimality cut is obtained by combining the Pareto-optimal cuts generated by each $PDSP_s$.

3.3.3 Maximum Feasible Subsystem Generated Cuts

The second strategy employed is maximum feasible subsystem generated cuts, referred to as the MFS method. Saharidis and Ierapetritou introduce this method for generating additional optimality cuts in [67]. Recall that if the dual subproblem is unbounded, then the primal subproblem is infeasible. If we can identify a collection of constraints to relax, then we are able to generate an optimality cut from the solution generated by this relaxed problem. This is the concept behind the MFS method. We now explain the process by which this collection of constraints is identified.

Notice the formulation of a primal subproblem for a single scenario is given by:

$$
\begin{aligned}
\min \quad & P\left[\mathbb{P}\{s\}\left(\sum_{i,t} C_k S^s_{i,t} + C_p L^s_{i,t}\right)\right] & \\
\text{subject to} \quad & D^s_{i,t} \le L^s_{i,t} + GD^s_{i,t} + SD^s_{i,t} + \left(O^s_{i,t}\right)^* & \forall i,t \\
& S^s_{i,t} \le S^s_{i,t-1} + \gamma GS^s_{i,t} + \gamma\left(OS^s_{i,t}\right)^* - SD^s_{i,t} & \forall i,t \\
& GS^s_{i,t} + GD^s_{i,t} \le K_g\left(W^s_{i,t}\right)\ g^* & \forall i,t \\
& S^s_{i,t} \le K_w\, w^* & \forall i,t \\
& L^s_{i,t} \le K_\ell\, \ell^* & \forall i,t \\
& S^s_{i,t},\ SD^s_{i,t},\ L^s_{i,t}\ GS^s_{i,t}\ GD^s_{i,t} \in \mathbb{R}_+. & \forall i,t
\end{aligned}
$$

If this problem is infeasible, a feasible subsystem is found by solving the feasible subsystem problem:

$$
\begin{aligned}
\min \quad & \sum_{j,i,t} \omega^j_{i,t} & (FSP_s) \\
\text{subject to} \quad & D^s_{i,t} \le L^s_{i,t} + GD^s_{i,t} + SD^s_{i,t} + \left(O^s_{i,t}\right)^* + M\omega^1_{i,t} & \forall i,t \\
& S^s_{i,t} \le S^s_{i,t-1} + \gamma GS^s_{i,t} + \gamma\left(OS^s_{i,t}\right)^* - SD^s_{i,t} + M\omega^2_{i,t} & \forall i,t \\
& GS^s_{i,t} + GD^s_{i,t} \le K_g\left(W^s_{i,t}\right)\ g^* + M\omega^3_{i,t} & \forall i,t \\
& S^s_{i,t} \le K_w\, w^* + M\omega^4_{i,t} & \forall i,t \\
& L^s_{i,t} \le K_\ell\, \ell^* + M\omega^5_{i,t} & \forall i,t \\
& S^s_{i,t},\ SD^s_{i,t},\ L^s_{i,t}\ GS^s_{i,t}\ GD^s_{i,t} \in \mathbb{R}_+ & \forall i,t \\
& \omega^j_{i,t} \in \{0,1\} & \forall i,t,j
\end{aligned}
$$

where M is a large constant such that if $w^j_{i,t} = 1$ the corresponding constraint is always satisfied. Notice that this problem is always feasible. Taking the optimal solution from FSP_s, $\{(\omega^j_{i,t})^*\}$, we generate a relaxation of the primal subproblem,

namely the maximum feasible subsystem problem

$$\begin{aligned}
\min \quad & P\left[\mathbb{P}\{s\}\left\{\sum_{i,t} C_k S^s_{i,t} + C_p L^s_{i,t}\right\}\right] && (MFSP_s)\\
\text{subject to} \quad & D^s_{i,t} \le L^s_{i,t} + GD^s_{i,t} + SD^s_{i,t} + \left(O^s_{i,t}\right)^* + M\,(\omega^1_{i,t})^* && \forall i,t\\
& S^s_{i,t} \le S^s_{i,t-1} + \gamma GS^s_{i,t} + \gamma\left(OS^s_{i,t}\right)^* - SD^s_{i,t} + M\,(\omega^2_{i,t})^* && \forall i,t\\
& GS^s_{i,t} + GD^s_{i,t} \le K_g\left(W^s_{i,t}\right)\, g^* + M\,(\omega^3_{i,t})^* && \forall i,t\\
& S^s_{i,t} \le K_w\, w^* + M\,(\omega^4_{i,t})^* && \forall i,t\\
& L^s_{i,t} \le K_\ell\, \ell^* + M\,(\omega^5_{i,t})^* && \forall i,t\\
& S^s_{i,t},\ SD^s_{i,t},\ L^s_{i,t}\, GS^s_{i,t}\, GD^s_{i,t} \in \mathbb{R}_+ && \forall i,t
\end{aligned}$$

Since $MFSP_s$ is feasible and a relaxation of the primal problem, we can solve the dual of this problem to generate an optimality cut for the relaxed Master problem. It is clear how the optimality cut is constructed if DSP_s is unbounded for all s (equivalently, if all the primal subproblems are infeasible), namely adding the resulting cutting planes. Our contribution is dealing with the case where there is a set $\hat{S} \subseteq S$ such that DSP_s is unbounded for $s \in \hat{S}$ and bounded for $s \in S \setminus \hat{S}$. In this case, we solve FSP_s followed by $MFSP_s$ for each $s \in \hat{S}$ to generate a collection of cutting planes, $\hat{\mathcal{P}}$. Next, let $\mathcal{P}$ by the collection of cutting planes generated using the traditional Benders' method by solving DSP_s for $s \in S \setminus \hat{S}$. Adding the cutting planes from $\hat{\mathcal{P}}$ and $\mathcal{P}$ results in a single optimality cut for the Master problem. Note if $\hat{S} = \emptyset$, then this results in a traditional Benders' optimality cut being added to the Master problem.

3.3.4 Heuristic Improvements and Outline of the Algorithm

In addition to the enhanced cut generation methods, two heuristics are used to improve the computation time of our solution algorithm. Before we describe these heuristics it is notable that the techniques mentioned in this section do not alter the

convergence of the algorithm, only the rate. That is, the algorithm will converge to the optimal value, but the time and number of iterations necessary most likely differs from the time and number of iterations needed were these techniques not employed.

First, we obtain an initial feasible solution to the complete model by rounding its linear programming solution to a closest feasible solution [48]. Then, the partial solution for the master problem will be passed to the dual subproblem to generate feasibility cuts or an optimality cut of a high quality. Also, its objective function value provides a lower bound on the optimal value. Note that any values for the Master problem variables can be used to initiate Benders' decomposition, however a partial solution from a solution feasible to the original problem will generate an optimality cut rather than a feasibility cut. Moreover, the closer this solution is to optimal, the higher the quality of the first cut generated. Since the rounded linear programming solution can be obtained quickly and is often good quality, we use this method for generating initial value for Benders' decomposition.

Second, we observe that the master problem is a difficult mixed integer programming problem and any of its feasible solutions can be used to generate optimality or feasibility cuts. As a result, we use two techniques to speed up the initial iterations of the algorithm. One is to relax the tolerance for a solution to be considered optimal by the commercial solver until the relative gap for the algorithm is sufficiently small. In this case, the default gap is 10^{-4}, which we initially adjust to 10^{-1}. Once the relative gap for the algorithm is 10^{-2}, the solver tolerance is returned to 10^{-4}. Additionally, a time limit is imposed on the Master problem. It is initially set at 60 seconds. Once the relative gap for the algorithm is 10^{-2}, an additional 30 seconds is added if the gap for the Master problem is at least 0.0005. The best feasible solution from the Master problem is always the one passed to the subproblems.

We conclude this section by providing the pseudocode for the Benders' algorithm implemented for this work in Table 10. We remind the reader of some notation used

in the table. For scenario s, the dual subproblem is given by DSP_s, the problem used for the MMW method is denoted $PDSP_s$, and FSP_s and $PFSP_s$ are the feasible subsystem problem and maximum feasible subsystem problem, respectively, used in the MFS method.

Table 10: Pseudocode for Solution Algorithm

Line	Code Description
1	Solve the LP relaxation of the original problem and
2	round integer variables up to obtain a feasible
3	solution for relaxed Master problem
4	**while** relative gap greater than ϵ
5	**for** $s \in S$
6	Solve DSP_s and $PDSP_s$
7	**end for**
8	**if** DSP_s is bounded for all $s \in S$
9	Add a traditional Benders' optimality cut to
10	relaxed Master problem
11	**else**
12	Let $S' = \{s \mid DSP_s$ is unbounded$\}$
13	**for** $s \in S'$
14	Add feasibility cut for extreme ray from DSP_s
15	Solve FSP_s then $MFSP_s$
16	**end for**
17	Use solutions from $MFSP_{s'}$ and DSP_s for $s' \in S'$
18	and $s \in S \setminus S'$ to add a MFS generated
19	optimality cut to the relaxed Master problem
20	**end if**
21	Use solutions from $PDSP_s$ for $s \in S$ to generate Pareto cut
22	**end while**

3.4 Computational Study and Management Insights

3.4.1 Data Description

Demand data were collected from [50] and the wind data from [16]. The demand data used were specifically from England and Wales for 2009. The data were scaled down by a factor of 10^{-3}. The population of England and Wales is approximately 53M [25], so we have an estimated population of 53,000 for our rescaled demand data. Twelve seasons were used, one for each month. Scenarios were created by perturbing

the observed data. A total of 55 datasets were created for numerical study. We summarize them into three categories based on their purpose. These categories are i) comparison to professional solver, ii) impact of variance, and iii) impact of magnitude. Within each dataset, there are an equal number of demand and wind scenarios. This is indicated in the descriptions below by using the notation $a \times a$ for the total number of scenarios in a dataset. The datasets in each category are described forthwith:

1. To compare the performance of our algorithm with a professional solver, 15 datasets of various sizes were generated. For demand scenarios, in a given time slot scenario data were generated from a normal distribution with mean equal to observed demand and standard deviation equal to 10 percent of the observed data. For the wind scenarios, in a given time slot the scenario data were generated by multiplying the observed wind speed by a random number drawn from a triangular distribution with parameters $(0.5, 1, 1.5)$. Within each dataset, an equal number of wind and demand scenarios were used. There are five datasets with 5×5 scenarios (S1-S5), five with 8×8 scenarios (M1-M5), and five with 11×11 scenarios (L1-L5). S, M, and L indicate small, medium, and large dataset sizes, respectively.

2. To compare the system configuration in situations with different variance, another 20 datasets were generated, all with 8×8 scenarios. Five dataset (DV1-DV5) were generated similar to first 15, but with 25 percent rather than 10 percent for demand standard deviation followed by five datasets with 64 scenarios (LDV1-LDV5) with 5 percent for demand standard deviation. Five datasets (WV1-WV5) were generated similar to the original 15 datasets, but with the parameters for the triangular distribution being $(0.25, 1, 1.75)$, then another five datasets (LWV1-LWV5) with triangular distribution parameters of $(0.75, 1, 1.25)$. DV and LDV indicate high de-

mand variance and low demand variance, while WV and LWV indicate high wind variance and low wind variance.

3. To evaluate the effect of rescaling the magnitude of wind and demand an additional 20 datasets with 8×8 scenarios were generated. Five datasets (TD1-TD5) were created by doubling original observed demand data and following the procedure for the original 15 datasets. Five datasets (TW1-TW5) were created by doubling original observed wind data and following the procedure for the original 15 datasets. Five datasets (HD1-HD5) were created by multiplying original observed demand data by one half and following the procedure for the original 15 datasets. Five datasets (HW1-HW5) were created by multiplying original observed wind data by one half and following the procedure for the original 15 datasets. TD and TW are datasets with twice demand and wind speed, respectively. HD and HW are datasets with half demand and wind speed, respectively.

The cost coefficients for our model were estimated based on information from engineers at a local utility company, as well as [52] and [24]. We describe the method by which they were determined here. A 600MW line was constructed from New Jersey to Long Island for \$600M, so to determine F_ℓ and C_ℓ we used this pricing information. Specifically, it was assumed that a 60 MW line (large enough for our demand data) would cost \$60M. Assuming 75% fixed cost and K_ℓ (the size of capacity units) is 20 MW, we have F_ℓ = \$45M and C_ℓ = \$5M. A number of storage devices were build in the United States for costs ranging from \$0.16M to \$0.65M per MW. Assuming the capacity is 200MW (large enough to store roughly three times the maximum demand) and the end cost is \$0.5M per MW, then using the same method as above with $K_w = 5$MW we have F_w = \$75M and C_w = \$0.625M. Finally, assuming a wind turbine's end cost is \$1.5M and that 250 turbines are purchased, then F_g = \$280M and C_g = \$0.38M. High fixed costs for storage and wind farm construction were used

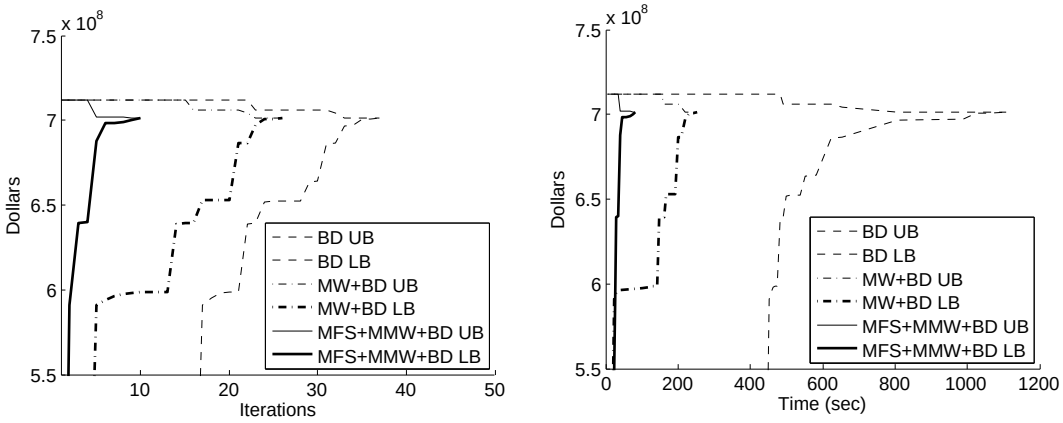
since we are considering pump storage, would require the construction of a reservoir, and an offshore wind farm, which also has considerable construction costs. The cost of storage C_k was set at \$0.1 per MW per hour in storage. We assumed the hourly operating cost of the local generating facilities C_O to be \$2000. The cost of energy in Tampa is \$100 per MWh, so this was the value used for C_p and energy from local sources was assumed to be three times as much ($\beta = 3$). The conversion factor for the energy placed into storage was taken to be 75% ($\gamma = 0.75$), and K_O was set to be 75% of peak demand. Finally, each scenario was weighted equally. For a sufficiently large number of scenarios, the true distribution will be approximated with little loss. Therefore, this is not an unreasonable assumption.

3.4.2 Traditional vs. Enhanced Benders' Decomposition

To see the benefit of utilizing Pareto-optimal cuts and MFS generated cuts in Benders' decomposition, the following experiment was conducted. The problem described by (3.1)-(3.14) with dataset M2 was solved using standard Benders' decomposition, Benders' with MW, and Benders' with MMW and MFS. In Figure 6(a), the upper and lower bounds for all three algorithms are depicted as a function of iteration number. Figure 6(b) shows the bounds for these three algorithms versus time. These figures depict how using MMW and MFS concurrently can vastly reduce the computation time for this model as compared to traditional Benders' and Benders' with MW. Therefore, in all our computation experiments, we implement MMW and MFS within the Benders' decomposition algorithm.

3.4.3 Experimental Results

The problem described by (3.1)-(3.14) was solved with datasets S1-S5, M1-M5, and L1-L5 using Benders' decomposition with Pareto-optimal cuts (BD) and with a commercial solver. This was done on a computer running Windows 7 (64-bit) with a 3GHz processor and 4 GB of RAM. The solver used was Gurobi 4.5.1 (64-bit) [53],



(a) Bounds on Optimal Solution vs Iteration (b) Bounds on Optimal Solution vs Time

Figure 6: Progression of Bounds on Optimal Solution for Benders' Decomposition with and without Enhancements

one of the state-of-the-art solvers on the market, through Python 2.6.6. Table 11 summarizes the results of solving these 15 problems given the above configuration. Table 12 provides the average values of the results with respect to problem size. The percent gap is defined as $100(UB - LB)/LB$. The algorithm tolerance was set at $10^{-2}\%$, which is the same value as Gurobi. We note that that Gurobi uses $100(UB - LB)/UB$ as default to define its percent gap, which is not the value given in Table 11, both % Gap columns are $100(UB - LB)/LB$. In general, it is true that $(UB - LB)/UB \leq (UB - LB)/LB$, so the gap utilized in our application of Benders' decomposition is stronger than Gurobi's. In all experiments, we set the computation time limit to 1 hour.

For small instances (S1-S5), our algorithm was roughly twice as fast as the commercial solver. As the problem size grows, the Benders' decomposition algorithm outperforms Gurobi by a larger margin. For medium size instances with 8×8 scenarios, our algorithm generally is 5 times faster than Gurobi. For all large size instances with 11×11 scenarios, Gurobi fails to obtain an optimal solution within the 1 hour time limit. In fact, Gurobi fails to close the gap to a reasonable value in all large instances before its termination.

Table 11: Summary of Results

Dataset	Gurobi % Gap	BD % Gap	Gurobi Time (s)	BD Time (s)	Its.	# Feas. Cuts	# Opt. Cuts
S1	0	0.0067	182.90	72.43	12	120	24
S2	0	0.0012	279.00	85.03	14	125	28
S3	0	0.0022	296.06	171.48	17	184	34
S4	0.0001	0.0049	176.97	161.94	15	174	30
S5	0	0.0022	183.92	96.05	15	181	30
M1	0	0.0056	1361.20	296.50	12	303	24
M2	0	0.0089	2631.15	306.49	11	256	22
M3	0.0001	0.0009	2351.80	551.20	17	512	34
M4	0	0.0061	1896.71	461.74	15	417	30
M5	0	0.0033	1596.27	347.02	15	366	30
L1	39.18	0.0023	3609.73*	1146.89	15	779	30
L2	39.13	0.0019	3610.35*	999.81	12	592	24
L3	38.36	0.0003	3610.62*	1576.71	17	858	34
L4	38.98	0.0036	3613.15*	1488.40	16	911	32
L5	36.44	0.0068	3610.40*	870.10	11	439	22

* indicates time limit met

Table 12: Average Values of Results by Problem Size

Dataset Size	Gurobi % Gap	BD % Gap	Gurobi Time (s)	BD Time (s)	Its.	# Feas. Cuts	# Opt. Cuts
S (5×5)	0.0000	0.0035	223.77	117.39	14.6	156.8	29.2
M (8×8)	0.0000	0.0050	1967.43	392.59	14	370.8	28
L (11×11)	38.42*	0.0030	3610.85*	1216.38	14.20	715.80	28.40

* indicates time limit met

A variety of situations were considered to determine the model's sensitivity to the data. We first considered the effects of variance in wind speed and demand on the configuration of the system in the optimal solution. This was done through the use of datasets WV1-WV5, DV1-DV5, LWV1-LWV5, and LDV1-LDV5. The results are shown in Table 13. The columns of Table 13 are as following: Dataset is the name of the dataset used, Obj. Val. is the objective value found, Iterations is the number of iterations needed to terminate the algorithm, # Feas. Cuts is the number of feasibility cuts added, # Opt. Cuts is the number of optimality cuts added, %

Local is the percentage of demand met by using locally generated energy, % Line is the percentage of demand met from the transmission line, and % Green is percentage of demand met from renewable energy. The percentage used in the last three columns is against all demand. For example, % Line is given by $\left(\sum_{s,i,t} L^s_{i,t}\right) / \left(\sum_{s,i,t} D^s_{i,t}\right)$, and the other two are computed similarly.

Table 13: Effect of Variance on System Configuration

Dataset	Obj. Val.	Its.	# Feas. Cuts	# Opt. Cuts	% Local	% Line	% Green
WV1	705543364.38	21	398	42	0.00	18.37	81.63
WV2	711512366.08	19	329	38	0.00	18.45	81.55
WV3	697301448.26	19	379	38	0.00	16.70	83.30
WV4	708697031.69	27	575	54	0.00	17.91	82.09
WV5	714032794.20	18	331	36	0.00	100.01	-0.01
DV1	712356936.58	12	321	24	0.01	24.38	75.61
DV2	708987196.99	17	513	34	0.03	28.26	71.71
DV3	717145177.53	17	440	34	0.04	29.27	70.69
DV4	711156664.84	21	763	42	0.03	28.92	71.05
DV5	713030456.73	17	525	34	0.01	28.91	71.08
LWV1	693585938.08	13	361	26	0.00	26.69	73.31
LWV2	689887229.76	16	414	32	0.00	25.46	74.54
LWV3	694491197.06	17	555	34	0.00	26.45	73.55
LWV4	691217524.44	15	405	30	0.00	25.86	74.14
LWV5	692841574.63	14	435	28	0.00	26.29	73.71
LDV1	700641441.37	17	464	34	0.00	27.73	72.27
LDV2	705779582.41	15	571	30	0.00	27.95	72.05
LDV3	699731192.17	17	463	34	0.04	25.61	74.35
LDV4	703307059.40	12	244	24	0.00	28.73	71.27
LDV5	703804593.78	16	443	32	0.00	27.63	72.37
M1	705944101.51	12	303	24	0.00	28.79	71.21
M2	701134559.45	11	256	22	0.00	27.51	72.49
M3	710613595.89	17	512	34	0.00	28.92	71.08
M4	699706556.33	15	417	30	0.00	27.89	72.11
M5	701073419.76	15	366	30	0.00	26.93	73.07

There is little effect of variance on the optimal solution value, with the exception of low wind variance in which case the optimal value was lowered slightly. As shown in Table 13, the optimal configuration of system changes with variance. In particular, high wind variance impacts the percent energy from various sources. This observation

could be explained by the introduction of a storage device in the configuration of the system. As a direct result, the percentage of demand met by green energy is significantly higher than the other cases (with the exception of WV5). This indicates that when there is enough wind at a higher speeds, it is worth while to store the excess energy. High variance in demand does not change the mixture of energy sources. Interestingly, local energy production is utilized in the presence of high demand variance. This indicates that the local generator will be used as the backup to mitigate the impact of variance. Overall, comparing variances in demand and wind, the latter one has more significant impact on system design.

Table 14: Average Values of Effect of Variance by Data Type

Dataset Type	Obj. Val.	Its.	# Feas. Cuts	# Opt. Cuts	% Local	% Line	% Green
WV	707417400.92	20.8	402.4	41.6	0.00	34.29	65.71
DV	712535286.53	16.8	512.4	33.6	0.02	27.95	72.03
LWV	692404692.79	15.0	434.0	30.0	0.00	26.15	73.85
LDV	702652773.83	15.4	437.0	30.8	0.01	27.53	72.46
M	703694446.59	14.0	370.8	28.0	0.00	28.01	71.99

Datasets TW1-TW5, TD1-TD5, HW1-HW5, and HD1-HD5 were used to examine the effects of rescaling the demand data and wind speed data on the optimal configuration of the energy system. Table 15 shows the results of these trials, and Table 16 provides the average values with respect to dataset type. Doubling the initial wind speed data results in a significant cost savings and increases green generation. Doubling the initial demand data creates variability in the optimal system configuration, though renewable energy tends to remain a significant portion of generation as reflected in Table 16. When demand is low, the transmission line could provide an economic advantage. Nevertheless, with demand increases, renewable energy generation facilities will become economically beneficial. In fact, this benefit is clearer as demand increases.

Table 15: Effects of Rescaling on Optimal Solution

Dataset	Obj. Val.	Its.	# Feas. Cuts	# Opt. Cuts	% Local	% Line	% Green
TW1	489385804.37	15	419	30	0.00	11.68	88.32
TW2	487143119.92	15	403	30	0.00	11.89	88.11
TW3	487332862.21	14	438	28	0.00	11.39	88.61
TW4	485731271.35	8	192	16	0.00	11.58	88.42
TW5	489612058.43	19	692	38	0.00	11.73	88.27
TD1	1025436900.65	21	302	42	0.00	18.73	81.27
TD2	1026781075.16	23	391	46	0.00	99.58	0.42
TD3	1018100784.25	20	318	40	0.00	19.21	80.79
TD4	1021423266.13	20	351	40	0.00	17.91	82.09
TD5	1032086992.68	19	336	38	0.00	99.93	0.07
HW1	724589905.39	10	240	20	0.10	99.90	0.00
HW2	720368367.77	6	144	12	0.04	99.96	0.00
HW3	722067291.05	12	216	24	0.05	99.95	0.00
HW4	724059893.31	9	160	18	0.05	99.95	0.00
HW5	726309583.98	10	232	20	0.05	99.95	0.00
HD1	385781244.25	8	320	16	0.00	100.00	0.00
HD2	385084899.26	8	320	16	0.00	100.00	0.00
HD3	383586693.10	9	384	18	0.00	100.00	0.00
HD4	385436764.32	9	320	18	0.00	100.00	0.00
HD5	384947960.05	12	512	24	0.00	100.00	0.00
M1	705944101.51	12	303	24	0.00	28.79	71.21
M2	701134559.45	11	256	22	0.00	27.51	72.49
M3	710613595.89	17	512	34	0.00	28.92	71.08
M4	699706556.33	15	417	30	0.00	27.89	72.11
M5	701073419.76	15	366	30	0.00	26.93	73.07

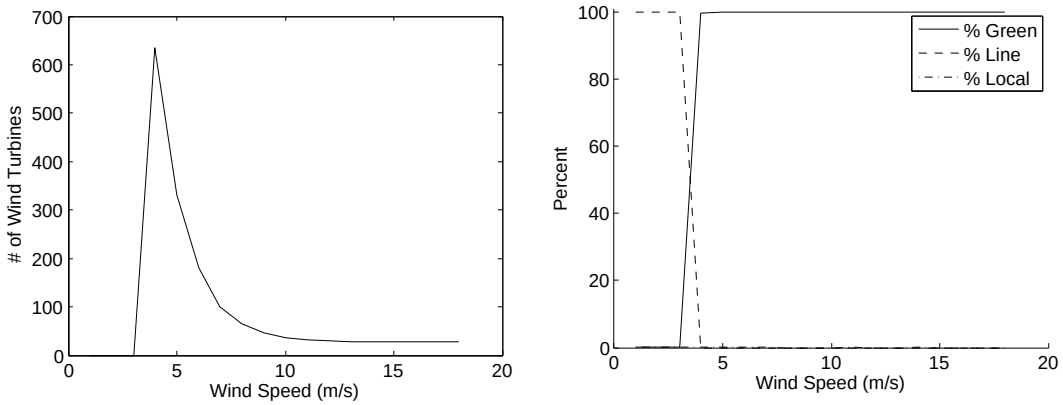
Table 16: Average Values of Effect of Rescaling by Data Type

Dataset Type	Obj. Val.	Its.	# Feas. Cuts	# Opt. Cuts	% Local	% Line	% Green
TW	487841023.26	14.2	428.8	28.4	0.00	11.65	88.35
TD	1024765803.77	20.6	339.6	41.2	0.00	51.07	48.93
HW	723479008.30	9.4	198.4	18.8	0.06	99.94	0.00
HD	384967512.20	9.2	371.2	18.4	0.00	100.00	0.00
M	703694446.59	14.0	370.8	28.0	0.00	28.01	71.99

Finally, the numerical study shows that the changes in the magnitudes of wind or demand changes the computational complexity as well. The larger the magnitude of demand (or wind), the larger the number of Benders' iterations required.

3.4.4 Additional Insights

It is to be expected that the yearly average wind speed could provide some insight to the expected number of wind turbines to purchase (if any). To explore this, wind scenarios were eliminated from dataset M1. The wind in every time slot was then replaced with values ranging from 1 to 18 for every season and scenario. The problem defined by (3.1)-(3.14) was solved for each wind speed from 1 to 18. Figure 7(a) shows the number of wind turbines purchased in the optimal solution versus wind speed. In Figure 7(b), the percent of green, line, and local energy (as defined in section 3.4.3) in the optimal solution are given as a function of wind speed. These figures indicate there is a critical wind speed that determines whether wind turbines are purchased. Additionally, there is a transition from energy delivered by the transmission line to energy delivered by the wind farm.



(a) Number of Wind Turbines vs. Wind Speed (b) % Generation vs. Wind Speed

Figure 7: Effects of Average Wind Speed on Optimal Solution

3.5 Conclusions and Future Research in Energy

In this chapter, we developed a stochastic mixed integer programming model to determine the optimal configuration of a hybrid system consisting of a renewable energy facility, storage device, long distance transmission lines, and a local energy facility. Additionally, an efficient algorithm utilizing Benders' decomposition was

created. It was found that the algorithm outperforms a solver (Gurobi) consistently for all problem sizes tested. Although the model was applied to wind energy, it is applicable to other forms of renewable energy such as solar and tides. To the best of our knowledge, this is the first time that the comprehensive hybrid system configuration problem is modeled by a stochastic integer program and solved by an efficient algorithm. Possible areas for future work include expanding the model to consider multiple renewable sources simultaneously, the unit commitment problem for local generator(s), and allowing construction at different times in the planning horizon along with associated lead times.

4 Palliative Chemotherapy Planning

In this chapter, a process for determining the optimal sequence of treatment decisions for palliative chemotherapy in late stage breast cancer patients is described. This chapter is organized as follows: we begin with an introduction to breast cancer including information on the staging process and treatment options, Section 4.2 describes the process by which patient data was analyzed in order to link patient attributes to response to treatment, next we describe a Markov decision process model for palliative chemotherapy in Section 4.3, and finally the chapter concludes with a discussion of the implications of this work as well as future research directions.

4.1 Introduction to Breast Cancer

Cancer is a term that describes a group of diseases. These diseases are all characterized by the presence of abnormal cells in the body that divide without control and invade other tissues. Broadly speaking, there are five categories of cancer:

1. Carcinoma - cancer that begins in the skin or in tissues that line or cover internal organs.

2. Sarcoma - cancer that begins in bone, cartilage, fat, muscle, blood vessels, or other connective or supportive tissue.

3. Leukemia - cancer that starts in blood-forming tissue, such as the bone marrow, and causes large numbers of abnormal blood cells to be produced and enter the blood.

4. Lymphoma and myeloma - cancers that begin in the cells of the immune system.

5. Central nervous system cancers - cancers that begin in the tissues of the brain and spinal cord.

Tumors, on the other hand, are clusters of abnormal cells in the body. Not all tumors are cancerous. Cancerous tumors are known as malignant, while noncancerous tumors are referred to as benign. Certain types of cancer do not form tumors. For example, in leukemia the abnormal cells do not cluster but rather travel throughout the bloodstream.

As of January 1, 2008, nearly 12 million Americans had been diagnosed with cancer at some point in their life [34]. That is approximately 3.8% of the total population. Nearly a quarter of these individuals were women diagnosed with breast cancer [34]. Breast cancer most commonly develops in either the inner lining of milk ducts or the lobules that supply the ducts with milk. It is referred to as either a ductal or lobular carcinoma depending on the tissue of origin. Once cancer develops in the breast, it can be carried to other parts of the body through the transmission of cancer cells that first are passed to lymph nodes through lymph vessels in the breast. From the lymph nodes, it is possible for cancer cells to enter the bloodstream and be transferred to other tissues in the body.

Breast cancer is classified depending on the progress of the disease. By establishing the primary tumor characteristics (T), nearby lymph node involvement (N), and the metastatic status (M), the progress of the disease is determined [73]. Once the disease is classified by the T, N, and M categories, it is put into a *stage grouping*, commonly referred to as the stage. There are four stages, three of which have subcategories, which are I, IIA, IIB, IIIA, IIIB, IIIC, and IV. An explanation of these classifiers and their relationship to the stages is given in Table 17. One common measure of the severity of these stages is the 5-year survival rate. This is the proportion of patients that live at least five years from date of diagnosis. Table 18 provides the 5-year survival rate by stage and the proportion of the breast cancer by regional

involvement. Notice that stage IV patients have an extremely low 5-year survival rate, which is the reason we focus our attention on this group.

Table 17: Notation for Classification of Breast Cancer into Stages

Class	Cat.	Characteristics	Stage I	IIA	IIB	IIIA	IIIB	IIIC	IV
T	X	Primary tumor cannot be assessed							
	0	No evidence of primary tumor							
	is	Carcinoma in situ		√					
	1	Tumor is 2 cm or less across	√			X			
	2	Tumor is more than 2 cm but not more than 5 cm across		X	√			√	√
	3	Tumor is more than 5 cm across			X	√			
	4	Tumor of any size growing into the chest wall or skin - includes inflammatory breast cancer					√		
N	X	Nearby lymph nodes cannot be assessed							
	0	Cancer has not spread to nearby lymph nodes	√	X	X				
	1	Cancer has spread to 1 to 3 axillary (underarm) lymph node(s), and/or tiny amounts of cancer are found in internal mammary lymph nodes (those near the breast bone) on sentinel lymph node biopsy		√	√	√	√		√
	2	Cancer has spread to 4 to 9 lymph nodes under the arm, or cancer has enlarged the internal mammary lymph nodes				√, X			
	3	involvement in at least 10 axillary lymph nodes or in lymph nodes above or under the clavicle with at least one area larger than 2mm in either case						√	
M	X	Presence of distant spread (metastasis) cannot be assessed							
	0	No distant spread is found on x-rays (or other imaging procedures) or by physical exam	√	√, X	√, X	√, X	√	√	
	1	Spread to distant organs is present							√

Table 18: Distribution of the Stages of Breast Cancer

Stage	5-year Survival Rate [74]	Proportion [34]
I	88%	60%
IIA	81%	
IIB	74%	
IIIA	67%	33%
IIIB	41%	
IIIC	49%	
IV	15%	5%

Once a patient's disease is staged, treatment is determined. There are five types of treatment for breast cancer: surgery, targeted therapy, radiation therapy, hormone

therapy, and chemotherapy [32]. Surgery can be breast-conserving, which consists of lumpectomy where a tumor is removed along with a small amount of surrounding healthy tissue or partial mastectomy where a portion of the breast is removed, or it can be a total mastectomy, where the entire breast is removed. Targeted therapy involves the use of drugs or substances that attack cancerous cells without harming normal cells.

Before discussing the other treatment options, it is necessary to define the terms neoadjuvant, adjuvant, and palliative. Neoadjuvant care is treatment given *prior* to primary care to reduce tumor load. Adjuvant care is treatment given *after* primary treatment to reduce the risk of relapsing. Last, palliative care is treatment given without curative intent in order to reduce tumor load and increase the quality of life in terminal patients. We now return to a description of treatment options.

Radiation therapy uses of high energy X-rays or other types of radiation to kill cancer cells or keep them from growing. Hormone therapy is a treatment that removes hormones or blocks their action and stops cancer cells from growing. Chemotherapy is treatment that uses drugs to stop the growth of cancer cells, either by killing the cells or by stopping them from dividing. These three treatments can each be given as neoadjuvant, adjuvant, curative, or palliative care. Table 19 presents the typical treatment options by stage, where "Neo/Adjuvant" indicates treatment can be either neoadjuvant or adjuvant, "Primary" indicates this is the primary treatment intended to eradicate the disease, and "Curative" indicates systemic treatment intended to eliminate the disease. Targeted therapy is not mentioned in the table as it is a treatment option for all stages, however it is dependent on the disease properties. For instance, there is a type of breast cancer known as "triple-negative." For this type of breast cancer, there is a type of targeted treatment that uses PARP inhibitors, which is a substance that blocks key enzymes in cancer cells to prevent cellular reproduction [32].

Table 19: Treatment Options for Breast Cancer Patients

Stage	Surgery	Hormone Therapy	Radiation Therapy	Chemo-therapy
I	Primary	Neo/Adjuvant	Neo/Adjuvant	Neo/Adjuvant
II	Primary	Neo/Adjuvant	Neo/Adjuvant	Neo/Adjuvant
IIIA	Primary	Neo/Adjuvant	Neo/Adjuvant	Neo/Adjuvant
IIIB	Yes	Curative	As Needed	Curative
IIIC Operable	Primary	Neo/Adjuvant	Neo/Adjuvant	Neo/Adjuvant
IIIC Inoperable	No	Curative	As Needed	Curative
IV	Yes	Curative/ Palliative	As Needed	Curative/ Palliative

Although stage IV breast cancer can be treated with curative intent, this is generally not the case since the 5-year survival rate is only 15%. Thus, treatment for this stage is often systemic and can last for many years. Given that the human mind can only track the effects of decisions so far into the future and the fact that each patient reacts differently to chemotherapy, it is of vital importance to provide oncologists with tools that incorporate the long-term effects and randomness associated with sequential treatment decisions like those present in palliative chemotherapy. This research aims to develop such a mechanism using stochastic optimization with information garnered from the identification of patient characteristics that can be used to predict responsiveness to treatment.

4.2 Predicting Patient Response to Chemotherapy

To improve the quality of long-term chemotherapy care, it is necessary to classify patients according to how they will respond to treatment. In addition, given that chemotherapy works by killing cells that divide rapidly, including healthy cells that grow rapidly under normal circumstances, such as bone marrow cells, digestive tract cells, and hair follicles, it is of utmost importance to administer chemotherapy in a manner which minimizes the risk to the patient. To that end, this research aims to

develop a model for predicting a stage IV breast cancer patient's response to palliative chemotherapy based on information available in typical private oncology practice's electronic medical records (EMRs).

A line of chemotherapy, also known as a protocol or regimen, is defined by a combination of drugs, dosage levels, a fine administration schedule, and a coarse administration schedule. This coarse schedule consists of cycles which dictate the beginning of each fine administration schedule. Ideally, these cycles are timed to attack cancer cells when they are most vulnerable.

There are periods in between cycles that allow patients to recover from treatment. At the beginning of each cycle, the patient must be evaluated to determine the best course of action. That is, continue current line, switch to a new line, or stop treatment altogether. For neoadjuvant and adjuvant chemotherapy, which are typically used for early stages of breast cancer, a fixed number of cycles are administered, so the toxicity effects, i.e., adverse reaction of healthy cells to treatment, are less of a concern than in long-term treatment. Thus a patient receiving (neo)adjuvant chemotherapy will typically finish all prescribed cycles, unless there is an extreme adverse reaction to the treatment. On the other hand, the decision made at the beginning of each cycle of palliative chemotherapy requires a more careful analysis.

Recall that palliative chemotherapy is reserved for patients with a terminal diagnosis, this is usually only stage IV. Thus, the purpose is to improve the patient's quality of life by reducing tumor load, but is not intended to eliminate the disease. At this stage, palliative chemotherapy will be given as long as the patient's benefits out weight the side effects. The benefits of treatment are measured through tumor response to treatment as defined by the RECIST guidelines [22], which involves measuring present tumors and determining the relative change in size.

Given that this is a disease-oriented definition of response, it does not capture any information about patient response to treatment. Therefore, we defined response

to treatment from a clinical standpoint to incorporate both the disease and patient responses. For example, due to the nature of chemotherapy, a patient may suffer from toxicity in which case the patient would be considered as not responding to treatment. Based on both, the disease and the patient response, treatment would be stopped for one of two reasons: i) cancer is progressing with current line of treatment (i.e., negative disease response); ii) the patient is experiencing significant side effects from the current line of treatment (i.e., negative patient response).

According to a leading oncologist, if a patient remains on a line of treatment for at least two months, neither of the above items occurred [20]. Based on this, we adopt the convention that if first-line chemotherapy lasts at least 50 days, then this would be classified as a positive response to treatment. From this point, patient response to treatment is based on this definition.

This research focuses on identifying a subset of commonly collected vital signs and laboratory results (collectively referred to as labs) that can be used to predict a patient's response to chemotherapy. This is accomplished through the use of logistic regression.

Logistic regression is a powerful tool for predicting dichotomous outcomes as a function with multiple inputs [29]. It can be particularly useful for predicting the disease state of a patient as well as determination of yes/no decisions [7]. Logistic regression has been a successful tool for classifications relating to cancer. In [18], Chhatwal et al. developed two models for predicting breast cancer risk based on the descriptors of the National Mammography Database. Moreover, the factors impacting patient mortality and transferring were explored in [78] by Zhang et al. through the use of logistic regression.

This work will provide insight to how late-stage breast cancer patients react to long term treatment. Due to the low 5 year survival rate for stage IV breast cancer patients, approximately 15% [33], adding to this body of knowledge is of substantial

importance. Moreover, the intent is to build on this work by developing tools to improve the quality of palliative care through the use of stochastic modeling (see Section 4.3).

4.2.1 Methods for Predicting Response to Chemotherapy

For this research, data were provided by a large medical oncology practice in the mid-west. EMOL Health [27] manages the database that houses this practice's EMRs as well as text files of dictations prepared by the physicians. The company has developed a number of tools to extract information from these dictations; using these tools and data from the EMR a total of 1253 potential stage IV breast cancer patients were selected for our study. This exploratory research focuses on stage IV patients where chemotherapy is the sole method of treatment. Additionally, at the practice supplying data, stage IV patients received a standard line of chemotherapy (i.e., a standard dose on a standard schedule).

Patient selection was done in a two-step process described in Table 20. This assumes the patient has only one type of cancer at a time.

Table 20: Process for Selecting Patients for Study

Line	Description of Process
1	**Step 1**
2	Select patients with any history of breast cancer by checking for an
3	ICD-9 code of 174.X in the EMR
4	**Step 2**
5	Of those patients selected in Step 1, select those meeting any of the
6	following criteria:
7	1. Maximum stage entered by the doctor in the EMR was "IV"
8	2. An ICD-9 code indicating cancer has metastasized to another
9	location is present
10	3. Information in the doctor's dictations indicated the cancer
11	had metastasized to another location

Patients that received any chemotherapy were selected from the group of 1253 potential stage IV patients; this resulted in 471 patients remaining. From this group, we needed to identify the true first-line of chemotherapy treatment (at stage IV).

Since the treatment of stage IV breast cancer does not include adjuvant therapy, any adjuvant therapy had to be identified and removed. This is due to the fact that, as noted in Section 4.2, patients nearly always finish adjuvant therapy. Since there is no field in the EMR to identify if a line a therapy is adjuvant, all commonly used adjuvant lines of therapy were removed.

A list of the commonly used adjuvant therapy lines at the practice that provided the data for this study is given in Table 21. A total of 209 patients remained after removing all adjuvant lines of therapy from the stage IV patients that received any chemotherapy. The first treatment a patient in this group received is considered to be a true first-line of treatment since all possible lines of neoadjuvant and adjuvant treatment were removed.

Table 21: List of Common Adjuvant Therapy Lines

Number	Medications
1	Adriamycin, Cytoxan, Taxol
2	Adriamycin, Cytoxan, Taxotere
3	Adriamycin, Cytoxan
4	Taxotere, Carboplatin, Herceptin

The number of labs reported in varying frequencies for each patient totaled 47. Due to the sparseness of data for certain labs, some were eliminated from consideration. Specifically, labs reported less than 50% of the time were excluded. After this exclusion, 29 labs were considered in the analysis, see Appendix A for a list of these labs.

Before describing the process of model creation, we digress to discuss the format of the data. EMRs are designed to store large amounts of data effectively and this storage method may not be conducive to statistical analysis and modeling. Therefore, it is vital that an efficient method for re-formatting EMR data be available for those working with real EMR data. By working closely with a company that understands oncology, oncology data, and are experts on EMR data extraction (EMOL Health

in our case), we were able to understand the format that is used to store data and reasons for using said format. Specifically, lab results are stored in the following format (`Patient ID`, `Lab Date`, `Lab Value`, `Lab Name`). After re-formatting the data, model construction was performed.

A variation of k-fold cross validation was used to create three consensus models for predicting a patient's responsiveness to therapy based on a subset of commonly recorded lab results. This variation is designed to exploit as much of the data as possible since our dataset is relatively small. Table 22 outlines this procedure.

We now outline the impacts of each step in the process of model construction in one sample run, which was coded in R [63]. After Step 0, there were 96 patients with all 29 lab results present. Six folds were used in Step 1 ($K = 6$ and $|N_0^i| = 16$). After Step 1, L_0 consisted of 4 labs; `AST`, `PLT`, `Temp (F)`, and `Total Protein`. Step 3 resulted in 104 patients with all labs from L_0 present. In Step 3, 8 folds were used with size 13 ($K = 8$ and $|N_1^i| = 13$). The predictors used in the eight multivariate regressions are given in Table 23 as well as the accuracy of each model against the corresponding test set. Accuracy is defined by $\frac{1}{N}\sum_{j=1}^{N}|y^i - \hat{y}^j|$, where y^j is the jth observed outcome and $\hat{y}^j$ the jth predicted outcome. Based on the results given in Table 23, it is evident that changes in `PLT`, `Temp (F)`, and `Total Protein` influence a patient's probability of successful first-line treatment.

Three methods for combining the k models generated by the cross-validation were considered. These consensus models, created in Step 4, were obtained by averaging the regression models found in Step 3 in various ways. Recall the classifier for a logistic regression is obtained by the following formula

$$y = R\left(\frac{1}{1+e^{-z}}\right), \text{ where } z = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n. \tag{4.1}$$

Table 22: Procedure for Model Construction

Line	Description of Action
1	Let N and L be the initial set of patients and predictors (labs)
2	**Step 0**
3	Select the subset of patients, N_0, from N with all predictors
4	in L present
5	**Step 1**
6	Divide N_0 into K subsets (folds) of equal size, $N_0^1, \ldots, N_0^K$
7	**for** $i = 1, \ldots, K$
8	**for** $\ell \in L$
9	Run a univariate regression with ℓ as the predictor and
10	$N_0 - N_0^i$ the training set
11	**if** p-value for ℓ is less than 0.05**:** Add ℓ to L_0
12	**end for**
13	**end for**
14	**Step 2**
15	Select the subset of patients, N_1, from N with all predictors
16	in L_0 present
17	**Step 3**
18	Divide N_1 into K subsets (folds) of equal size, $N_1^1, \ldots, N_1^K$
19	**for** $i = 1, \ldots, K$
20	**for** $\ell \in L$
21	Run a univariate regression with ℓ as the predictor and
22	$N_1 - N_1^i$ the training set
23	**if** p-value for ℓ is less than 0.05**:** Add ℓ to L_1^i
24	Run a multivariate regression with L_1^i as the set of predictors,
25	$N_1 - N_i^i$ the training set, and N_1^i as the test set.
26	Call the resulting model M_i
27	**end for**
28	**end for**
29	**Step 4**
30	Create a consensus model M using $\{M_i\}_{i=1}^K$

In (4.1), $R(-)$ is the rounding function, the predictors are $x_1, \ldots, x_n$, and the model coefficients $\beta_0, \ldots, \beta_n$ are estimated by maximizing the log-likelihood for a given set of observations.

Consensus model 1 (CM1) was created by averaging the coefficients of the logistic regression models found in Step 3. Specifically, letting $(\beta_0^i, \ldots, \beta_n^i)$ be the coefficients

Table 23: Sample Results from Multivariate Regressions

Model	Variables	Coefficient	p-Value	Accuracy
1	*Intercept*	1.898 ± 1.384	0.0067	0.6
	`PLT`	-0.00504 ± 0.00457	0.0289	
2	*Intercept*	2.323 ± 1.459	0.0016	0.5
	`PLT`	-0.00578 ± 0.00484	0.0180	
3	*Intercept*	-7.310 ± 6.185	0.0188	0.3
	`PLT`	-0.00722 ± 0.00512	0.0053	
	`Total Protein`	1.454 ± 0.937	0.0021	
4	*Intercept*	-4.084 ± 5.231	0.1209	0.8
	`PLT`	-0.00595 ± 0.00476	0.0132	
	`Total Protein`	0.874 ± 0.758	0.0224	
5	*Intercept*	66.436 ± 65.841	0.0459	0.3
	`Temp (F)`	-0.670 ± 0.671	0.0482	
6	*Intercept*	-3.942 ± 5.312	0.1401	0.5
	`PLT`	-0.00625 ± 0.00484	0.0107	
	`Total Protein`	0.895 ± 0.775	0.0221	
7	*Intercept*	-4.360 ± 4.966	0.0806	0.5
	`Total Protein`	0.701 ± 0.706	0.0483	
8	*Intercept*	77.049 ± 71.915	0.0340	0.3
	`PLT`	-0.00629 ± 0.00509	0.0143	
	`Temp (F)`	-0.873 ± 0.752	0.0216	
	`Total Protein`	1.525 ± 0.913	0.0009	

from model i in Step 3, define $(\beta_0, \ldots, \beta_n)$ for CM1 by

$$\beta_k = \sum_{i=1}^{8} \frac{\beta_k^i}{8}, \text{ for } k = 0, \ldots, n.$$

Letting $(x_1^j, \ldots, x_n^j)$ be the set of observed predictors for patient j, then the predicted outcome from CM1 for patient j is given by

$$\hat{y}_1^j = R\left(\frac{1}{1+e^{-z^j}}\right), \text{ where } z^j = \beta_0 + \beta_1 x_1^j + \cdots + \beta_n x_n^j.$$

Consensus model 2 (CM2) was generated by averaging the probabilistic outcome of

the logistic regression models then rounding the result. That is,

$$\hat{y}_2^j = R\left(\sum_{i=1}^{8} \frac{1}{8\left(1+e^{-z_i^j}\right)}\right), \text{ where } z_i^j = \beta_0^i + \beta_1^i x_1^j + \cdots + \beta_n^i x_n^j.$$

Finally, consensus model 3 (CM3) was generated by averaging the classifiers from the logistic regression models:

$$\hat{y}_3^j = R\left(\sum_{i=1}^{8} \frac{1}{8} R\left(\frac{1}{1+e^{-z_i^j}}\right)\right), \text{ where } z_i^j = \beta_0^i + \beta_1^i x_1^j + \cdots + \beta_n^i x_n^j.$$

The performance of these models was compared to determine the best method for creating a single classification model from the k models generated during the cross validation procedure.

4.2.2 Prediction Model Results and Conclusions

Steps 3 and 4 in Table 22 were iterated 100 times. At the end of each iteration, the consensus models were used to predict the outcome for the set of 104 patients found in Step 3 of model construction. The accuracy for each was recorded along with the sensitivity (true positive prediction rate) and specificity (true negative prediction rate). The average and half-width of the 95% confidence interval for each performance measure are given in Table 24 for each of the consensus models.

Table 24: Summary of Performance of Consensus Models

Model	Accuracy (%)	Sensitivity (%)	Specificity (%)
CM1	71.96 ± 0.22	91.79 ± 0.51	36.05 ± 0.59
CM2	71.87 ± 0.22	91.60 ± 0.50	36.14 ± 0.53
CM3	71.03 ± 0.15	88.70 ± 0.25	39.05 ± 0.36

Since the accuracy of CM1 was superior to that of CM2 and CM3 ($p = 0.02$ and $p < 0.001$, respectively), the remainder of our discussion uses only this model. Although the model had an average sensitivity above 90%, the specificity performance was low. This indicates conservative model performance in the sense that a patient is

classified as responding to treatment only if there truly is a high chance for success. Thus, we can be confident in treating a patient if the model indicates they will respond well to treatment. On the other hand, if the model indicates the patient will not respond well to treatment, then the patient should be more carefully examined before beginning treatment. These results are promising given the small dataset and restricted number of covariates available for consideration.

In this section, we discussed the process of extracting data from an EMR at a private oncology practice in order to create a model for predicting a stage IV breast cancer patient's response to chemotherapy. In particular, three models were constructed to predict a stage IV breast cancer patient's response to first-line treatment. Based on the results given in Table 23, we can see that as `PLT` or `Temp (F)` decrease, probability of successful first line treatment increases while an increase in `Total Protein` causes an increased probability of success. This means that within the range of observed values, it is desirable for a patient to have high `Total Protein` and low `PLT` and `Temp (F)`.

Although the accuracy of our best model was only 72%, this is in fact a promising result. We were able to extract valuable information from a relatively small dataset, indicating that further study is merited. This will include exploring more characteristics of patients as predictors in our model. For instance, it is indicated in [39] and [78] that various comorbities, diseases or disorders present in addition to the primary disease, impact a patient's response to chemotherapy. This data is often available in the doctor's dictations, but tools must be developed to accurately extract it. Additionally, increasing the population size would allow for more robust modeling techniques to be used, thereby improving model performance. It would also be beneficial to validate our model's performance with tumor size data and using RECIST criteria as the measure of a patient's response to treatment.

4.3 Palliative Chemotherapy as a Markov Decision Process

This section outlines the process of modeling palliative chemotherapy as a Markov decision process (MDP). The findings from Section 4.2 serve a platform for establishing the state space and reward function of the MDP.

4.3.1 Optimization Models in Chemotherapy

Optimization of the within cycle administration schedule has received a good deal of attention in the literature, however there is currently no research addressing long term planning for chemotherapy. This work is worth explaining for two reasons. First, it provides insights into modeling diseases, although the methodology used here is not directly applicable. Second, since such a body work exists for the within cycle treatment schedule, it is justified to ignore this portion of treatment while attempting to determine the best long term treatment schedule.

The basic building block for most of this research is pharmacokinetic models coupled with cell growth models. Pharmacokinetics is the study of determining an external substance's progression in a living organism. As applied to cancer treatment, this means tracking the progression of chemotherapy drugs throughout the cells of the body and any tumors that are present.

Bellman developed mathematical models for simple pharmacokinetics by viewing the body as being composed of regions called compartments and determining the concentration of a substance as a function of time in each compartment [10]. These compartments can be real from a physiological stand point, e.g. blood, ear, or bones, or they can be mathematical constructs that make model formulation more convenient. The movement of a substance between regions is modeled using differential equations (difference equations are used if time is treated discretely).

Consider the one compartment example by Bellman from [10]. Let $x(t)$ be the concentration of a substance at time t in the compartment, $f(t)$ be the rate of infusion of the substance into the compartment, and assume the rate at which the substance

disappears from the compartment is proportional to the amount of the substance present in the compartment. If the compartment has a volume v and the fraction of reduction in the concentration level for a small amount of time, dt, is given by $k\ dt$, where k is a constant, then the system can be described by $v\ x(t+dt) = (1-k\ dt)v\ x(t) + f(t)dt$. Assuming $x(t)$ is "well-behaved," for example it can be expanded as a Taylor series, then this can be re-written as the differential equation $v\ x'(t) = -k\ v\ x(t) + f(t)$. Given an initial concentration, a closed form solution can easily be found;

$$x(t) = x(0)e^{-kt} + \int_0^t \frac{e^{-k(t-s)}f(s)}{v} ds.$$

Martin and Teo combined Bellman's work of modeling substance concentration in compartments with cell growth models [44]. Cell growth models are used to estimate the growth rate of a population of cells. For tumor growth, the objective is to determine $P(t)$, the cell population of a tumor at time t, assuming the tumor has an initial cell population of P_0, where P satisfies $P'(t) = F(P(t))$ and $P(0) = P_0$. These equations say that the growth rate of a tumor is equal to some function of the current cell population. The function $F(-)$ is referred to as the growth function. By estimating a tumor's reaction to the concentration of chemotherapy drugs, the cell population of the tumor is then modeled subject to cell death from chemotherapy and natural cell replication. This work has been extended over the years to include additional compartments [56], optimizing drug doses given a treatment schedule [31], and cell-cycle specific treatments [2].

Although pharmacokinetic models coupled with cell growth models have a number of strengths, such as providing very specific information about cellular response to treatment since modeling is done on this level and including constraints regarding toxicity levels by adding compartments corresponding to healthy tissues in the body, these are deterministic models. This means that once the parameters of the model are chosen, a given treatment scheme (schedule/dosage) always provides the same

result. Thus, there is an assumption that every patient will have the same reaction to treatment, which is certainly not the case. In spite of this restriction, this modeling does provide a solid framework for the within cycle treatment scheduling. Therefore, it is justified to turn attention to optimizing long-term treatment planning.

4.3.2 Markov Decision Processes in Medical Modeling

Before describing the process of modeling palliative chemotherapy as a MDP, a brief overview of MDPs is provided along with previous medical applications. A MDP consists of a collection of states that describe a system, actions that can be taken by a decision maker along with rewards associated with those actions, the probability the system transitions to a particular state subject to the current state and an action, and a set of decision epochs [62]. They are applicable to cases where the transition probabilities and reward function are history independent, meaning only information regarding the current state is needed to determine the transition probabilities and rewards. Many medical treatment decisions are sequentially made and are subject to uncertainty. Such decision making environments are well suited to be modeled with MDPs [69].

Successful applications of MDPs in medical decision making include drug infusion plans for the administration of anesthesia [30], optimal acceptance for kidney transplants [3], cost-benefit analysis of mammograms and treatment options in breast cancer [35], and optimal acceptance of living-donor and cadaveric liver transplants [4, 5]. Although, MDPs are well suited for modeling medical treatment planning, there are disadvantages to using them. As the size of the problem grows, MDPs become computationally intractable [69]. This means that using a large number of dimensions to describe a patient's state can cause the problem to be extremely difficult to solve, while it may be necessary to use many dimensions in the state space in order to capture the true complexity of the treatment process in question. The

other challenge in applying MDPs to healthcare treatment decision making is the availability and reliability of data.

4.3.3 Markov Decision Process Model for Palliative Chemotherapy

To solve the problem of determining the optimal sequence of treatment decisions for a late stage breast cancer patient receiving palliative chemotherapy, a discrete-time, infinite-horizon, discounted MDP model is formulated. The objective of the model is to maximize the patient's expected predicted response to treatment as defined in Section 4.2 and [15]. There are three assumptions made in this model:

1. The "best" protocol has been selected by the physician and the patient will remain on this protocol for the duration of treatment.

2. Stage IV breast cancer is terminal.

3. The transition probabilities and reward function are stationary.

The first assumption implies that we do not consider altering the medications a patient is receiving, so the only decision options are treat or don't treat. We note that this is a strong assumption and in reality oncologists may adjust the medicines and doses to fit a patient's needs. Incorporating these options provides future research material. Assumption two means that a patient remains in treatment until he/she dies. Since the 5-year survival rate for stage IV breast cancer is 15%, this is not always the case, but it is plausible. Finally, assumption three allows us to use steady state equations when solving the model.

The notation for the model is now presented; a summary is provided in Table 25. Let $N = \{1, \ldots, \infty\}$ be the set of decision epochs in the model, $\mathcal{S}$ be the patient state space, and $\mathcal{A}$ be the action space. For $a \in \mathcal{A}$, $\mathcal{T}_a$ is the transition probability matrix with a single entry given by $\mathcal{T}_a(s, s')$, where $s, s' \in \mathcal{S}$. For $s \in \mathcal{S}$ and $a \in \mathcal{A}$, let $V(s)$ be the value of being in state s and $r(s, a)$ the reward for taking action a while in

state s. Since this process is modeled as an infinite horizon problem with stationary transition probabilities and reward function, the value function for a given state can be written as

$$V(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \beta \sum_{s' \in \mathcal{S}} \mathcal{T}_a(s,s') V(s') \right\} \tag{4.2}$$

where $\beta \in (0,1)$ is the discount factor [62]. Letting $\mathbf{V}^* = (V^*(s))_{s \in \mathcal{S}}$ be the collection of solutions to (4.2), our objective is to determine the optimal policy, $\mathbf{a}^* = (a^*(s))_{s \in \mathcal{S}}$ for this problem.

Table 25: Notation for Palliative Chemotherapy MDP Model

N	time periods, $\{1, \ldots, \infty\}$
$\mathcal{S}$	patient state space
s_t	patient state at time t
$\mathcal{A}$	action space, {treat, don't treat}
a_t	action at time t
$r(s,a)$	reward for action a when in state s
$V(s)$	value of being in state s
β	discount factor
$\mathcal{T}_a$	transition probability matrix for action a

4.3.4 Model Data

In the MDP model for palliative chemotherapy, the state space and reward function were selected based on the analysis conducted in Section 4.2 and [15]. Of the nearly 50 labs reported (at various frequencies), it was found that only three were required to predict a patient's response to first-line chemotherapy: platelet count (`PLT`), total protein (`Total Protein`), and temperature (`Temp (F)`). Further analysis showed that after discretizing these labs a four levels, using only `PLT` and `Total Protein` produced similar accuracies (71.8% average accuracy in Section 4.2 versus 68% for two discrete variables). Based on this, the state space selected is

$$\mathcal{S} = \{\texttt{PLT} \text{ level}, \texttt{Total Protein} \text{ level}, \texttt{Status}\}$$

where the possible levels for both PLT and Total Protein are 0, 1, 2, 3, and NA, and Status can be either Alive or Dead. Not all combinations of PLT, Total Protein, and Status are possible. In particular, if Status = Dead then PLT and Total Protein are both NA. Also, if Status = Alive then at most one of PLT and Total Protein can equal NA. With these restrictions, a patient has a total of 25 possible states. Based on assumption 1 of the model, the set of possible actions consists of {Treat, Don't Treat}. That is, we do not consider changing protocols as a possible action.

In this model, the reward function is generated in following manner. Let $f : \mathcal{S} \to (0, 1)$ be defined by

$$f\left(\texttt{PLT}, \texttt{Total Protein}\right) = \left(1 + e^{-\beta_0 - \beta_1 \texttt{PLT} - \beta_2 \texttt{Total Protein}}\right)^{-1} \tag{4.3}$$

where β_i are determined by performing a logistic regression as in [15], then define $r(s, a)$ as

$$r(s,a) = \begin{cases} f(s) & \text{if } a = \texttt{Treat} \\ \gamma(1 - f(s)) & \text{if } a = \texttt{Don't Treat} \end{cases} \tag{4.4}$$

with $\gamma \in [0, 1]$. This reward function is designed to keep patients in a state that has a high probability of successful treatment. It should be noted that the value chosen for γ can significantly affect the optimal policy. For instance, if $\gamma = 0$ then the optimal strategy would be to always treat the patient. Hence, determination of this parameter is extremely important. Note we will often write $a = 1$ to mean Treat and $a = 0$ to mean Don't Treat.

Finally, the transition probabilities were estimated using data from a large oncology practice in the Midwest. Specifically, the levels of PLT and Total Protein along with the action taken at the time of these observations were analyzed to estimate the transition probabilities. There were 5065 observations used in this process. Presum-

ably, the major drivers of the transitions between states are the treatment action and random processes in the patient.

4.3.5 Solution Method and Optimal Policy

The problem was solved using policy iteration [58, 62]. For the sake of completeness, Table 26 provides the details of policy iteration.

Table 26: Policy Iteration Solution Algorithm for MDP Models

Line	Process Description
1	Select a policy π_0, set $n = 1$ and value = `True`
2	**while** value
3	Compute $\mathcal{T}_{a_{\pi_{n-1}}}$ and $r(s, a_{\pi_{n-1}})$ for all s
4	Let V_n solve $\left(I - \beta\mathcal{T}_{a_{\pi_{n-1}}}\right) V = r(S, a_{\pi_{n-1}})$
5	Let π_n be the policy defined by
6	$a_n(s) \in \text{argmax}_{a\in\mathcal{S}} \left\{r(s,a) + \beta \sum_{s'\in\mathcal{S}} V(s')\mathcal{T}_a(s,s')\right\}$
7	**if** $a_n(s) = a_{n-1}(s)$ for all s **:** value = `False`
8	**end while**

The general idea of policy iteration is to change actions based only on the current state being considered. This process continues until the policy is consistent for all states. This algorithm is known to converge to the optimal solution in finitely many iterations [62].

The optimal policy and values are provided in Table 27. Recall that Action = 1 to means `Treat` and Action = 0 to means `Don't Treat`.

The results in Table 27 confirm the suspicion that there are patient states in which treatment should be withheld. Although there is no claim to the physiological drivers of this, one possible explanation is that withholding treatment will allow patients to recover from the toxic effects of chemotherapy permitting them to transition to a state with a higher probability of successful treatment.

4.3.6 Conclusions and Treatment Insights

To assess the impact of different policies on treatment, a numeric experiment was performed. Specifically, a pure treatment strategy was compared to the optimal

Table 27: Optimal Policy for Palliative Chemotherapy MDP Model

State	Tuple	Action	Value
0	$\{0, 0, \texttt{Alive}\}$	1	2.76
1	$\{0, 1, \texttt{Alive}\}$	1	2.87
2	$\{0, 2, \texttt{Alive}\}$	1	2.93
3	$\{0, 3, \texttt{Alive}\}$	1	2.95
4	$\{1, 0, \texttt{Alive}\}$	1	2.65
5	$\{1, 1, \texttt{Alive}\}$	1	2.76
6	$\{1, 2, \texttt{Alive}\}$	1	2.88
7	$\{1, 3, \texttt{Alive}\}$	1	2.99
8	$\{2, 0, \texttt{Alive}\}$	0	2.23
9	$\{2, 1, \texttt{Alive}\}$	1	2.52
10	$\{2, 2, \texttt{Alive}\}$	1	2.65
11	$\{2, 3, \texttt{Alive}\}$	1	2.91
12	$\{3, 0, \texttt{Alive}\}$	0	2.11
13	$\{3, 1, \texttt{Alive}\}$	0	2.29
14	$\{3, 2, \texttt{Alive}\}$	0	2.47
15	$\{3, 3, \texttt{Alive}\}$	1	2.54
16	$\{NA, 0, \texttt{Alive}\}$	0	2.43
17	$\{NA, 1, \texttt{Alive}\}$	1	2.64
18	$\{NA, 2, \texttt{Alive}\}$	1	2.81
19	$\{NA, 3, \texttt{Alive}\}$	1	2.41
20	$\{0, NA, \texttt{Alive}\}$	1	2.93
21	$\{1, NA, \texttt{Alive}\}$	1	2.68
22	$\{2, NA, \texttt{Alive}\}$	1	2.52
23	$\{3, NA, \texttt{Alive}\}$	0	2.38
24	$\{-, -, \texttt{Dead}\}$	0	0

policy shown in Tables 27. To do this, patient progression through treatment was simulated first using the optimal policy then using a pure treatment policy and the duration of treatment and number of treatment cycles were recorded. This was done assuming the patient started treatment in each of the 24 living states, then repeated 100 times. Table 28 provides the impact of the two treatment policies on the duration of treatment and the number of cycles.

The column headings of Table 28 are State which is the initial state of the simulated patient, Opt. Pol. represents optimal policy and Dur./Trts. represents duration and number of treatment cycles, and P.T. Pol. represents pure treatment policy and

Table 28: Impact of Optimal Policy on Treatment

State	Opt. Pol. Dur./Trts.	P.T. Pol. Dur.	State	Opt. Pol. Dur./Trts.	P.T. Pol. Dur.
0	23.61 18.21	21.25	12	26.04 18.5	20.06
1	19.87 16.03	18.39	13	23.11 15.68	18.64
2	22.77 17.86	18.63	14	27.12 19.42	22.08
3	19.73 15.36	21.11	15	24.33 18.19	23.24
4	23.46 18.01	20.85	16	25.93 18.59	17.32
5	25.01 19.26	17.76	17	23.16 17.19	17.1
6	25.42 20.06	20.83	18	27.61 20.92	20.29
7	24.33 19.16	21.48	19	17.23 13.76	18.17
8	24.79 18.33	18.63	20	21.16 16.17	22.9
9	27.12 21.1	20.74	21	27.71 21.15	16.32
10	25.71 19.79	16.97	22	26.67 20.87	21.13
11	23.36 18.16	19.82	23	23.95 17.03	20.62

Dur. represents duration of treatment. Note that for the pure treatment policy the duration of treatment equals the number of treatment cycles. In the Opt. Pol. column, the first number is the average duration and the second number is the average number of treatment cycles.

The most significant feature to notice is that the optimal policy provides a longer duration of treatment with fewer treatment cycles than the pure treatment policy on average with the exception of seed states 3, 19, and 20. The implications of this result are actually quite significant. First, adopting the optimal policy results in longer life for the patient based on the assumption that stage IV breast cancer is terminal. Second, since in general less treatment cycles are needed to achieve longer life, the overall cost of care should be less and the patient should experience less suffering. The latter is based on the fact that chemotherapy is quite toxic, as noted in Section 4.2, and therefore has many side effects that cause patient suffering.

4.4 Conclusions and Future Research in Treatment Planning

Cancer is a group of complex diseases with a variety of complex treatment options, nearly all of which are harmful to healthy tissue in addition to cancerous tissue. Moreover, for late-stage cancer patients, i.e., stage IV, treatment is generally palliative. One implication of this is that treatment will be long lasting. By developing treatment plans that incorporate future information into current decisions, the quality of patient care will certainly be improved. In this chapter, a framework was developed for identifying patient attributes that are linked to patient response to treatment using clinical EMR data. The information provided by this methodology when applied to stage IV breast cancer patients receiving chemotherapy served as the foundation for constructing a stochastic optimization model to improve the timing of treatment cycles. This will provide oncologist with a decision support tool while treating terminally ill patients.

Some suggested directions for extending this work include: First, identifying the physiological drivers of the patient attributes associated with response to treatment as defined in Section 4.2 would be of great interest to the medical community. Such an understanding could assist in the development of additional treatment options. Second, this study focused on patients at a single practice and region. Expanding the data used to include more practices and regions would help eliminate any bias introduced by treatment style and regional factors. Third, it is crucial that the models developed be validated against the medical response to cancer, the RECIST guidelines. One possible method for this would be to pair with a research group that has conducted clinical trials, since they would have access to complete data including tumor response information.

5 Concluding Remarks

In this dissertation, an effort has been made to expand the body of knowledge pertaining to stochastic optimization. This began with an explanation of the threads that connect the seemingly disparate areas of two-stage and multistage modeling within the field. With these connections made clear, the task of pushing forward the boundaries of methodology and applications of stochastic optimization was undertaken.

The primary contributions of the work presented in this dissertation are:

1. Developing a novel solution method for CCMIPs. This class of problems has received very little attention in the literature. Moreover, the solution method developed deviates from traditional branch and cut strategies thereby exploiting the specific structure of the deterministic reformulation of CCMIPs.

2. Applying stochastic modeling to the comprehensive hybrid energy system design problem. This also led to an improved implementation of Benders' decomposition which includes multiple cut generation strategies. In particular, the notion of maximum feasible subsystem generated cuts was extended to a separable subproblem.

3. Developing a model to aid in long-term planning for palliative chemotherapy. This involved identifying a methodology for linking clinical patient data to patient response to treatment then translating this information to a multistage stochastic decision model. More specifically, the palliative chemotherapy treatment process was modeled as a Markov decision process.

The information garnered from this model has the potential to provide clinical oncologists with valuable information regarding patient treatment.

In spite of the impact of this research and the steps taken, there is a great deal left to be done. More specifically, the following items will be addressed in future research:

1. Extension of the PERC Heuristic to an exact solution method through the identification of lower bound information based on the current feasible solution. One approach for this is to combine a relaxation technique, such a Lagrangian relaxation, with the PERC Heuristic.

2. Extend the hybrid energy system design model to include multiple building periods. This will possibly require a revamping of the current solution method to accommodate the new problem structure imposed by the multistage nature of this extension.

3. Develop a chance constrained hybrid energy system design model. As mandates are passed requiring minimum usage levels for renewable energy, for example Executive Order S-14-08 [51] in California requires 33% of energy from renewable sources by 2020, generating capacity will need to be altered. Given the stochastic nature of renewable resources, it is reasonable to design a system that can meet such guidelines with high probability.

4. Validation of the model for predicting first line chemotherapy response in stage IV breast cancer patients and the MDP model for palliative chemotherapy with use of tumor response data. Such data could be obtained by partnering with an organization that conducts clinical trials such as Moffitt Cancer Center or the Southwest Oncology Group.

5. Expand the action space in the MDP for palliative chemotherapy to include switching lines of therapy. This will require a large dataset in order to

accurately estimate the transition probabilities between patient states. As above, organizations with larger more complete datasets would be excellent partners for this future research.

6. Attempt to identify and explain the physiological drivers of the factors impacting response to treatment in stage IV breast cancer patients. This will involve further collaboration with oncologists and other subject matter experts.

These items will address a number of gaps in the literature not filled by the work in this dissertation. Moreover, they all have results that will translate to solutions for real problems decision makers are facing, which will ultimately improve the quality of life for everyone. In the end, this is the true objective of the intellectual exercise known as research. Although many will claim "knowledge for knowledge's sake," this sentiment often overlooks the impact of more knowledge and its benefits to society.

References

[1] C. Abbey and G. Joos, *A stochastic optimization approach to rating of energy storage systems in wind-diesel isolated grids*, IEEE Transactions on Power Systems **24** (2009), no. 1, 418–426.

[2] Z. Agur, R. Hassin, and S. Levy, *Optimizing chemotherapy scheduling using local search heuristics*, Operations Research **54** (2006), no. 5, 829–846.

[3] J.H. Ahn and J.C. Hornberger, *Involving patients in the cadaveric kidney transplant allocation process: A decision-theoretic perspective*, Management Science **42** (1996), 629–641.

[4] O. Alagoz, L. Maillart, A.J. Schaefer, and M. Roberts, *Determining the acceptance of cadaveric livers using an implicit model of the waiting list*, Operations Research **55** (2007), no. 1, 24–36.

[5] O. Alagoz, A.J. Schaefer, L.M. Maillart, and M.S. Roberts, *The optimal timing of living-donor liver transplantation*, Management Science **50** (2004), no. 10, 1420–1430.

[6] H. An and J.W. Eheart, *A screening technique for joint chance-constrained programming for air quality management*, Operations Research **55** (2007), 792–798.

[7] S.C. Bagley, H. White, and B.A. Golomb, *Logistic regression in the medical literature: Standards for use and reporting, with particular attention to one medical domain*, Journal of Clinical Epidemiology **54** (2001), no. 10, 979–985.

[8] J.P. Barton and D.G. Infield, *Energy storage and its use with intermittent renewable energy*, IEEE Transactions on Energy Conversion **19** (2004), no. 2, 441–448.

[9] R. Bellman, *Dynamic programming*, Princeton University Press, 1957.

[10] ———, *Mathematical models in medicine*, World Scientific Publishing Company Inc., 1983.

[11] J.F. Benders, *Partitioning procedures for solving mixed-variables programming problems*, Computational Management Science **2** (2005), 3–19.

[12] J.L. Bernal-Agustín and R. Dufo-López, *Simulation and optimization of stand-alone hybrid renewable energy systems*, Renewable and Sustainable Energy Reviews **13** (2009), no. 8, 2111–2118.

[13] J.R. Birge and F. Louveaux, *Introduction to stochastic programming*, Springer Verlag, 2004.

[14] P.D. Brown, J.A.P. Lopes, and M.A. Matos, *Optimization of pumped storage capacity in an isolated power system with large renewable penetration*, IEEE Transactions on Power Systems **23** (2008), no. 2, 523–531.

[15] G. Centeno, L. Kuznia, B. Zeng, B. Decker, V. Decker, and D. Decker, *Data mining techniques for predicting response to palliative chemotherapy*, Working Paper.

[16] National Climatic Data Center, www.ncdc.noaa.gov.

[17] A. Charnes, W.W. Cooper, and G.H. Symonds, *Cost horizons and certainty equivalents; an approach to stochastic programming of heating oil*, Management Science **4** (1958), 235263.

[18] J. Chhatwal, O. Alagoz, M.J. Lindstrom, C.E. Kahn, K.A. Shaffer, and E.S. Burnside, *A logistic regression model based on the national mammography database format to aid breast cancer diagnosis*, American Journal of Roentgenology **192** (2009), no. 4, 1117–1127.

[19] G. Côté and M. Laughton, *Large-scale mixed integer programming: Benders-type heuristics*, European Journal of Operations Research **16** (1984), 327–333.

[20] D. Decker, Personal Interview, May 2010.

[21] N. Duić and M.d.G. Carvalho, *Increasing renewable energy sources in island energy supply: case study Porto Santo*, Renewable and Sustainable Energy Reviews **8** (2004), no. 4, 383–399.

[22] E.A. Eisenhauer, P. Therasse, J. Bogaerts, L.H. Schwartz, D. Sargent, R. Ford, J. Dancey, S. Arbuck, S. Gwyther, M. Mooney, L. Rubinstein, L. Shankar, L. Dodd, R. Kaplan, D. Lacombe, and J. Verweij, *New response evaluation criteria in solid tumours: Revised RECIST guideline (version 1.1)*, European Journal of Cancer **45** (2009), 228–247.

[23] O. Ekren and B.Y. Ekren, *Size optimization of a PV/wind hybrid energy conversion system with battery storage using simulated annealing*, Applied Energy **87** (2010), no. 2, 592–598.

[24] Wind Power for Every Home, www.mywindpowersystem.com/.

[25] Office for National Statistics, www.statistics.gov.uk/CCI/nugget.asp?ID=6.

[26] GovTrack, www.govtrack.us/congress/bill.xpd?bill=h111-2454.

[27] EMOL Health, www.emolhealth.com/.

[28] R. Henrion and A. Möller, *Optimization of a continuous distillation process under random inflow rate*, Computers and Mathematics with Applications **45** (2003), 247–262.

[29] D.W. Homer and S. Lemeshow, *Applied logistic regression*, John Wiley and Sons, Inc., 2000.

[30] C. Hu, W.S. Lovejoy, and S.L. Shafer, *Comparison of some suboptimal control policies in medical drug therapy*, Operations Research **44** (1993), 696–709.

[31] A. Iliadis and D. Barbolosi, *Optimizing drug regimens in cancer chemotherapy by an efficacy toxicity mathematical model*, Computers and Biomedical Research **33** (2000), 211–226.

[32] National Cancer Institute, www.cancer.gov/cancertopics/pdq/treatment/breast/Patient/.

[33] ———, www.cancer.gov.

[34] ———, *Surveillance epidemiology and end results*, seer.cancer.gov/statfacts/.

[35] J.S. Ivy, *A maintenance model for breast cancer detection and treatment*, Submitted for Publication, 2002.

[36] B. Karimi, S.M.T. Fatemi Ghomi, and J.M. Wilson, *The capacitated lot sizing problem: A review of models and algorithms*, Omega - The International Journal of Management Science **31** (2003), 365–378.

[37] Y. Katsigiannis and P. Georgilakis, *Optimal sizing of small isolated hybrid power systems using tabu search*, Journal of Optoelectronics and Advanced Materials **10** (2008), no. 5, 1241–1245.

[38] L. Kuznia, B. Zeng, G. Centeno, and Z. Miao, *Stochastic optimization for power system configuration with renewable energy in remote areas*, to appear in Annals of Operations Research.

[39] L. Lee, W.Y. Cheung, E. Atkinson, and M. Krzyzanowska, *Impact of comorbidity on chemotherapy use and outcomes in solid tumors: A systematic review*, Journal of Clinical Oncology **29** (2011), no. 1, 106–117.

[40] M.A. Lejeune and A. Ruszczyński, *An efficient trajectory method for probabilistic inventory-production distribution problems*, Operations Research **55** (2007), 378–394.

[41] J. Luedtke, *An integer programming and decomposition approach to general chance-constrained mathematical programs*, Lecture Notes in Computer Science **6080** (2010), 271–284.

[42] J. Luedtke, S. Ahmed, and G. Nemhauser, *An integer programming approach for linear programs with probabilistic constraints*, Mathematical Programming **122** (2010), no. 2, 247–272.

[43] T.L. Magnanti and R.T. Wong, *Accelerating Benders' decomposition: Algorithmic enhancement and model selection criteria*, Operations Research **29** (1981), no. 3, 464–484.

[44] R. Martin and K.L. Teo, *Optimal control of drug administration in cancer chemotherapy*, World Scientific Publishing Company Inc., 1994.

[45] B. Miller and H. Wagner, *Chance constrained programming with joint constraints*, Operations Research **13** (1965), no. 6, 930–945.

[46] M.R. Murr and A. Prékopa, *Solution of a product substitution problem using stochastic programming*, Probabilistic Constrained Optimization: Methodology and Applications (S.P. Uryasev, ed.), Kluwer Academic Publishers, 2000, pp. 252–271.

[47] P. Nema, R.K. Nema, and S. Rangnekar, *A current and future state of art development of hybrid energy system using wind and PV-solar: A review*, Renewable and Sustainable Energy Reviews **13** (2009), no. 8, 2096–2103.

[48] G. Nemhauser and L. Wolsey, *Integer and combinatorial optimization*, Wiley-interscience, 1988.

[49] E.M. Nfah, J.M. Ngundam, and R. Tchinda, *Modelling of solar/diesel/battery hybrid power systems for far-north Cameroon*, Renewable Energy **32** (2007), no. 5, 832–844.

[50] National Grid: The Power of Action, www.nationalgrid.com.

[51] State of California Office of the Governer, http://gov38.ca.gov/index.php?/executive-order/11072/.

[52] Symbiotics A New Generation of Hydropower, www.symbioticsenergy.com/.

[53] Gurobi Optimization, www.gurobi.com.

[54] B.K. Pagnoncelli, S. Ahmed, and A. Shapiro, *Sample average approximation method for chance constrained programming: Theory and applications*, Journal of Optimization Theory and Applications **142** (2009), 399–416.

[55] N. Papadakos, *Practical enhancements to Magnanti-Wong method*, Operations Research Letters **36** (2008), 444–449.

[56] F.L. Pereira, C.E. Pedreira, and J.B. de Sousa, *A new optimization based approach to experimental combination chemotherapy*, Frontiers of Medical and Biological Engineering **6** (1995), no. 4, 257–268.

[57] S. Pereira, R. Segurado, A. Costa, A. Pipio, and L. Alves, *Energy storage and its use with intermittent renewable energy*, Chemical Engineering Transactions **18** (2009), 629–634.

[58] W. Powell, *Approximate dynamic programming: Solving the curses of dimensionality*, Wiley-Interscience, 2007.

[59] A. Prékopa, *On probabilistic constrained programming*, Proceedings of the Princeton Symposium on Mathematical Programming (Princeton, NJ), Princeton University Press, 1970, pp. 113–138.

[60] ———, *Contributions to the theory of stochastic programming*, Mathematical Programming **4** (1973), 202–221.

[61] ———, *Probabilistic programming*, Stochastic Programming (A. Ruszczynski and A. Shapiro, eds.), Handbooks in Operations Research and Management Science, vol. 10, Elsevier, 2003, pp. 267 – 351.

[62] M. Puterman, *Markov decision processes: Discrete stochastic dynamic programming*, Wiley-Interscience, 1994.

[63] R Development Core Team, *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria, 2011, ISBN 3-900051-07-0.

[64] S. Ross, *Introduction to probability models*, Academic Press, 2009.

[65] A. Ruszczyński, *Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra*, Mathematical Programming Series A **93** (2002), no. 2, 195–215.

[66] G. Saharidis, M. Boile, and S. Theofanis, *Initialization of the benders master problem using valid inequalities applied to fixed-charge network problems*, Expert Systems with Applications **38** (2011), 6627–6636.

[67] G. Saharidis and M. Ierapetritou, *Improving Benders' decomposition using maximum feasible subsystem (MFS) cut generation strategy*, Computers and Chemical Engineering **34** (2010), 1237–1245.

[68] G. Saharidis, M. Minoux, and M. Ierapetritou, *Accelerating benders method using covering cut bundle generation*, International Transactions in Operational Research **17** (2010), 221–237.

[69] A.J. Schaefer, M. Bailey, S. Shechter, and M. Roberts, *Modeling medical treatement using Markov decision processes*, Handbook of Operations Research/Management Science Applications in Health Care, Kluwer Academic Publishers, 2004.

[70] M. Sellmann, G. Kliewer, and A. Koberstein, *Lagrangian cardinality cuts and variable fixing for capacitated network design*, Tech. report.

[71] T. Senjyu, D. Hayashi, A. Yona, N. Urasaki, and T. Funabashi, *Optimal configuration of power generating systems in isolated island with renewable energy*, Renewable Energy **32** (2007), 1917–1933.

[72] A. Shapiro, D. Dentcheva, and A.P. Ruszczyński, *Lectures on stochastic programming: modeling and theory*, MPS-SIAM series on optimization, Society for Industrial and Applied Mathematics, 2009.

[73] American Cancer Society, www.cancer.org.

[74] ———, www.cancer.org/cancer/breastcancer/detailedguide/breast-cancer-survival-by-stage/.

[75] General Algebraic Modeling System, www.gams.com.

[76] A.K. Takyi and B.J. Lence, *Surface water quality management using a multiple-realization chance constraint method*, Water Resources Research **35** (1999), 1657–1670.

[77] M. Tanner and L. Ntaimo, *IIS branch-and-cut for joint chance-constrained programs with random technology matrices*, 2008.

[78] S. Zhang, J. Simmons Ivy, F. Cobb Payton, and K. M. Diehl, *Modeling the impact of comorbidity on breast cancer patient outcomes*, Health Care Management Science **13** (2010), no. 2, 137–154.

Appendices

Appendix A Collection of Labs

Table 29: Complete Set of Labs Considered in Predicting Response to Chemotherapy

Name	Description
ALP	Alkaline Phosphatase
ALT	Alanine Aminotransferase
AMGFR	A Multiple of Glomerular Filtration Rate
AST	Aspartate Aminotransferase
BSA	Body Surface Area
BUN	Blood Urea Nitrogen
Calcium	Calcium
Chloride	Chloride
CO2	Bicarbonate
Creatinine	Creatinine
Diastolic	Diastolic Blood Pressure
Glucose	Blood Glucose Level
HCT	Hematocrit
Height (in)	Height in Inches
HGB	Hemoglobin
MCH	Mean Corpuscular Hemoglobin
MCHC	Mean Corpuscular Hemoglobin Concentration
MCV	Mean Corpuscular Volume
PLT	Platelet
Potassium	Blood Serum Potassium
RBC	Red Blood Cell Count
Sodium	Blood Serum Sodium
Systolic	Systolic Blood Pressure
TBILI	Total Bilirubin
Temp (F)	Temperature in Degrees Fahrenheit
Total Protein	Total Blood Serum Protein
WBC	White Blood Cell Count
Weight (lb)	Weight in Pounds
AGE	Age in Years

About the Author

Ludwig Kuznia received his Ph.D. in Industrial Engineering in 2012 and a M.A. in Mathematics in 2009 from the University of South Florida. He earned a B.S. in Mathematics from Western Michigan University in 2006. His research interests are in the area of stochastic optimization with applications in healthcare treatment planning and capacity expansion models for hybrid energy systems. Additionally, he works on algorithm development for chance constrained mixed integer programs. He will be joining the Revenue Management and Analytics Group at Walt Disney World in May 2012.