Stochastic Capacity and Facility Location Planning with Ambiguous Probabilities

By

Heejung Kim

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering-Industrial Engineering and Operations Research

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Philip M Kaminsky, Chair
Professor Anil Aswani,
Professor David Aldous

Spring 2020

Abstract

Stochastic Capacity and Facility Location Planning with Ambiguous Probabilities

by

Heejung Kim

Doctor of Philosophy in Engineering - Industrial Engineering and Operations Research

University of California, Berkeley

Professor Philip M. Kaminsky, Chair

Motivated by the pharmaceutical industry, we consider two strategic models - a capacity planning model and a facility location planning model - where demand in each period of the problem horizon will be determined by the outcome of series of random events with two possible outcomes. These strategic planning problems are complicated by the fact that the probability of each of the alternative outcomes of these events is uncertain. Given this setting, we develop approaches for solving these models that are robust to ambiguities in outcome probabilities.

In the first model, we consider a capacity planning model under this setting. We develop approaches to solving capacity expansion models in this setting that are robust to ambiguities in probability of success, and we consider a variety of different objective functions, including minimizing expected cost, minimizing value at risk and minimizing conditional value at risk. We formulate these models as multistage (stochastic) robust integer programs. For cases where these integer programs are challenging to solve to optimality, we propose two heuristic approaches, a straightforward rolling horizon approach and the more sophisticated event spike approach. The idea of event spike is adapted from Beltran-Royo et al. (2014), and in our version, we relax the stage-wise independence restriction which is the assumption in the original approach. These approaches are further developed to be applicable in the robust setting with different objectives. The effectiveness of these heuristics is shown through computational experiments. We also explore how alternative objectives and parameters affect solutions, and when explicitly modeling ambiguous probabilities adds value.

In the second model, we extend the capacity planning model by considering the additional decisions of facility location and demand allocation. In addition to ambiguous probabilities of binary event outcomes, we assume that the uncertainty of demand has not completely resolved after the binary event occurs. We formulate models that are robust to ambiguities in both probabilities and demands by using (stochastic) robust integer programs. We propose variants of Nested Decomposition and Stochastic Dual Dynamic Programming to solve these models more efficiently. We extend these approaches by incorporating ambiguities in probabilities and demand. We also propose approaches that involve decomposing our large problem into a smaller number of sub-problems with longer horizons, and then applying Nested Decomposition and Stochastic Dual Dynamic Programming in this revised setting. In the computational study, we explore the effectiveness of the robust approach in this setting, and demonstrate the performance of the proposed heuristic approaches.

Contents

List of Figures

List of Tables

Acknowledgements

I first would like to express my deep gratitude to my advisor, Professor Philip Kaminsky, who expertly guided me through my Ph.D study. Without his consistent support and patience this dissertation would not have been possible.

I would like to thank my dissertation committee members, Professor Anil Aswani and Professor David Aldous for their suggestion and advice. I also would like to thank Professor Lee Schruben for his advice.

I am extremely grateful to my parents for their love, caring and support. I also want to thank my parents in law for their support. I am also grateful to my sister, my brothers in law, and my sister in law for cheering me up with warm words.

I would like to thank my son, Minjun, for being a joy of my life since he has arrived. Finally, my special thanks goes to my husband Kanghoon, for always being there to support and encourage me.

Chapter 1

Introduction

Capacity expansion and facility location decisions play an important strategic role in many capital intensive industries, and are a key concern of chemical, communication, electrical power, semiconductor, and pharmaceutical firms, among others (Luss, 1982; Wu et al., 2005).

Capacity expansion planning involves determining the timing, size, type, and locations of investments in new facilities, or in expanding the capacity of existing facilities. Facility location planning deals with the decisions of geographical locations of facilities and allocation of customers to facilities such that the demand for some service or product is satisfied.

Significant investments are typically made based on these decisions, and are often irreversible in the short run. Therefore, a mismatch between strategic decisions and demand can significantly impact the profitability of the firm.

The problems are challenging because there is often a high level of uncertainty around future demand, costs, and other problem parameters. Effective planning must both model this uncertainty to the extent possible, and focus on objectives that in some way take this uncertainty into account in a way that is useful for managers.

In particular, we consider a class of discrete-time planning models where demand in each period is determined by the outcome of a series of "binary events," or events that can have two possible outcomes. Depending on the specific problem context, these events represent clinical trials, experimental outcomes, approvals, competitive product introductions, for example. We are specifically motivated by our experiences working with several firms in the biopharmaceutical industry. In this industry, demand for products is quite often well-understood given a series of events, such as outcomes of clinical trials for new products or new indications for existing products, competitive actions, the outcomes of licensing reviews, etc. However, the outcome of these events is often uncertain, and in many cases, the probability of outcomes cannot be effectively estimated. At the same time, building and licensing traditional commercial-scale production capacity can take 4-5 years and can cost up to 1 billion US dollars.

A classic example of the interaction of clinical trial uncertainty and capacity planning involves Genentech, a biopharmaceutical company, which in 2004 was successfully producing Avastin, a new cancer drug (Snow et al., 2006). The firm was expecting steep demand growth, and considered investing in multiple facilities, in South San Francisco (California), Vacaville (California) and Porrino (Spain). However, at the time, this new drug was in clinical trials for use on other forms of cancer, and these trials take at least five years. Ultimate demand was highly dependent on trial outcomes, and strategic decisions at the company were being supported by models with hundreds of scenarios. Generally, in this industry, although the demand is highly dependent on trial outcome, it is quite well understood given the trial results. However, the underlying probability of success of trial is not well understood, and ultimately, the firm invested in capacity that went unused.

In some situations, demand uncertainty may not be completely resolved after the binary event outcome. For example, in the biopharmaceutical industry, although demand is quite often well-understood given a series of binary events, this is not always true. There may still be a good deal of uncertainty once trial results are realized. In particular, if demand is defined for *each customer* as well as time period, its exact size is even more difficult to estimate accurately, which is the case for the facility location planning. In addition, it is possible that the demand distribution after the binary event outcome is not known or not valid to model.

In this thesis, we present mathematical models for capacity expansion planning and facility location planning problems, where demand is dependent on the outcome of a series of Bernoulli trials with ambiguous probabilities of successes (which we often call "trials" in what follows). We apply robust modeling techniques in this context to find solutions that are robust to these ambiguities in parameters (probability of success in Chapter 3; probability of success and demand in Chapter 4), and thus the models are formulated as (stochastic) robust integer programs. For capacity expansion planning models, we formulate robust models with a variety of different objectives in this context. These models enable us to explore questions when robust models are effective, and how alternative objectives affect solutions.

These integer programs are in some cases challenging to solve to optimality, and thus we also develop and test fast heuristic approaches for solving models. Building on approaches in the literature, we develop new approaches with significant modifications, because it is non-trivial to adapt these techniques directly to our models in this setting.

For capacity planning models, we propose two heuristic approaches, a straightforward rolling horizon approach and the more sophisticated event spike approach. The idea of the event spike approach is adapted from Beltran-Royo et al. (2014), who initially developed it for stage-wise independent multistage stochastic linear programmings. We develop a version of the event spike approach that relax the stage-wise independence restriction. These approaches are further developed to be applicable in the robust setting with different objectives.

For facility location planning models, we propose two solution approaches. The event spike approach developed in Chapter 4 mainly focuses on reducing the number of scenarios, but the problem size of facility location planning models depends on the number of scenarios, the number of facility sites and the number of demand locations. Motivated by Yu et al. (2018), who introduce the use of the Stochastic Dual Dynamic Programming (SDDP) algorithm for a multi-period stochastic facility location problem, we propose heuristic approaches that extend the concepts of Nested Decomposition (ND) and SDDP to our setting. In both standard ND and SDDP algorithms, multistage stochastic programming problems are formulated as the equivalent recursive dynamic programs (DP). In the ND, the cost-to-go functions of DP is approximated by adding Benders' cuts, and it converges in finite steps to an optimal solution (Birge, 1985). The SDDP, first developed by Pereira and Pinto (1991), is a sampling based ND, and based on stage-wise independence, and it reduces the number of DP equations significantly. In our setting which includes ambiguities in probability and demand, the standard ND and SDDP cannot be directly applied to our problems. We extend these approaches by incorporating these ambiguities. We also propose approaches where we first decompose our large problem into a smaller number of sub-problems with longer horizons, and then we apply the ND or SDDP to these decomposed problems. This reduces the number of DP equations, and in some cases the number of iterations required for convergence.

The remainder of this thesis is organized as follows. In Chapter 2, we present a review of the literature related to the models in this thesis. In Chapter 3, we formulate capacity expansion models that capture a variety of different objectives, where the probability of each trial success is known, using a multistage stochastic mixed-integer programming. We next extend these models to incorporate ambiguous probabilities. We also develop heuristic algorithms for these models, and present computational results. In Chapter 4, we formulate facility location model as a (stochastic) robust integer program to find a solution that is robust to the ambiguities in probability and demand. We also present solution approaches and computational analysis. Finally, in Chapter 5, we present summary and closing remarks.

Chapter 2

Literature Review

We explore the capacity planning and the facility location planning literature which is most obviously connected to our models in this thesis. For the capacity planning problem, we consider a variety of alternative objectives for our models, and thus we also briefly survey some related optimization models that feature relevant objective functions. Because our motivating problems consider ambiguities in parameters, and our goal is to find solutions that are robust to these parameters, we also survey literature that use robust optimization approaches in settings similar to ours.

2.1 Capacity Planning

Models for capacity planning have been studied intensively since the early 1960's. Luss (1982) provides a comprehensive survey of early work, including modeling approaches, algorithmic solutions, and relevant applications, and a more recent review appears in Julka et al. (2007). Capacity management problems under uncertainty are surveyed by Van Mieghem (2003), who also discusses the incorporation of risk aversion into these models.

Early capacity planning models utilize the tools of stochastic control theory. Manne (1961), in one of the earliest papers that explicitly incorporates stochastic demand, models demand growth as an infinite horizon stochastic process. Related work can be found in Bean et al. (1992), Freidenfelds (1980) and Davis et al. (1987), but overall, that is challenging to model complex constraints using this approach.

Stochastic programming has also been employed for this class of problems (Birge and Louveaux, 2011). This class of models can incorporate complex problem details, but the resulting models may be challenging to solve. Early work in this area focuses on two-stage models, where all fixed charges occur at the first stage where capacity decisions are made, facilitating the use of decomposition methods. Eppen et al. (1989) propose a model for capacity planning in the auto industry using a two-stage stochastic integer programming with recourse, and additional work employing two-stage stochastic models for capacity expansion problems can be found in Fine and Freund (1990), Riis and Andersen (2002), Riis and Lodahl (2002), and Swaminathan (2000).

Multistage models, where capacity expansion decisions that depend on uncertainty resolved up to that time can be made at each time stage, have become more common as computing power has increased and optimization software has advanced. In the context of capacity planning, Rajagopalan (1994) proposes a multistage model for capacity expansion and replacement over time. Demand is assumed to be non-decreasing and the available capacity of a technology, when it appears, is assumed to be sufficient. Z.-L. Chen et al. (2002) develop a multistage stochastic programming model for determining technology choices and capacity plans given a linear investment cost function. They provide a solution procedure based on the augmented Lagrangian method that solves problems with up to $5,000$ scenarios in a reasonable amount of time. Both papers assume either non-decreasing demand or linear investment cost for tractability. Ahmed and Sahinidis (2003) develop a fairly general capacity expansion model that does not require these assumptions. They utilize a multistage stochastic mixed-integer model, and propose a linear programming-based approximation scheme to solve large instances of this model. They prove this scheme converges asymptotically to an optimal solution as the problem size increases. Ahmed et al. (2003) reformulate these problems using variable disaggregation and solve them using a specialized branch-and-bound procedure. Huang and Ahmed (2009) propose an asymptotically optimal approximation scheme to solve capacity expansion problems and apply this scheme to semiconductor tool planning. Singh et al. (2009) apply the variable splitting technique to reformulate a stochastic multistage capacity expansion planning with one or more production facilities, and solve these using Dantzig-Wolfe decomposition.

In contrast to the bulk of traditional stochastic capacity expansion planning literature, which considers randomness (mainly in demand) with known probability distributions, we consider a setting where the probability distribution parameters of uncertain events are not known.

2.2 Alternative Objectives: Minimizing Regret

The most common objective functions for capacity expansion models involve minimizing the expected cost or the worst-case cost. However, solutions that minimize the expected cost may allow too much variability in possible outcomes, and solutions that minimize the worst-case may prove to be too conservative. Regret theory postulates that individuals may explicitly incorporate notions of minimizing potential regret into their decision-making process (Loomes and Sugden, 1982). In finance, for example, Value-at-Risk(VaR) and Conditional Value-at-Risk(CVaR) are popular risk-based objectives. VaR for a specified quantile of potential losses captures the threshold value that losses may exceed with the specified probability. Although VaR is popular, it is not a coherent risk (a measure that satisfies properties of monotonicity, sub-additivity, homogeneity, and translational invariance) as shown in Artzner et al. (1999). CVaR, also known as Expected Shortfall, equals the expected value of losses conditioned on those losses exceeding the threshold. CVaR is a coherent risk measure, making models that use this objective easier to solve, and it is also less sensitive to rare events in the extreme tail of the event probability distribution (Rockafellar and Uryasev, 2002).

In the context of capacity expansion problems, the regret associated with each scenario under a given plan is defined as the difference between the objective value for the scenario using the given plan and the objective value that comes from using the optimal plan for that scenario. Chien and Zheng (2011) propose a model with a min-max regret strategy for capacity planning under demand uncertainty in semiconductor manufacturing. The model considers each possible outcome of the multi-period demand forecast, and the regret of an expansion decision plan is given in terms of capacity oversupply and shortage. Claro and de Sousa (2008) use a mean-risk objective, where CVaR is the risk measure, for a multistage capacity investment problem with economies of scale in capacity costs.

Daskin et al. (1997) propose a model called the α-reliable min-max regret model and apply it to a stochastic facility location model. The model minimizes the maximum regret associated with the scenarios in an endogenously selected subset of given scenarios with associated probabilities. The planner defines a number of scenarios and estimates the probability of each scenario occurring. G. Chen et al. (2006) present an α-reliable mean-excess regret model for the stochastic facility location model. In this model, they minimize the expectation of regret associated with the scenarios in the tail of outcome probabilities with a collective probability of $1 - \alpha$. Compared with the α-reliable min-max regret, this model explicitly controls the magnitude of the regrets in the tail and is computationally much easier to solve. They also show that the α-reliable mean-excess criterion matches the α- reliable min-max criterion closely. Noyan (2012) considers a two-stage stochastic programming model with CVaR as the risk measure. The author presents two decomposition algorithms based on a generic Benders-decomposition to solve these problems and applies the proposed model to a disaster management problem, determining response facility locations and the inventory levels of relief supplies at each facility. Schultz and Tiedemann (2005) also develop a solution algorithm for a two stage stochastic mixed-integer programming involving CVaR. The algorithm is based on the Lagrangian relaxation of non-anticipativity constraints.

In the first capacity planning models, we consider many of these same objectives in our setting, as we discuss in Section 3, and explore how alternative objectives and parameters affect solutions.

2.3 Facility Location

Owen and Daskin (1998) provide a comprehensive review of facility location problems, including dynamic and stochastic versions of the problem. Dynamic characteristics involve difficult timing issues in locating facilities over an extended time horizon. Stochastic characteristics involve uncertainty in problem input parameters such as forecast demand or distance values.

Most of the models for dynamic facility location problems in the literature consider relocation of facilities between time periods in response to changes in demand when there exist relocation costs such as initial investment costs for new facilities, closure costs for existing facilities. Early multi-period models that include dynamic characteristics were studied by Ballou (1968) and Wesolowsky (1973). Many researchers point out that capacity expansion is one of the crucial features to be considered when modeling this class of the problem in many application (Antunes and Peeters, 2001, Canel et al., 2001, and Melo et al., 2006). In Torres Soto (2009), the author formulates the model as a multi-period capacitated fixed charge location problem when the demand is changing in a deterministic manner over time. The objective is to minimize the maximum regret(deviation) between a robust configuration of facilities and the optimal configuration for each time period. Local search and simulated annealing meta-heuristics are implemented to solve this model efficiently. A multi-period facility location problem with multiple commodities and multiple capacity levels in a dynamic setting is also studied recently by Jena et al. (2016). They suggest a hybrid heuristic that first applies Lagrangian relaxation and then constructs a restricted mixed-integer programming model based on the previously obtained Lagrangian solutions.

To handle the uncertainty in problem input parameters, stochastic programming and robust optimization have traditionally been employed to formulate these sequential decision-making problems. Both robust and stochastic facility location models which handle demand uncertainty in facility location problems are surveyed by Snyder (2006) and Correia and da Gama (2015).

Most models for multi-period stochastic facility location problems in the literature are formulated as a two-stage stochastic programs, where facility locations for the entire planning horizon are determined in the first stage, and then, in the second stage, transportation decisions are made. Multistage stochastic programming is also used to formulate stochastic multi-period facility location problems, but solving these problems is computationally challenging. Nickel et al. (2012) formulate a multi-period facility location problem with uncertainty in demand and interest rates as a multi-stage stochastic mixed-integer linear program. They observe that this model takes a long time to solve. For example, 6 hours is required to solve a 216 scenario instance using a commercial software solver. To tackle this class of the problems, researchers have explored a variety of approximation approaches, such as a heuristic combining branch and bound algorithm (Hernández et al., 2012), and fix and relax coordination approximation (Albareda-Sambola et al., 2013). Yu et al. (2018) compare two modeling frameworks, a two-stage stochastic model and a multistage stochastic model, for a multi-period facility location problems under risk-neutral and risk-averse measures. They employ a stochastic dual dynamic programming approach to solve these models.

Robust optimization is often applied when there are uncertain parameters with no distributional information available.

Atamtürk and Zhang (2007) study a model related to two-stage robust facility location problems, and recognize that evaluating the objective is already computationally intractable even for a single period problem. Gabrel et al. (2014) also consider a robust version of the facility location problems with uncertain demand. They use a two-stage formulation, and employ Kelley's cutting- plane algorithm to solve the model. Zeng and Zhao (2013) also propose a cutting plane algorithm to solve the same problem, and their method, a column and constraint generation algorithm, improves the computation time to solve the problem.

Baron et al. (2010) use robust optimization for a multi-period fixed charge facility location problem with uncertainty in demand with a profit maximization objective. They consider two types of demand uncertainty set: a box uncertainty set and an ellipsoid uncertainty set. They show that applying the robust optimization approach in this setting is tractable and improves on the nominal solution to the problem.

Ardestani-Jaafari and Delage (2018) and Bertsimas and de Ruiter (2016) also use a robust optimization approach for the problem described in Baron et al. (2010). Ardestani-Jaafari and Delage (2018) notice that the decisions of production and shipment can be delayed until demand is realized in each period in this problem, and add a small amount of flexibility to these delayed decisions to correct for the overly conservative solutions observed in the approach of Baron et al. (2010). This idea is implemented in six tractable conservative approximation models employing a different form of the application of affine adjustments. To tackle realistic size problem instances, a row generation algorithm is also presented. Bertsimas and de Ruiter (2016) apply affine adjustments on a dual reformulation of the same problem setting in Ardestani-Jaafari and Delage (2018), and show a significant improvement in computational time.

In most of the facility location literature, to handle the uncertainty in problem input parameters such as demand or cost, either stochastic programming or robust optimization is used. We consider a setting where (stochastic) demand in each period is determined by the outcome of a series of binary events, where the likelihood of outcomes is unknown. To handle this setting, our approach combines robust optimization and stochastic programming.

2.4 Robust Optimization

One drawback of the stochastic programming approach is that probability distributions must be known, although in many cases they are not. Robust optimization addresses this problem by minimizing the objective for the worst instances of an ambiguous parameter set. Ben-Tal et al. (2009) compile the research related to robust optimization.

Some of the models we present in this thesis incorporate ambiguity in the probability of trial success and in demand, and to solve these we employ a robust optimization approach.

We briefly review the robust optimization principle that we rely on in this thesis. Consider the following problem:

$$\begin{aligned} min \quad & c^T x \\ s.t. \quad & Ax \geq b \\ & l \leq x \leq u \end{aligned}$$

In a typical setting, we assume that c, b, and A are deterministic and we can often obtain an optimal solution by solving the problem using a commercial software. In robust optimization, parameter values lie within a so-called uncertainty set. The solution of the robust optimization problem is feasible when parameters take on any value within the uncertainty set. In this approach, the problem with uncertain parameters is reformulated as its robust counterpart by replacing each constraint that has uncertain coefficients constraints that reflects the uncertainty set. Thus, a key feature of the robust optimization approach is its tractability, which depends on the structure of the uncertainty set. Our approach is motivated by Bertsimas and Sim (2003), Bertsimas and Thiele (2006), and Thiele (2007), who demonstrate the tractability of this approach in other settings.

Specifically, in the problem above, we assume that the data uncertainty only affects the elements in matrix A, without any loss of generality. Each uncertain coefficient a_{ij} is known to belong to an interval centered at its nominal value $\bar{a}_{ij}$ and of half length $\hat{a}_{ij}$, but its exact value is unknown. The scaled deviation of parameter a_{ij} from its nominal value is also defined as z_{ij}, and total (scaled) variation of the parameters is limited to a threshold, known as the budget of uncertainty. Bertsimas and Sim (2003) show that uncertain linear programming problems can be formulated as robust linear programs in this setting and this budget of uncertainty allows greater flexibility to build a robust model without excessively affecting the optimal cost.

Bertsimas and Thiele (2006) apply this technique to supply chain problems. Thiele (2007) uses this idea to reformulate the min-max stochastic programming problem where the decision-maker minimizes the maximum expected cost over a family of probability distributions. In Bertsimas and Thiele (2006), the uncertainty set is defined for outcomes of the random variables, but in Thiele (2007) as well as in our research, it is defined for their probabilities. However, the approach in Thiele (2007) differs from ours in that it builds on a set of recursive equations for dynamic programs in order to formulate an approach for multi-stage stochastic problem, whereas we directly apply the robust optimization approach to the problem. In addition, we explore a variety of different objectives, and we also focus on algorithm development.

Chapter 3

Capacity Planning with Ambiguous Probabilities

3.1 Introduction

In this chapter, we consider a capacity planning in a setting where results of experiments, trials, or tests are received each period, and demand in any period is a function of results received up to that period. Specifically, we consider a single product discrete-time planning problem with a horizon of T time periods. In each period, the amount of the product that can be manufactured is constrained by available capacity. Capacity expansion costs are assumed to be known, and capacity investment decisions to expand the amount of available capacity can be made in any period. At the same time, over the horizon, demand evolves based on test outcomes. These tests are modeled as a series of Bernoulli experiments, one in each period, which have either a successful or an unsuccessful outcome. Given a particular series of outcomes, the demand is assumed to be known, and thus, demand uncertainty can be modeled using a scenario tree. This demand model is motivated by our experience working with biopharmaceutical firms, where, although demand is random, it is well-characterized given the outcome of a series of events, but can vary quite substantially depending on the outcome of these events (where events might be approval for a new indication, trial outcomes, introduction of a competing product, etc.). Several facility types to invest are considered which have different cost and construction lead time. This is also motivated by biopharmaceutical firms, where two options of facilities are available which are an expensive new technology equipment with shorter lead time to build and relatively cheaper traditional equipment with longer lead time.

In this setting, we consider the problem of deciding when and by how much to expand capacity in order to meet a variety of different objects, including minimizing expected cost, or the α-quantile of regret. When optimizing capacity investment decisions, it is often challenging to determine what objective to focus on. Our goal is to explore how alternative objectives affect solutions, and find some insights to be useful for decision makers. We initially assume that the probability of success of each Bernoulli trial is known, and we later extend our models to account for ambiguity in these probabilities. These models are typically challenging to solve when the planning horizon is relatively long (which is typical for capacity expansion planning), so we present heuristics for these models in Section 3.4.

The remainder of this chapter is organized as follows. In Section 3.2, we begin by formulating capacity expansion models where the probability of each trial success[1] is known, using a multistage stochastic mixed integer programming, following the approach introduced by Ahmed and Sahinidis (2003). We extend our basic model, which minimizes expected cost to a variety of other objectives. Finally, we extend these models to incorporate ambiguous probabilities. In Section 3.4, we develop heuristic algorithms for these models based on a rolling horizon approach. In Section 3.5, we present the results of computational experiments. We evaluate the performance of our heuristics, and we explore the impact of different objectives in this robust optimization setting.

3.2 Models with Unambiguous Probabilities

In this section, we present our preliminary models, which assume that the probability of each trial success is known. As described above, in this setting, demand uncertainty can be modeled by a scenario tree where each node n in stage t of the tree represents a realization of the demand, and each edge is labeled with the probability of a trial outcome. We denote this probability by π_n, which equals the conditional probability of reaching node n given parent node $A(n)$. Each node n of the tree, except the root ($n = 0$), has a unique parent $A(n)$, and each non-leaf node has children $C(n)$. A scenario p corresponds to a single path from the root to each unique leaf node. $P(p)$ is used to denote the set of nodes on the path of scenario p and $S(n)$ to denote the set of scenarios in which node n is included. The probability of realization of scenario p, denoted by θ^p, is computed by $\prod_{i \in P(p)} \pi_i$. A node in any stage t also can be identified with a "bundle" B of scenarios passing through it. We denote the set of bundles at time t by B_t. For example, at time 1, the set B_1 contains a single bundle consisting of all scenarios. Figure 3.1 illustrates the notation related to a scenario tree.

We utilize a fixed charge to model economies of scale in capacity investment. For each facility type i there is a unit capacity size U_i, and capacity can be expanded by multiples of U_i for facility type i. Our results generalize to continuous multiples of increment, but integer variables more closely model instances we have seen in practice, particularly for large unit capacity sizes. We summarize notation used to formulate the model as follows.

Sets and indices:

- P: Set of scenarios
- N: Set of nodes
- B_t: Bundle of scenarios passing through a node at time stage t
- t: Index for time periods ($t = 1, ..., T$)
- i: Index for facility type ($i = 1, ..., I$)

Parameters:

[1] Recall that when are using the terminology *trial* to represent a Bernoulli trial, which models an event with two possible outcomes, not specifically restricted to clinical trials.

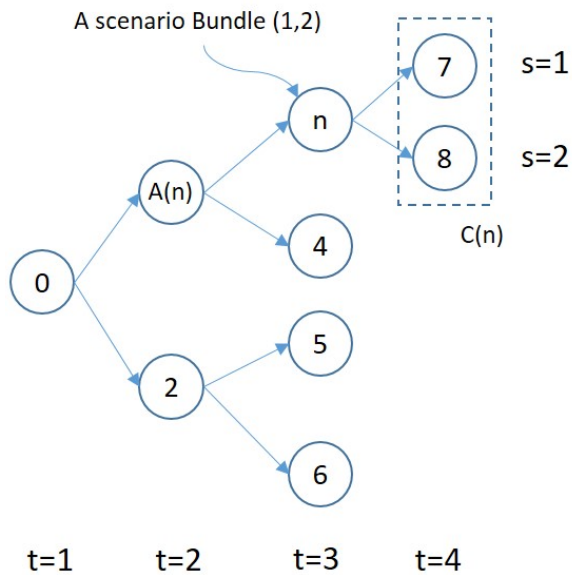


Figure 3.1. The scenario tree notation

- f_{it}: Fixed cost of facility type i at time t
- c_{it}: Construction cost per unit capacity of facility type i at time t
- s_t: Shortage penalty cost of product at time t
- U_i: Unit capacity of a facility i
- d_t^p: Demand at time t under a scenario p
- θ^p: Probability of realization of a scenario p ($\sum_{p \in P} \theta^p = 1$)
- l_i: Construction lead time of facility type i
- M: Sufficiently large number($\max_{p \in P} \sum_{t=1}^{T} \sum_{g \in G} d_{tg}^p$)

Decision Variables:

- x_{it}^p: 1 if capacity of facility type i is acquired at time t under scenario p, 0 otherwise
- y_{it}^p: Amount of capacity added to facility type i at time t under scenario p
- z_{it}^p: Cumulative capacity of facility type i at time t under scenario p
- w_t^p: Amount of shortage at time t under scenario p
- $F(p)$: Total cost for scenario p

Following Ahmed and Sahinidis (2003), the risk neutral model with known probabilities (**RN**) can be formulated as follows:

RN:

$$min \quad \sum_{p \in P} \theta^p F(p) \tag{3.1}$$

$$s.t. \quad F(p) = \sum_{t=1}^{T} (s_t w_t^p + \sum_{i=1}^{I} (f_{it} x_{it}^p + c_{it} y_{it}^p)), \quad p \in P \tag{3.2}$$

$$z_{it}^p = z_{i(t-1)}^p + U_i y_{i(t-l_i)}^p, \quad \forall i \leq I, t > l_i, p \in P \tag{3.3}$$

$$z_{it}^p = z_{i(t-1)}^p, \quad \forall i \leq I, t \leq l_i, p \in P$$

$$d_t^p - \sum_{i=1}^{I} z_{it}^p \leq w_t^p \quad t \leq T, p \in P \tag{3.4}$$

$$w_t^p \leq d_t^p \quad t \leq T, p \in P \tag{3.5}$$

$$M x_{it}^p \geq y_{it}^p, \quad \forall i \leq I, t \leq T, p \in P \tag{3.6}$$

$$x_t^{p_1} = x_t^{p_2}, y_t^{p_1} = y_t^{p_2}, s_t^{p_1} = s_t^{p_2}, \quad \forall (p_1, p_2) \in B_t, t \leq T \tag{3.7}$$

$$x_{it}^p \in \{0, 1\}, \quad \forall i \leq I, t \leq T, p \in P \tag{3.8}$$

$$y_{it}^p \in Z^+, \quad \forall i \leq I, t \leq T, p \in P \tag{3.9}$$

The objective function (3.1) minimizes the expected cost of fixed and construction cost and shortage cost. Constraint (3.3) describes the amount of available capacity of facility type i at time t under scenario p. Constraint (3.4) represents that the amount of shortage at time t under scenario p. Constraint (3.5) ensures that the amount of shortage cannot exceed demand at time t under scenario p. Constraint (3.6) ensures that the fixed cost is charged when additional capacity investment is made. Constraints (3.7) are known as non-anticipativity constraints. That is, if two scenarios p_1 and p_2 belong to the same node B_t of the scenario tree at time stage t then the corresponding decisions up to that point have to be identical. Constraints (3.8) and (3.9) enforce binary and integer conditions on variables.

Alternative Objectives

We also extend model **RN** by considering alternative objectives that minimize regret. The regret R^p associated with scenario p under a given solution is defined to be the difference between the cost of the best possible solution given scenario p, denoted by V^p and the cost under the given solution.

For the first extension, we minimize the maximum of the α-quantile of the regret, denoted by W, where α is the desired reliability level. The maximum regret is computed over a reliability set whose collective probability of occurrence is at least a desired reliability level α. This model only considers the α-quantile of the regret, and does not assess the magnitude of the regret associated with the scenarios that are not included in the reliability set. We follow the approach proposed by Daskin et al. (1997) for the p-median problem, but adapt it to our setting. In addition to the notation defined earlier, we define the binary variable n^p. n^p is to equal 1 if scenario p is included in the set over which the maximum regret is minimized, and 0 otherwise.

MMR:

$$\begin{aligned} & min && W && (3.10) \\ & s.t. && (3.2) - (3.9) \\ & && R^p - (F(p) - V^p) = 0, \quad \forall p \in P && (3.11) \\ & && W - R^p + M(1 - n^p) \geq 0, \quad \forall p \in P && (3.12) \\ & && \sum_{p \in P} \pi^p n^p \geq \alpha && (3.13) \\ & && n^p \in \{0, 1\}, \quad \forall p \in P && (3.14) \end{aligned}$$

The objective function (3.10) minimizes the α-reliable maximum regret. Constraint (3.11) defines the regret associated with scenario p. Constraint (3.12) defines W in terms of individual scenario regrets and indicator variables that denote the scenarios to be included in the maximum regret computation. Constraint (3.13) ensures that the probability associated with the set of scenarios over which W is computed must be at least α.

In **MMR**, the regret in the tail could be significantly higher than α-reliable maximum regret since the model does not control the regret in the tail. We formulate another model that considers the magnitude of regrets in the tail by minimizing the expectation of regrets associated with the scenarios in the tail, whose collective probability is no greater than $1-\alpha$. For this model, we closely follow G. Chen et al. (2006) model adapted from a p-median model to the capacity expansion setting. Before we present the actual model, we briefly explain the α- reliable mean-excess regret as follows.

Suppose X represents the vector for all decision variables not associated with regret for a model. In addition, we have the following decision variables associated with regret.

- $R(X, p)$: Regret with function of X and scenario p
- $f(X, \zeta)$: Given X, the collective probability of scenarios in which the regret does not exceed ζ, that is, $P\{p | R(X, p) \leq \zeta\}$
- $\zeta_\alpha(X)$: $\alpha-$ quantile regret, that is, $min\{\zeta : f(X, \zeta) \geq \alpha\}$
- $\phi_\alpha(X)$: Conditional probability weighted average of the regret which is greater than $\zeta_\alpha(X)$, that is, $\dfrac{\sum_{p:R(X,p)>\zeta_{\alpha(X)}} \pi^p R(X, p)}{\sum_{p:R(X,p)>\zeta_{\alpha(X)}} \pi^p}$

Then, the α-reliable mean-excess regret for a given feasible solution X is:

$$\lambda \zeta_\alpha(X) + (1 - \lambda)\phi_\alpha(X),$$

where $\lambda = \dfrac{f(X, \zeta_\alpha(X)) - \alpha}{1 - \alpha} \in [0, 1]$. As shown in Rockafellar and Uryasev (2000) and Rockafellar and Uryasev (2002), α-reliable mean-excess regret given X can be modified as follows:

$$\zeta + \frac{1}{1-\alpha}\sum_{p\in P}\pi^p Max\{[R(X,p)-\zeta],0\}$$

This can be further simplified by denoting U^p as the amount of individual scenario regret that exceeds ζ, which is $Max\{[R(X,p)-\zeta],0\}$.

Now, with these two additional variables ζ and U^p, the model that minimizes the α-reliable mean-excess regret with known probabilities can be formulated as follows:

MER: :

$$\begin{aligned}
&min && \zeta + \frac{1}{1-\alpha}\sum_{p\in P}\pi^p U^p && && (3.15)\\
&s.t. && (3.3)-(3.9), (3.11), (3.12) && && \\
& && U^p \geq R^p - \zeta && \forall p \in P && (3.16)\\
& && U^p \geq 0 && \forall p \in P && (3.17)
\end{aligned}$$

The objective function (3.15) minimizes the α-reliable mean-excess regret. Constraint (3.16) defines the amount of individual scenario regret that exceeds ζ, the α-quantile regret.

3.3 Models with Ambiguous Probabilities

Next, we extend models (**RN, MMR**, and **MER**) to model ambiguity in the probability of success of each Bernoulli trial. While in practice it may be possible to estimate these probabilities using relevant historical data, models (**RN, MMR**, and **MER**) are sensitive to small changes in probability estimates. Consider the following example:

Example 1. *We consider 4 periods of capacity expansion model with 2 types of facilities. Two facilities have different lead time and costs. Uncertain demand is described in Figure 3.2 as a scenario tree. For each branch of the scenario tree, the demand takes either upper node value or lower node value with probabilities* $p+\delta\hat{p}$ *and* $1-p-\delta\hat{p}$*, respectively. Parameters used in this example are* $l_1 = 0, l_2 = 1, s_t^p = 20, f_{1t} = 500, f_{2t} = 200, c_{1t} = 100$*,* $c_{2t} = 50, p = 0.5$ *and* $\hat{p} = 0.5$*. Now we solve the risk neutral model for 5 values of* $\delta : 0, -0.2, 0.2, -0.6, 0.6$*. The available capacity for each facility of each path in the optimal solution for these 5 values of* δ *is provided in Table 3.1. As shown, optimal solutions are quite different depending on probability distributions, but it is not easy to describe the changes using some simple function of* δ*. This example explains why we need an approach that considers ambiguous probabilities.*

This motivates us to consider a robust optimization(RO) approach. In RO, rather than assuming a particular distribution on the realization of uncertain data, we assume that possible realizations belong to a given set, the so-called uncertainty set. In particular, we employ an approach introduced by Bertsimas and Sim (2003), who develop a general framework of robust linear optimization and present a technique involving polyhedral uncertainty sets that leads to robust linear counterparts to ambiguous linear programs.

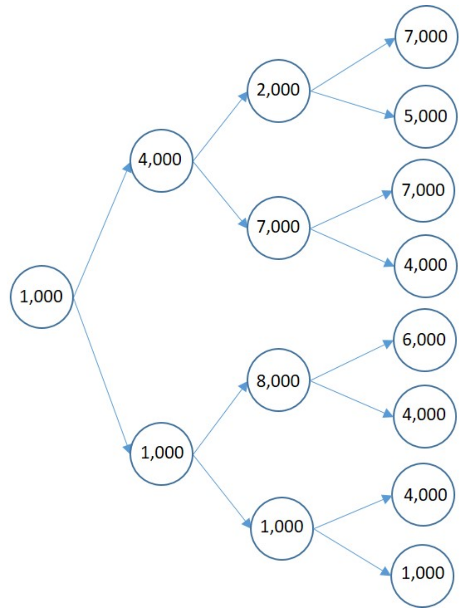


Figure 3.2. Demand for Example1

Table 3.1. Available capacity of each path in the optimal solutions under different probability distributions

δ	0	-0.2	+0.2	-0.6	+0.6
Path 1	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0, 0, 0)	(10,30, 0, 0)	(10, 0, 0, 0)
	(60, 0,0, 0)	(30,30, 0, 0)	(60, 0, 0, 0)	(0,30, 0, 0)	(60, 0, 0, 0)
Path 2	(10, 0, 0, 0)	(10, 0, 0, 0)	(10,0, 0, 0)	(10,30, 0, 0)	(10, 0, 0, 0)
	(60, 0,0, 0)	(30,30, 0, 0)	(60, 0, 0, 0)	(0,30, 0, 0)	(60, 0, 0, 0)
Path 3	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0,0, 0)	(10,30, 0, 0)	(10, 0, 0, 0)
	(60, 0,0, 0)	(30,30, 0, 0)	(60, 0, 0, 0)	(0,30, 0, 0)	(60, 0, 0, 0)
Path 4	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0, 0, 0)	(10,30, 0, 0)	(10, 0, 0, 0)
	(60, 0, 0, 0)	(30,30, 0, 0)	(60, 0, 0, 0)	(0,30, 0, 0)	(60, 0, 0, 0)
Path 5	(10, 0, 0, 0)	(10, 0,40, 0)	(10, 0, 0, 0)	(10, 0,70, 0)	(10, 0, 0, 0)
	(60,10, 0, 0)	(30, 0, 0, 0)	(60,10, 0, 0)	(0, 0, 0, 0)	(60,10, 0, 0)
Path 6	(10, 0, 0, 0)	(10, 0,40, 0)	(10, 0, 0, 0)	(10, 0,70, 0)	(10, 0, 0, 0)
	(60,10, 0, 0)	(30, 0, 0, 0)	(60,10, 0, 0)	(0, 0, 0, 0)	(60,10, 0, 0)
Path 7	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0, 0,30)	(10, 0, 0, 0)
	(60,10, 0, 0)	(30, 0, 0, 0)	(60,10, 0, 0)	(0, 0, 0, 0)	(60,10, 0, 0)
Path 8	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0, 0, 0)	(10, 0, 0, 0)
	(60,10, 0, 0)	(30, 0, 0, 0)	(60,10, 0, 0)	(0, 0, 0, 0)	(60,10, 0, 0)

We denote the ambiguous probability by $\tilde{\pi}_n$ to distinguish it from unambiguous probability π_n. $\tilde{\pi}_n$ belongs to an interval centered at $\bar{\pi}_n$ and of half length $\hat{\pi}_n$, but its exact value is ambiguous. The sum of probabilities branched from node n must satisfy $\sum_{i \in C(n)} \tilde{\pi}_i = 1$, where n can be any node except leaf nodes in a scenario tree. The probability of scenario p is computed by $\prod_{i \in P(p)} \tilde{\pi}_i$, which also becomes ambiguous and we denote this by

$\tilde{\theta}^p$. The nominal value of $\tilde{\theta}^p$, θ^p_{mid}, is $\prod_{i\in P(p)} \bar{\pi}_i$, and $\tilde{\theta}^p$ can take the value in the range of $[\prod_{i\in P(p)} \pi^-_n, \prod_{i\in P(p)} \pi^+_n]$ where $\pi^+_n = \bar{\pi}_n + \hat{\pi}_n$ and $\pi^-_n = \bar{\pi}_n - \hat{\pi}_n$. Formally, the uncertainty set Ω for ambiguous parameter $\tilde{\pi}_n$ is:

$$\Omega = \{\tilde{\pi} | \tilde{\pi}_n \in [\pi^-_n, \pi^+_n] \forall n \in N \setminus \{0\}, \sum_{i\in C(n)} \tilde{\pi}_i = 1, \forall n \in N, C(n) \neq \emptyset\}$$

To capture ambiguity in our models, we modify **RN**, **MMR**, and **MER** by replacing θ^p with $\prod_{i\in P(p)} \tilde{\pi}_i$ given Ω. These modified models minimize the objective values for the worst-case over Ω. However, in general, this approach may be too conservative (or pessimistic), so we adjust the level of "conservativeness" using a budget of uncertainty approach.

Specifically, we define the positive and negative scaled deviation of $\tilde{\theta}^p$ from its nominal value θ^p_{mid} as

$$z^p_+ = (\tilde{\theta}^p - \theta^p_{mid})/\hat{\theta}^p_+$$
$$z^p_- = (\tilde{\theta}^p - \theta^p_{mid})/\hat{\theta}^p_-$$

where $\hat{\theta}^p_+ = \prod_{i\in P(p)} \pi^+_n - \theta^p_{mid}$ and $\hat{\theta}^p_- = \theta^p_{mid} - \prod_{i\in P(p)} \pi^-_n$, respectively. The scaled deviations take values in $[0, 1]$. Then, we impose a budget of uncertainty, where the total variation of scenario realization probabilities cannot exceed some threshold Γ, not necessarily integer:

$$\sum_{p\in P} z^p_+ + z^p_- \leq \Gamma$$

We substitute Ω by Θ, which is the uncertainty set for $\tilde{\theta}^p$, in order to replace the non-linear term $\prod_{i\in P(p)} \tilde{\pi}_i$, and add in the budget of uncertainty to Θ. Note that $\tilde{\pi}_n$ can be represented as the sum of probabilities of scenario realizations that include node n divided by the sum of probabilities of realized scenarios that include the parent of node n, i.e., $\sum_{i\in S(n)} \tilde{\theta}^i / \sum_{i\in S(A(n))} \tilde{\theta}^i$. Then

$$\pi^-_n \leq \tilde{\pi}_n \leq \pi^+_n \Leftrightarrow \pi^-_n \sum_{i\in S(A(n))} \tilde{\theta}^i \leq \sum_{i\in S(n)} \tilde{\theta}^i \leq \pi^+_n \sum_{i\in S(A(n))} \tilde{\theta}^i$$

Also, the sum of scenario probabilities must satisfy $\sum_{p\in P} \tilde{\theta}^p = 1$. Under this transformation, the probability constraints $\sum_{i\in C(n)} \tilde{\pi}_i = 1$ become trivial, as we show below:

$$\sum_{i\in C(n)} \tilde{\pi}_i = 1 \Leftrightarrow \sum_{i\in C(n)} \sum_{k\in S(i)} \tilde{\theta}^k / \sum_{j\in S(n)} \tilde{\theta}^j = 1 \Leftrightarrow \sum_{j\in S(n)} \tilde{\theta}^j / \sum_{j\in S(n)} \tilde{\theta}^j = 1 \Leftrightarrow 1 = 1$$

Therefore, the uncertainty set Θ for $\tilde{\theta}^p$ is:

$$\Theta = \Bigg\{ \tilde{\theta} \in R^{|P|} | \tilde{\theta}_p = \theta^p_{mid} - \hat{\theta}^p_- z^p_- + \hat{\theta}^p_+ z^p_+,$$

$$0 \le z^p_+ \le 1, 0 \le z^p_- \le 1, \quad \forall p \in P,$$

$$\pi^-_n \sum_{i \in S(A(n))} \tilde{\theta}^i \le \sum_{i \in S(n)} \tilde{\theta}^i \le \pi^+_n \sum_{i \in S(A(n))} \tilde{\theta}^i, \quad \forall n \in N \setminus \{0\}$$

$$\sum_{p \in P} \tilde{\theta}_p = 1, \quad \sum_{p \in P} z^p_+ + z^p_- \le \Gamma \Bigg\}$$

Given this, the robust versions of **RN**, **MMR**, and **MER** that minimize their respective objective values for the worst-case over Θ can be written as follows:

$$\textbf{RN-R:} \quad min \quad \max_{\tilde{\theta} \in \Theta} \sum_{p \in P} \tilde{\theta}^p F(p) \tag{3.18}$$

$$s.t. \quad (3.2) - (3.9)$$

$$\textbf{MMR-R:} \quad min \quad W$$

$$s.t. \quad \sum_{p \in P} \tilde{\theta}^p n^p \ge \alpha \quad \forall \tilde{\theta} \in \Theta \tag{3.19}$$

$$(3.2) - (3.9), (3.11) - (3.12)$$

$$\textbf{MER-R:} \quad min \quad \max_{\tilde{\theta} \in \Theta} \zeta + \frac{1}{1 - \alpha} \sum_{p \in P} \tilde{\theta}^p U^p \tag{3.20}$$

$$s.t. \quad (3.2) - (3.9), (3.11), (3.16) - (3.17))$$

Since the ambiguous parameter, $\tilde{\theta}^p$ does not take on integer values, we can adapt the standard approach used for robust linear programming to this setting. We identify an auxiliary linear program that finds the worst-case probability for any given solution, and use this insight to arrive at the following theorem:

Theorem 1. ***RN-R***, ***MMR-R***, *and* ***MER-R*** *can be formulated as MILPs:*

$$\textbf{\textit{RN-PR:}} \quad min \quad \sum_{p \in P} \theta^p_{mid} F(p) + C' \tag{3.21}$$

$$s.t. \quad (3.2) - (3.9)$$

$$K^+(p) \ge F(p) \hat{\theta}^p_+, K^-(p) \ge -F(p) \hat{\theta}^p_- \quad p \in P \tag{3.22}$$

$$t \ge 0 \tag{3.23}$$

$$r^p_+ \ge 0, r^p_- \ge 0 \quad p \in P \tag{3.24}$$

$$w^n_+ \ge 0, w^n_- \ge 0 \quad \forall n \in N \setminus \{0\} \tag{3.25}$$

$$\textbf{\textit{MMR-PR:}} \quad \min \quad W$$
$$\begin{aligned} s.t. \quad & (3.2)-(3.9),(3.11),(3.12),(3.23)-(3.25) \\ & -\alpha \geq -\sum_{p\in P}\theta^p_{mid}n^p + C' \qquad (3.26) \\ & K^+(p) \geq -n^p\hat{\theta}^p_+, K^-(p) \geq n^p\hat{\theta}^p_- \quad p\in P \end{aligned}$$

$$\textbf{\textit{MER-PR:}} \quad \min \quad \zeta + \frac{1}{1-\alpha}\sum_{p\in P}U^p\theta^p_{mid} + C' \qquad (3.27)$$
$$\begin{aligned} s.t. \quad & (3.2)-(3.9),(3.11),(3.16),(3.17),(3.23)-(3.25) \\ & K^+(p) \geq \frac{1}{1-\alpha}U^p\hat{\theta}^p_+, \\ & K^-(p) \geq -\frac{1}{1-\alpha}U^p\hat{\theta}^p_- \quad p\in P \end{aligned}$$

where

$$\begin{aligned} C' =& t\Gamma + \sum_{p\in P}(r^p_+ + r^p_-) + \sum_{n\in C(0)}\hat{\pi}_n(w^n_+ + w^n_-) \\ & + \sum_{n\in N^-}\hat{\pi}_n\sum_{i\in S(A(n))}\theta^i_{mid}(w^n_+ + w^n_-) \\ K^+(p) =& q\hat{\theta}^p_+ + t + r^p_+ + Ka(\hat{\theta}^p_+) + Kb(\hat{\theta}^p_+) + Kc(\hat{\theta}^p_+) \\ K^-(p) =& -q\hat{\theta}^p_- + t + r^p_- + Ka(-\hat{\theta}^p_-) + Kb(-\hat{\theta}^p_-) + Kc(-\hat{\theta}^p_-) \\ Ka(a) =& \sum_{\substack{m\in P(p)\\ A(m)=0}} a(w^m_+ - w^m_-), \\ Kb(b) =& \sum_{\substack{n\in P(p)\\ A(n)\neq 0}} b\{(1-\pi^+_n)w^n_+ - (1-\pi^-_n)w^n_-\} \\ Kc(c) =& \sum_{\substack{A(n)\in P(p)\\ n\notin P(p)}} \{c(\pi^-_n w^n_- - \pi^+_n p w^n_+)\} \end{aligned}$$

Proof. The constraint (3.18) is equivalent to solving the following problem with variables z^p_+, z^p_-.

$$
\begin{aligned}
\text{RNX:}\quad & max\ \sum_{p\in P}(\theta^p_{mid}+\hat{\theta}^p_+ z^p_+ - \hat{\theta}^p_- z^p_-)F(p) && \\
& s.t.\ \ \tilde{\theta}^p = \theta^p_{mid}+\hat{\theta}^p_+ z^p_+ - \hat{\theta}^p_- z^p_- && (3.28)\\
& \sum_{p\in P}\tilde{\theta}^p = 1 && (3.29)\\
& \pi^-_n \sum_{i\in S(A(n))}\tilde{\theta}^i \le \sum_{i\in S(n)}\tilde{\theta}^i \qquad \forall n\in N\setminus\{0\} && (3.30)\\
& \sum_{i\in S(n)}\tilde{\theta}^i \le \pi^+_n \sum_{i\in S(A(n))}\tilde{\theta}^i \qquad \forall n\in N\setminus\{0\} && (3.31)\\
& \sum_{p\in P} z^p_+ + z^p_- \le \Gamma && (3.32)\\
& 0\le z^p_+\le 1 \qquad \forall p\in P && (3.33)\\
& 0\le z^p_-\le 1 \qquad \forall p\in P && (3.34)
\end{aligned}
$$

RNX can be rewritten as a minimization problem using strong duality, because the feasible set is nonempty ($z^p_+ = 0, z^p_- = 0$ is solution) and bounded.

We introduce dual variables $q, w^n_-, w^n_+, t, r^p_+$, and r^p_- for constraints (3.29), (3.30), (3.31), (3.32), (3.33) and (3.34). Then, the dual problem becomes:

$$
\begin{aligned}
& min\ \ C' \\
& s.t.\ \ K^+(p)\ge F(p)\hat{\theta}^p_+ \quad p\in P\\
& K^-(p)\ge -F(p)\hat{\theta}^p_- \quad p\in P\\
& t\ge 0\\
& r^p_+\ge 0, r^p_-\ge 0 \quad p\in P\\
& w^n_+\ge 0, w^n_-\ge 0 \quad \forall n\in N\setminus\{0\}
\end{aligned}
$$

Finally, by reinjecting this into the original problem, we can reformulate **RN-R** with an uncertainty budget as a Mixed-Integer Linear Program (MILP), **RN-PR**. Similarly, we can reformulate **MMR-R** and **MER-R** with uncertainty budget constraints as the MILPs **MMR-PR** and **MER-PR**. □

3.4 Heuristic Algorithms

As we detail in Section 3.5, we can solve only relatively small instances of our models using commercial optimization software. However, in our experience with the pharmaceutical industry, for example, a planning horizon of at least 20 periods (quarterly buckets for five years) is typically considered for planning, in light of the duration of clinical trials and the required lead times for construction and permitting. Thus we are motivated to find efficient heuristics for bigger problems.

We propose two heuristic approaches for this problem, a straightforward rolling horizon approach and the more sophisticated event spike approach, in this section. We adapt the idea of the event spike approach from Beltran-Royo et al. (2014), who initially developed it for stage-wise independent (where current stage results are independent of previous results) multistage stochastic LPs(MSLP). This approach requires significant modifications for our setting because we're optimizing an multistage stochastic *integer* program that is not stage-wise independent. In particular, we propose a version of the event spike approach that works for a special case of our problem setting, where events are "pathwise-independent" (so that events can depend on past events, but not on when those events occurred). This is applicable, for example, if events at the current stage depend on the number of previous successes, but not the timing of those previous successes. Later, we introduce an approach that allows us to relax this restriction.

3.4.1 Rolling Horizon

Before introducing the event-spike approach, we consider a rolling horizon approach, in which we solve the problem for a subset of the entire horizon in each period, and make decisions based on the truncated horizon. Of course, to evaluate the overall effectiveness of a policy like this, we need to determine the rolling horizon decisions along each of the possible paths through the scenario tree (stepping through one period at a time and incrementing the final period in the truncated horizon each time), and then average over solutions).

Specifically, in each iteration i, we solve the problem from period i to $i-1+r$, where r is the length of rolling horizon. At the beginning of each iteration i, decisions for nodes before time i are already fixed, and after the iteration i, decisions for nodes between period i to $i-1+r$ are re-solved. However only decisions in period i are fixed, and others are recalculated at each iteration. The set of nodes considered in each iteration i are decomposed into 2^{i-1} independent sub-trees, thus each sub-tree can be solved in parallel. In the last iteration, when $i = T - r + 1$, decisions for all nodes between i to T are fixed, and thus we obtain the final set of solutions (and can determine the expected cost of that set of solutions). A graphical example is shown in Figure 3.3. In this example, a rolling horizon $r = 3$ is considered. In iteration 1, a sub-tree from period 1 to 3 is solved, then the solution of the root node is fixed. In iteration 2, given the fixed decision at the root node, 2 sub-trees from time 2 to 4 are solved independently, and then the decisions for the two nodes in period 2 are fixed.

3.4.2 Event Spike Approach

More sophisticated approaches are also possible. As described at the beginning of this section, motivated by Beltran-Royo et al. (2014), we approximate the current problem with a smaller problem using a version of the so-called "event spike" approach. The original version of this approach proposed by Beltran-Royo et al. (2014) leverages conditional expectation for instances of MSLP where the stochastic process is stage-wise independent, so that the multistage scenario tree is very simple. In other words, at a given stage t, all the nodes have the same set of successors. For example, in Figure 3.4, at time 2, there are two nodes **A** and **B** that have same immediate successor events **C** and **D**. The term "event" here is defined as

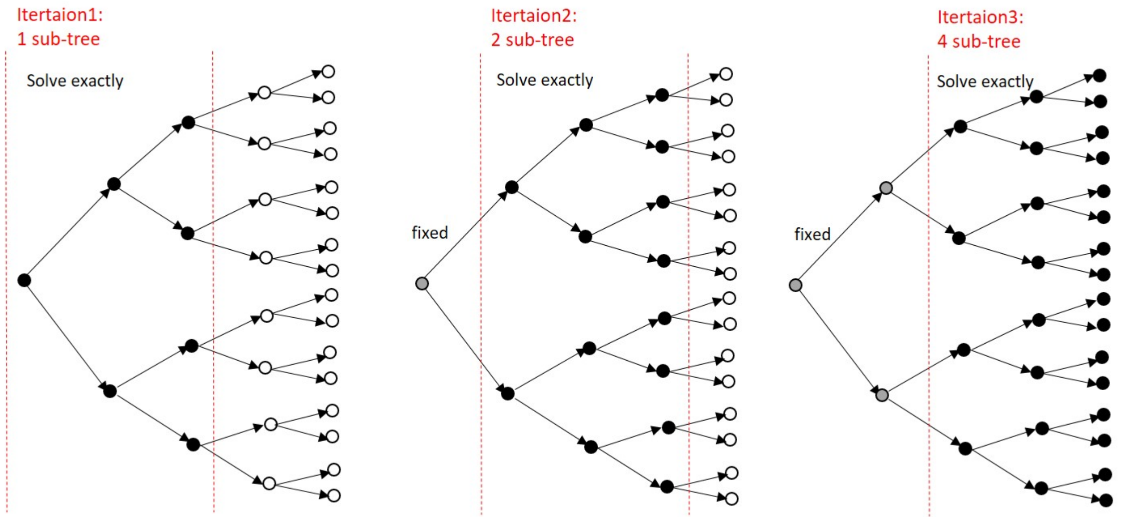


Figure 3.3. Concept of Rolling Horizon Approach

the realization of uncertainties (demand in our problem). Uncertainties are represented in a simpler way in the "event spike graph". That is, the event spike graph considers events, rather than scenarios. For instance, suppose only two events occur in each stage as shown Figure 3.4. All 8 possible paths are considered in the scenario tree, whereas 3 connected two stage trees are considered in the event spike graph. In a two-stage tree in an event spike graph at time t, the node of the first stage accounts for the expected value of decision variables at t, and the nodes of the second stage account for all the possible events and the corresponding decisions at $t+1$. The expected value of the decision variables at time $t+1$ is determined by taking the expectation of all the corresponding decision variables at $t+1$.

To apply the concept behind this approach to our problem, we start by reformulating **RN** into an equivalent node-based formulation **RN(N)**. In **RN(N)**, the objective and constraints are defined over nodes of the scenario tree. We summarize notation relevant to **RN(N)** below:

Parameters

- t: Index for time periods($t = 1, ..., T$)
- m: Index for node counters in time t($m = 1, ..., 2^{(t-1)}$)
- $\mathcal{P}(tm)$: Path from the root to a node tm
- p_{tm}: Probability of reaching node tm, that is, $\prod_{n \in \mathcal{P}(tm)} \pi_n$

Variables

- x_{itm}: 1 if capacity of facility type i is acquired at node tm, 0 otherwise
- y_{itm}: Amount of capacity added to facility type i at node tm
- z_{itm}: Cumulative capacity of facility type i at node tm
- w_{tm}: Demand shortage at node tm

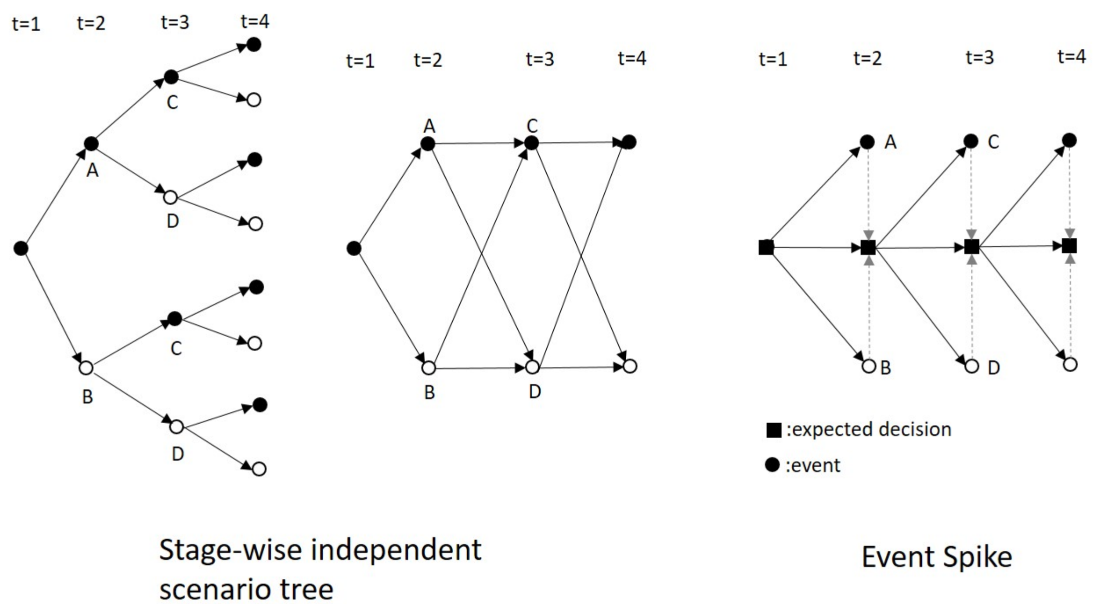


Figure 3.4. Concept of Event Spike Approach (Beltran-Royo et al., 2014)

Given this notation, the node-based formulation of **RN** is:

RN(N) :

$$min \quad \sum_{t=1}^{T}\sum_{m=1}^{2^{(t-1)}} p_{tm}(s_t w_{tm} + \sum_{i=1}^{I}(f_{it}x_{itm} + c_{it}y_{itm})) \tag{3.35}$$

$$s.t. \quad z_{itm} = z_{iA(tm)} + U_i y_{i(t-l_i)m}, \quad \forall i \leq I, t > l_i, m \leq 2^{(t-1)} \tag{3.36}$$

$$z_{itm} = z_{iA(tm)}, \quad \forall i \leq I, t \leq l_i, m \leq 2^{(t-1)}$$

$$d_{tm} - \sum_{i=1}^{I} z_{itm} \leq w_{tm}, \quad \forall t \leq T, m \leq 2^{(t-1)} \tag{3.37}$$

$$w_{tm} \leq d_{tm}, \quad \forall t \leq T, m \leq 2^{(t-1)} \tag{3.38}$$

$$Mx_{itm} \geq y_{itm}, \quad \forall i \leq I, t \leq T, m \leq 2^{(t-1)} \tag{3.39}$$

$$x_{itm} \in \{0,1\}, y_{itm} \in Z^+ \quad \forall i \leq I, t \leq T, m \leq 2^{(t-1)} \tag{3.40}$$

The objective function (3.35) minimizes the expected cost. Constraint (3.36) describes the amount of available capacity of facility type i at node tm. Constraint (3.37) determines that the amount of shortage at node tm. Constraint (3.38) ensures that the amount of shortage cannot exceed demand at node tm. Constraint (3.39) ensures that the fixed cost is charged when additional capacity investment is made. Note that we do not need non-anticipativity constraints in this formulation.

The original event spike approach developed in Beltran-Royo et al. (2014) assumes stage-wise independence. We relax this restriction, as described below, to a setting where events are "pathwise" independent but not necessarily stage-wise independent. Thus, demand at a node can depend on the number of previous trial successes, but not on when those successes occurred. In other words, for example, if $t = 4$, demand will be the same if there were two successful trials in the first four periods, regardless of when those successes occurred. While this is in many cases an approximation, it is significantly less constraining than insisting on stage-wise independence. We describe below how we combine this approach with a rolling horizon approach in a way that proves to be computationally effective in our general settings.

To modify and apply the event spike approach to $\mathbf{RN(N)}$, let us denote each node by tmk where k is the number of trial success, t is the time period, and m is a node counter. Nodes tmk and $tm'k$, that have k successes by time t, are collapsed into one node tk, which we call a pseudo-event node, as shown in Figure 3.5. A decision for pseudo-event node $(t+1)k$ is based on the decisions made as a consequence of pseudo-events tk and $t(k-1)$, and thus we take the expectation of these, and call this $t\bar{k}$. Additionally, we define the probability of reaching the pseudo-event node tk,

$$p'_{tk} = \sum_{m=1,(NS(tm)=k)}^{2^{t-1}} p_{tm}$$

where $NS(n)$ is the number of trial successes at node n. We also let $p'(t,t',k,k')$ is the probability that number of trial successes at time t and t' are k and k' respectively.

For each pseudo-event node tk and facility type i, we define decision variables, $\hat{x}_{itk}, \hat{y}_{itk}$ and $\hat{z}_{itk}$. The cumulative capacity is the input to the next stage, and thus the expected decision is defined for $\bar{z}_{itk}$, which is expressed as

$$\bar{z}_{itk} = \frac{1}{p'_{(t+1)k}} \sum_{k'=\max(k-1,1)}^{\min(k,t)} p'(t+1,t,k,k')\hat{z}_{itk'}$$

When there is a construction lead time, the capacity expansion decision made at time $t - l_i$ is added to cumulative capacity at stage t, and thus the expected decision is also defined for $\bar{y}_{itk}$, so

$$\bar{y}_{itk} = \frac{1}{p'_{(t+l_i)k}} \sum_{k'=\max(k-l_i,1)}^{\min(k,t)} p'(t+l_i,t,k,k')\hat{y}_{itk'}$$

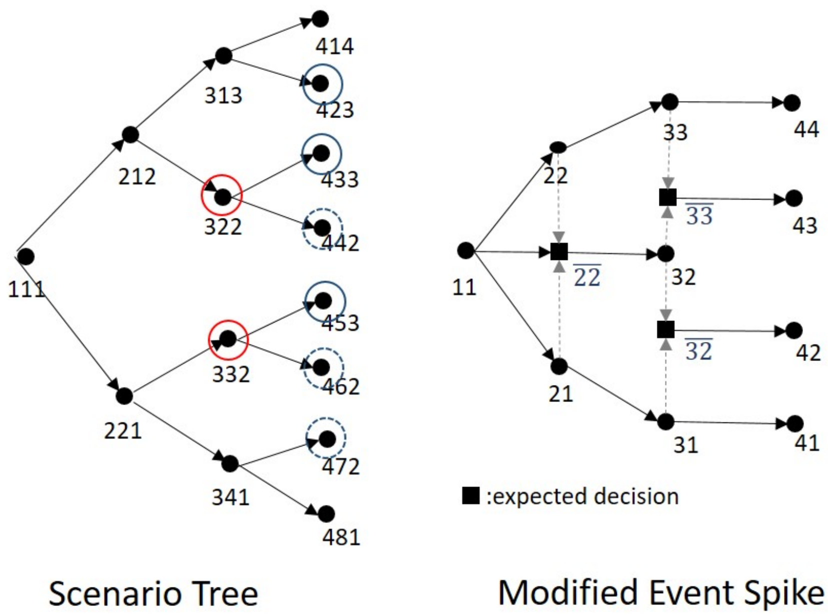


Figure 3.5. Modified Event Spike Approaches for our model

The event spike version of our problem with a risk neutral objective can be formulated as a MILP, **RN(ES)**, and the detailed formulation is as follows.

RN(ES) :

$$
\begin{aligned}
&min && \sum_{t=1}^{T}\sum_{k=1}^{t} p'_{tk}(s_t\hat{w}_{tk} + \sum_{i=1}^{I}(f_{it}\hat{x}_{itk} + c_{it}\hat{y}_{itk})) && \\
&s.t. && \hat{z}_{itk} = \bar{z}_{i(t-1)k} + U_i\hat{y}_{itk} && \forall i \le I, k \le t, t > l_i \\
& && z_{itk} = z_{i(t-1)k} && \forall i \le I, k \le t, t \le l_i \\
& && d_{tk} - \sum_{i=1}^{I}\hat{z}_{itk} \le \hat{w}_{tk}, \hat{w}_{tk} \le d_{tk} && \forall t \le T, k \le t \\
& && M\hat{x}_{itk} \ge \hat{y}_{itk} && \forall i \le I, t \le T, k \le t \\
& && \bar{z}_{itk} = \frac{1}{p'_{(t+1)k}} \sum_{k'=\max(k-1,1)}^{\min(k,t)} p'(t+1,t,k,k')\hat{z}_{itk'} && \forall i \le I, t < T, k \le t+1 \\
& && \bar{y}_{itk} = \frac{1}{p'_{(t+l_i)k}} \sum_{k'=\max(k-l_i,1)}^{\min(k,t)} p'(t+l_i,t,k,k')\hat{y}_{itk'} && \forall i \le I, t \le T-l_i, k \le t+1 \\
& && \hat{x}_{itk} \in \{0,1\}, \hat{y}_{itk} \in Z^+ && \forall i \le I, t \le T, 2 \le k \le t-1 \\
& && 0 \le \hat{x}_{itk} \le 1, \hat{y}_{itk} \ge 0 && \forall i \le I, t \le T, k = 1, t
\end{aligned}
$$

Lemma 2. ***RN(ES)*** *is a relaxation of* ***RN(N)*** *when the stochastic process is pathwise independent.*

Proof. We show that **RN(ES)** can be derived from the constraint aggregation induced by the conditional expectation operator applied to objective function and constraints of **RN(N)** as follows. The objective function (3.35) can be rewritten as follows.

$$\begin{aligned}
&\sum_{t=1}^{T}\sum_{t=1}^{2^{(t-1)}} p_{tm}(s_t w_{tm} + \sum_{i=1}^{I}(f_{it}x_{itm} + c_{it}y_{itm})) \\
&= \sum_{t=1}^{T} E[s_t w_{tm}] + \sum_{i=1}^{I} E[f_{it}x_{it} + c_{it}y_{it}] \\
&= \sum_{t=1}^{T} s_t E[w_{tm}] + \sum_{i=1}^{I} f_{it}E[x_{it}] + c_{it}E[y_{it}] \\
&= \sum_{t=1}^{T} s_t E[E[w_{tm}|d_t]] + \sum_{i=1}^{I} f_{it}E[E[x_{it}|d_t]] + c_{it}E[E[y_{it}|d_t]]
\end{aligned}$$

Constraint (3.36) can be aggregated by using the conditional expectation operator.

$$\begin{aligned}
E[z_{itm}|d_t] &= E[z_{iA(tm)} + U_i y_{i(t-l_i)m}|d_t] \\
&= E[z_{iA(tm)}|d_t] + U_i E[y_{i(t-l_i)m}|d_t] \\
&= E[E[z_{iA(tm)}|d_{t-1}]|d_t] + U_i E[E[y_{i(t-l_i)m}|d_{t-l_i}]|d_t]
\end{aligned}$$

Constraint(3.37) can be aggregated by using the conditional expectation operator.

$$E[d_{tm}|d_t] - E[\sum_{i=1}^{I} z_{itm}|d_t] \le E[w_{tm}|d_t] \Leftrightarrow d_{tk} - \sum_{i=1}^{I} E[z_{itm}|d_t] \le E[w_{tm}|d_t]$$

$E[d_{tm}|d_t] = d_{tk}$ holds by assumption that demand depends only on the number of success and total time.

Constraint(3.38) can be aggregated by using the conditional expectation operator.

$$E[w_{tm}|d_t] \le E[d_{tm}|d_t] \Leftrightarrow E[w_{tm}|d_t] \le d_{tk}$$

Constraint(3.39) can be aggregated by using the conditional expectation operator.

$$ME[x_{itm}|d_t] \ge E[y_{itm}|d_t]$$

Integral constraints (3.40) can also be aggregated by using the conditional expectation operator.

$$\begin{aligned}
&0 \le E[x_{itm}|d_t] \le 1 && \forall i \le I, t \le T, m \le 2^{(t-1)} \\
&E[y_{itm}|d_t] \ge 0 && \forall i \le I, t \le T, m \le 2^{(t-1)}
\end{aligned}$$

Thus, applying the conditional expectation operator to **RN(N)** results in the following problem with aggregated constraints:

P1 :

$$\begin{aligned}
min \quad & \sum_{t=1}^{T}\sum_{i=1}^{I} f_{it}E[E[x_{it}|d_t]] + c_{it}E[E[y_{it}|d_t]] \\
s.t. \quad & E[z_{itm}|d_t] = E[E[z_{iA(tm)}|d_{t-1}]|d_t] + U_iE[E[y_{i(t-l_i)m}|d_{t-l_i}]|d_t] && \forall i \leq I, k \leq t, t > l_i \\
& E[z_{itm}|d_t] = E[E[z_{iA(tm)}|d_{t-1}]|d_t] && \forall i \leq I, k \leq t, t \leq l_i \\
& d_{tk} - \sum_{i=1}^{I} E[z_{itm}|d_t] \leq E[w_{tm}|d_t] && \forall t \leq T, k \leq t \\
& E[w_{tm}|d_t] \leq d_{tk} && \forall t \leq T, k \leq t \\
& ME[x_{itm}|d_t] \geq E[y_{itm}|d_t] && \forall i \leq I, t \leq T \\
& 0 \leq E[x_{itm}|d_t] \leq 1, E[y_{itm}|d_t] \geq 0 && \forall i \leq I, t \leq T
\end{aligned}$$

Let $\hat{x}_{itk} = E[x_{itm}|d_t], \hat{y}_{itk} = E[y_{itm}|d_t], \hat{z}_{itk} = E[z_{itm}|d_t], \hat{w}_{tk} = E[w_{tm}|d_t]$, and the problem **P1** becomes as follows.

P2 :

$$\begin{aligned}
min \quad & \sum_{t=1}^{T}\sum_{i=1}^{I} f_{it}E[\hat{x}_{itk}] + c_{it}E[\hat{y}_{itk}] \\
s.t. \quad & \hat{z}_{itk} = E[\hat{z}_{iA(tmk)}] + U_iE[\hat{y}_{i(t-l_i)mk}] && \forall i \leq I, k \leq t, t > l_i \\
& \hat{z}_{itk} = E[\hat{z}_{iA(tmk)}] && \forall i \leq I, k \leq t, t \leq l_i \\
& d_{tk} - \sum_{i=1}^{I} \hat{z}_{itk} \leq \hat{w}_{tk} && \forall t \leq T, k \leq t \\
& \hat{w}_{tk} \leq d_{tk} && \forall t \leq T, k \leq t \\
& M\hat{x}_{itk} \geq \hat{y}_{itk} && \forall i \leq I, t \leq T \\
& 0 \leq \hat{x}_{itk} \leq 1, \hat{y}_{itk} \geq 0 && \forall i \leq I, t \leq T
\end{aligned}$$

Replacing $E[\hat{z}_{itk}] = \bar{z}_{itk}$, $E[\hat{y}_{itk}] = \bar{y}_{itk}$, we see that **P2** is equivalent to **RN(ES)**. Let $z^*_{RN(ES)}, z^*_{P_1}, z^*_{P_2}$ and $z^*_{RN(N)}$ be the optimal objective values of model **RN(ES), P1, P2** and **RN(N)** respectively. Then, we will prove that

$$z^*_{RN(ES)} = z^*_{P_2} \leq z^*_{P_1} \leq z^*_{RN(N)}$$

$z^*_{P_1} \leq z^*_{RN(N)}$ follows from the fact that problem **P1** is a relaxation of **RN(N)** by constraint aggregation. To prove $z^*_{P_2} \leq z^*_{P_1}$, assume that $X^* = (x, y, z, w)$ is the optimal solution for **P1** and define $\hat{X}$ such that

$$\hat{X}_{itk} = E[X_{itm}|d_t]$$

Then, $\hat{X}$ is a feasible solution for **P2**, and thus $z^*_{P_1}(\hat{X}) = z^*_{P_2}(X^*)$, and this proves $z^*_{P_2} \leq z^*_{P_1}$. Therefore, we can conclude that **RN(ES)** is a relaxation of **RN(N)**. □

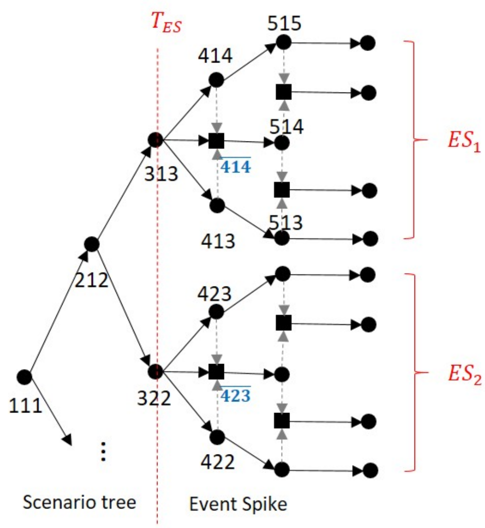


Figure 3.6. Delayed Event Spike

3.4.3 The Delayed Event Spike Approach

In some cases, the "pathwise" independence restriction required for the event spike approach is quite limiting. In this section, we introduce an approach to partially relax this restriction.

Recall that in the event spike approach, we *collapse* the scenario tree to efficiently develop a heuristic solution. In the Delayed Event Spike (DES) approach, as the name implies, we delay the time period in which the scenario tree is collapsed. In other words, until a given time period, T_{ES}, the original scenario tree is maintained, and after T_{ES}, the remaining portion of the scenario tree is approximated by using the event spike approach, as shown in Figure 3.6. In DES, we are not restricted to pathwise independence prior to T_{ES}, and events after T_{ES} can depend on past events, and if those events occurred before or at T_{ES}, on the timing of those events.

For example, demand at a node at $t = 4 (T_{ES} = 2)$ can depend on the number of previous successes, and when those successes occurred if they occurred before or at $T_{ES} = 2$. Now suppose there were two successful trials in the first four periods. Those successes could occur in periods one and two, one and three, or one and four, and let corresponding demand be d_1, d_2 and d_3, respectively. Then, d_2 and d_3 must be the same, but d_1 can be different. We call this "partial- pathwise independence".

In DES, at time T_{ES} there are $B = 2^{T_{ES}-1}$ nodes, and B sub-event spike trees are constructed rooted from these nodes. Let ES_b be the sub-event tree, where $b = 1...B$. Nodes tmk and $tm'k$, which have the same ancestor node at time T_{ES} and represent k successes by time t, are collapsed into one pseudo-event node tbk. We define the set of event nodes prior to T_{ES} and pseudo-event nodes at or after T_{ES} as follows:

$$E = \{tlk | l = m, m = 1..2^{t-1}, \forall t < T_{ES}, l = b, b = 1...B, \forall t \geq T_{ES}\}$$

Each sub-event tree ES_b has $O_t = t - T_{ES} + 1$ pseudo-event nodes at t.

As with the event spike approach described in the previous sub-section, a decision for pseudo-event node $(t + 1)bk$ is based on the decisions made as a consequence of pseudo-events tbk and $tb(k-1)$, and thus we take the expectation of these, and call this $t\bar{b}k$. The probability of reaching node tlk is therefore

$$p''_{tlk} = \begin{cases} p_{tm} & l = m, t \leq T_{ES} \\ \sum_{\substack{B(tm)=b,\\ NS(tm)=k}} p_{tm} & l = b, t > T_{ES} \end{cases}$$

where $B(tm) = \lceil m/B \rceil$ is the group b to which node tm belongs. In addition, we define

$$ED_1 = \{tbk | t > T_{ES}, k - r_b = 0\}$$

and

$$ED_m = \{tbk | t > T_{ES}, k - r_b = t - T_{ES}\}$$

where $r_b = NS(T_{ES}b)$, the number of trial successes at the root node of ES_b. Given these definitions and assumptions, the DES version of the problem can be formulated as a MILP. The detailed formulation is as follows:

RN(DES):

$$\begin{aligned}
min \quad & \sum_{tlk \in E} p''_{tlk} F(tlk) \\
s.t. \quad & \hat{z}_{itmk} = \hat{z}_{iA(tmk)} + U_i \hat{y}_{itmk}, \quad \forall i \leq I, t \leq T_{ES}, m \leq 2^{t-1}, k \leq t && (3.41) \\
& \hat{z}_{itbk} = \bar{z}_{i(t-1)bk} + U_i \hat{y}_{itbk}, \quad \forall i \leq I, t > T_{ES}, b \in B, k \leq t && (3.42) \\
& F(tlk) = s_t w_{tlk} + \sum_{i=1}^{I} (f_{it} \hat{x}_{itlk} + c_{it} \hat{y}_{itlk}), \quad \forall tlk \in E && (3.43) \\
& d_{tlk} - \sum_{i=1}^{I} \hat{z}_{itlkp} \leq \hat{w}_{tlk}, \quad \forall tlk \in E && (3.44) \\
& \hat{w}_{tlk} \leq d_{tlk}, \quad \forall tlk \in E && (3.45) \\
& M \hat{x}_{itlk} \geq \hat{y}_{itlk}, \quad \forall tlk \in E, i \leq I && (3.46) \\
& \bar{z}_{itbk} = \hat{z}_{itbk}, \quad \forall i \leq I, tbk \in ED_1 && (3.47) \\
& \bar{z}_{itb(k+1)} = \hat{z}_{itbk}, \quad \forall i \leq I, tbk \in ED_m && (3.48) \\
& \bar{z}_{itbk} = G_i(tbk), \quad \forall i \leq I, t \geq T_{ES}, tbk \notin ED_1 \cup ED_m, && (3.49) \\
& \hat{x}_{itmk} \in \{0,1\}, \hat{y}_{itmk} \in Z^+, \quad \forall i \leq I, t \leq T_{ES}, k \leq t && (3.50) \\
& \hat{x}_{itbk} \in \{0,1\}, \hat{y}_{itbk} \in Z^+, \quad \forall i \leq I, tbk \in ED_1 \cup ED_m \\
& 0 \leq \hat{x}_{itbk} \leq 1, \hat{y}_{itbk} \geq 0, \quad \forall i \leq I, t > T_{ES}, tbk \notin ED_1 \cup ED_m
\end{aligned}$$

where $G_i(tbk) = \frac{1}{p''_{(t+1)bk}} \sum_{k'=\max(k-1,1)}^{\min(k,t)} p''(b, t+1, t, k, k') \hat{z}_{itbk'}$, and $p''(b, t, t', k, k')$ is probability that the number of trial success at time t and t' are k and k', respectively, and node tm for any m belongs to block b. Note that if $T_{ES} = 1$, the formulation is the same as the formulation of **RN(ES)**. For simplicity, we do not include the construction lead time, but it can be implemented in a straightforward way.

A rolling horizon approach is often used to obtain good feasible solutions for multistage stochastic linear programs (see, e.g., Beltran-Royo et al., 2014). As we illustrate in Figure

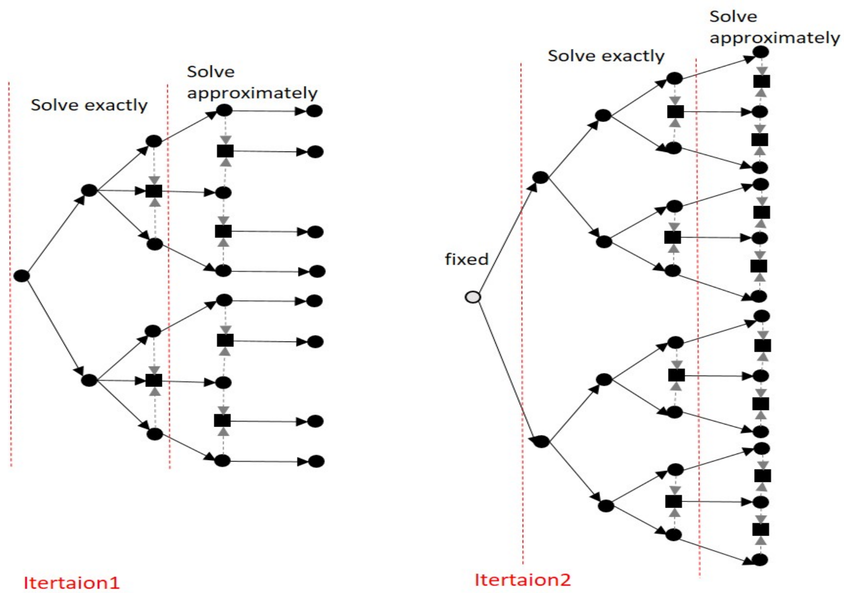


Figure 3.7. Concept of rolling horizon with event spike

3.7, we similarly use a rolling horizon strategy in this case. We solve the approximate problem using the DES approach for the entire remaining horizon, fix decisions in the current period, and then move forward.

Specifically, in iteration/time stage i, we solve the problem for periods i to T using DES, where the tree in periods $i + T_{ES}$ to T is approximated by an event spike tree. At the beginning of each iteration i, decisions for nodes before time i are fixed, and after the iteration i, decisions for nodes between i and T are re-solved. However, at each stage, only decisions for period i are fixed, and other decisions are recalculated in future iterations. In the final iteration, when $i = T - T_{ES}$, all decisions are fixed. This algorithm solves the problem sequentially, and the decisions for nodes in the starting period of each iteration are feasible for **RN(N)**, so after the algorithm terminates, we obtain a feasible solution for **RN(N)**.

We can also approximate a problem that is not pathwise independent by using the conditional expectation of instances of the problem given time t and the number of successes. This approximate problem does not provide a lower bound on **RN(N)** because it is not a relaxation of **RN(N)** (which requires pathwise independence). However, the decisions for the first period of the approximate problem is feasible for the original problem, and thus, we can find a feasible solution for problems without pathwise independence using the DES approach in a rolling horizon setting.

3.4.4 Delayed Event Spike Approach for Models Minimizing Regrets

In **MMR** and **MER**, regret is associated with specific scenarios, but scenarios are aggregated in the event spike approach. Thus, we need to an approach to represent regret in the event spike approach in order to apply DES. We note that a scenario can be approximated by one of the paths to reach the pseudo-event node Tbk at time T in the event spike, and thus we can define a mapping between pseudo-event node Tbk and scenario p. For example, in Figure 3.6, pseudo-event nodes 515 and 514 can be mapped to scenario 1, and to scenarios 2 and 3, respectively. Let $\mathcal{M}_{Tbk}$ be a set of scenarios which is mapped to pseudo-event node Tbk. We denote cumulative total cost to reach node tlk by $CF(tlk)$, which can be expressed as

$$CF(tmk) = \begin{cases} F(tmk) + CF(A(tmk)), \text{for} \quad t <= T_{ES} \\ \bar{F}((t-1)bk) + CF(A(tbk)), \text{for} \quad t > T_{ES} \end{cases}$$

where $F(tlk)$ is the total cost at each node and $\bar{F}(tbk)$ is the expected cost based on decisions for pseudo-events tbk and $tb(k-1)$. Given V^p, the minimum cost given scenario p, we let V_{Tbk} be the minimum cost to reach pseudo-event node Tbk, which is the average over all of the minimum cost scenarios mapped into Tbk:

$$V_{Tbk} = \sum_{(p,Tbk)\in\mathcal{M}} \frac{\pi^p V^p}{\sum_{(p,Tbk)\in M} \pi^p}$$

Then, the regret R_{Tbk}, defined as difference between the total cost to reach pseudo-event node Tbk under a given solution and the expected minimum cost of the corresponding scenarios leading to that node, is expressed as follows.

$$R_{Tbk} = CF(Tbk) - V_{Tbk}$$

We define the binary variable n_{Tbk} to be equal to 1 if at least one scenario $p \in \mathcal{M}_{Tbk}$ is included in the set (called the reliability set) over which W is minimized, and 0 otherwise. Because it is possible that only part of scenarios in $\mathcal{M}_{Tbk}$ can be included in the reliability set, we introduce a fractional variable n'_{Tbk} to compute the sum of probabilities in the reliability set, which must be at least α. Then, the **MMR** and **MER** models based on the DES approach can be formulated as MILPs, which we denote by **MMR(DES)** and **MER(DES)**, respectively.

MMR(DES):

$$
\begin{aligned}
&min \;\; W \\
&s.t. \;\; (3.41) - (3.50) \\
&\quad CF(tmk) = F(tmk) + CF(A(tmk)) \quad t \leq T_{ES} && (3.51)\\
&\quad CF(tbk) = \bar{F}((t-1)bk) + CF(A(tbk)) \quad t > T_{ES} && (3.52)\\
&\quad V_{Tbk} = \sum_{(p,Tbk)\in\mathcal{M}} \frac{\pi^p V^p}{\sum_{(p,Tbk)\in M} \pi^p} \quad \forall Tbk \in E && (3.53)\\
&\quad R_{Tbk} - (CF(Tbk) - V_{Tbk}) = 0 \quad \forall Tbk \in E && (3.54)\\
&\quad \bar{F}(tbk) = F(tbk) \quad tbk \in ED_1 && (3.55)\\
&\quad \bar{F}(tbk) = F(tb(k-1)) \quad tbk \in ED_m && (3.56)\\
&\quad \bar{F}(tbk) = \frac{p''_{tbk}F(tbk) + p''_{tb(k-1)}F(tb(k-1))}{p''_{tbk} + p''_{tb(k-1)}} \quad tbk \notin ED_1 \cup ED_m && (3.57)\\
&\quad W - R_{Tbk} + M(1 - n_{Tbk}) \geq 0 \quad Tbk \in E && (3.58)\\
&\quad \sum_{Tbk\in E} p''_{Tlk} n'_{Tbk} \geq \alpha && (3.59)\\
&\quad n'_{Tbk} - n_{Tbk} \leq 0 \quad Tbk \in E && (3.60)\\
&\quad n'_{Tbk} \in \{0,1\} \quad Tbk \in ED_1 \cup ED_m, n_{Tbk} \in \{0,1\} && (3.61)\\
&\quad 0 \leq n'_{Tbk} \leq 1 \quad Tbk \notin ED_1 \cup ED_m && (3.62)
\end{aligned}
$$

Constraint (3.58) defines W in terms of individual regret to reach pseudo-event nodes in the last period, as well as indicator variables that denote paths to the nodes in the last period that are to be included in the maximum regret computation. Constraints (3.59) and (3.60) together ensure that the probability associated with the set of scenarios over which W is computed must be at least α.

Similarly, **MER** based on the DES approach can be formulated as follows.

MER(DES):

$$
\begin{aligned}
&min \;\; \zeta + \frac{1}{1-\alpha} \sum_{Tbk\in E} p''_{Tlk} U_{Tbk} \\
&s.t. \;\; (3.41) - (3.57) \\
&\quad U_{Tbk} \geq R_{Tbk} - \zeta \quad Tbk \in E && (3.63)\\
&\quad U_{Tbk} \geq 0 \quad Tbk \in E && (3.64)
\end{aligned}
$$

As in the previous section, we can utilize a rolling horizon approach, in which we solve the approximate problem using **MMR(DES)** or **MER(DES)** for the entire remaining horizon, fix decisions in the current period, and then move forward.

3.4.5 Delayed Event Spike Approach with Ambiguous Probabilities

Models with ambiguous probabilities also can be approximated using the DES approach. To do this, we first develop robust versions of the DES models developed in previous subsection (**RN(DES)**, **MMR(DES)** and **MER(DES)**), and then use the approach described in Section 3.3 to derive equivalent MILP models. Below, we illustrate the details of this approach using **RN(DES)**.

We first develop the uncertainty set Ω' for ambiguous probability $\tilde{\pi}_{tm}$.

$$\Omega' = \{\tilde{p}|\pi^-_{tm}\tilde{p}_{A(tm)} \leq \tilde{p}_{tm} \leq \pi^+_{tm}\tilde{p}_{A(tm)}, \sum_{m\in 2^{t-1}} \tilde{\pi}_{tm} = 1, \forall t \leq T\}$$

To do this, we substitute $\tilde{p}_{tm} = \prod_{n\in\mathcal{P}(tm)} \tilde{\pi}_n = \tilde{\pi}_{tm}\tilde{p}_{A(tm)}$ into Ω. For simplicity, we do not consider an uncertainty budget, but this can be incorporated in a straightforward way. Recall that we use the probability of reaching pseudo-event node tlk ($\tilde{p}''_{tlk}$) to formulate **RN(DES)**, and thus we define the uncertainty set in terms of $\tilde{p}''_{tlk}$. Let this uncertainty set be $\Omega'' = \Omega''_1 \cup \Omega''_2$, where

$$\Omega''_1 = \{\tilde{p}|\pi^-_{tmk}\tilde{p}_{A(tm)} \leq \tilde{p}_{tmk} \leq \pi^+_{tm}\tilde{p}_{A(tm)}, \sum_{m\in 2^{t-1}} \tilde{\pi}_{tm} = 1, \forall t \leq T_{ES}\}$$

$$\Omega''_2 = \{\tilde{p}|\pi^-_{tbk} \leq \tilde{p}_{tbk} \leq \pi^+_{tbk}, \sum_{b\leq B, k\leq t} \tilde{\pi}_{tbk} = 1, \tilde{p}_{T_{ES}bk} = \sum_{k\leq t} \tilde{p}_{tbk}, \forall t > T_{ES}\}$$

When the probabilities of successes are ambiguous, both the objective function and constraints (3.49) in **RN(DES)** must be modified using the robust optimization approach. Given Ω'', the objective function of **RN(DES)** with ambiguous probabilities and equality constraint (3.49) can be rewritten as follows:

$$min\{\max_{\tilde{p}_{tlk}\in\Omega''} \sum_{tlk\in E} \tilde{p''}_{tlk}(\sum_{i=1}^{I}(f_{it}\hat{x}_{itlk} + c_{it}\hat{y}_{itlk}))\} \tag{3.65}$$

$$\bar{z}_{itbk} - G_i(tbk) \leq 0 \quad \forall\tilde{p}_{tbk} \in \Omega'' \tag{3.66}$$

$$-\bar{z}_{itbk} + G_i(tbk) \leq 0 \quad \forall\tilde{p}_{tbk} \in \Omega'' \tag{3.67}$$

Using strong duality, we can derive the deterministic robust counterpart of this model, and thus formulate a MILP for the robust version of **RN(DES)**. We denote this model by **RN(DES)-PR**, and the detailed formulation is as follows:

$$
\begin{aligned}
min \quad & \sum_{t=1}^{T} q_t + \sum_{tlk \in E} (p^+_{tlk}\gamma^+_{tlk} - p^-_{tlk}\gamma^-_{tlk}) \\
s.t. \quad & (3.41) - (3.48), (3.50) \\
& q_1 = F(111), \delta_{11} = 0 \\
& q_t + \delta_{tmk} - \gamma^-_{tmk} + \gamma^+_{tmk} - \delta_{A(tmk)} \geq F(tmk), && 2 \leq t < T_{ES}, tmk \in E \\
& q_t + \delta_{tbk} - \gamma^-_{tbk} + \gamma^+_{tbk} - \delta_{A(tbk)} + \sum_{\tau=1}^{T-T_{ES}+1} \zeta_{\tau b} \geq F(tbk), && j \leq 2^{t-1}, t = T_{ES} \\
& q_t - \gamma^-_{tbk} + \gamma^+_{tbk} - \delta_{A(tbk)} \geq F(tbk), && t = T_{ES} + 1 \\
& q_t - \gamma^-_{tbk} + \gamma^+_{tbk} - \zeta_{tb} \geq F(tbk), && t \geq T_{ES} + 2 \\
& \sum_{k \leq t} (p''^+_{tbk}\gamma^d_{tbk} - p''^-_{tbk}\gamma^c_{tbk}) + p^+_{T_{ES}bk}\gamma^d_{T_{ES}b} - p^-_{T_{ES}bk}\gamma^c_{T_{ES}b} \leq G_i(tbk) && \forall i \leq I, t \geq T_{ES}, \\
& && tbk \notin ED_1 \cup ED_m \\
& \nu^+_{tbk} + \gamma^d_{tbk} - \gamma^c_{tbk} \geq \bar{z}_{itbk} - \hat{z}_{itbk} \\
& \nu^+_{tbk} + \gamma^d_{tb(k-1)} - \gamma^c_{tb(k-1)} \geq \bar{z}_{itbk} - \hat{z}_{itb(k-1)} \\
& \nu^+_{tbk} + \gamma^d_{tbl} - \gamma^c_{tbl} \geq 0 \quad l \neq m, m-1 \\
& -\nu^+_{tbk} + \gamma^d_{T_{ES}b} - \gamma^c_{T_{ES}b} \geq 0 \\
& \sum_{k \leq t} (p''^+_{tbk}\gamma^g_{tbk} - p''^-_{tbk}\gamma^f_{tbk}) + p^+_{T_{ES}bk}\gamma^g_{T_{ES}b} - p^-_{T_{ES}bk}\gamma^f_{T_{ES}b} \leq -G_i(tbk) && \forall i \leq I, t \geq T_{ES}, \\
& && tbk \notin ED_1 \cup ED_m \\
& \nu^-_{tbk} + \gamma^g_{tbk} - \gamma^f_{tbk} \geq -\bar{z}_{itbk} + \hat{z}_{itbk} \\
& \nu^-_{tbk} + \gamma^g_{tb(k-1)} - \gamma^f_{tb(k-1)} \geq -\bar{z}_{itbk} + \hat{z}_{itb(k-1)} \\
& \nu^-_{tbk} + \gamma^g_{tbl} - \gamma^f_{tbl} \geq 0 && l \neq m, m-1 \\
& -\nu^-_{tbk} + \gamma^g_{T_{ES}b} - \gamma^f_{T_{ES}b} \geq 0 \\
& \gamma^+_{tlk} \geq 0, \gamma^-_{tlk} \geq 0, && tlk \in E \\
& \gamma^c_{tbk} \geq 0, \gamma^d_{tbk} \geq 0, \gamma^f_{tbk} \geq 0, \gamma^g_{tbk} \geq 0 \quad \forall k \leq t, \gamma^f_{T_{ES}b} \geq 0, \gamma^g_{T_{ES}b} \geq 0
\end{aligned}
$$

3.5 Computational Study

In the previous sections, we proposed a series of MILP models with varying objectives and ambiguous success probabilities, as well as heuristic approaches to solve these problems. Here, we conduct a numerical study to answer the following questions:

- How large are the models that we can solve to optimality using commercial software? How effective are the proposed heuristics for instances that are too large to solve to optimality?

- What is the impact of explicitly modeling ambiguous probabilities of success in this class of planning models? When is it worth doing so?
- How do alternative objective functions change capacity expansion decisions? Are there the decisions that these models make robust to alternative objectives?

We implement and test our models on a desktop computer with a Core i7 3.4 GHz processor and 16 GB of RAM. We code all models using Optimization Programming Language(OPL) and solve them using CPLEX, version 12.6, from IBM. Recall that we use the following abbreviations for models presented in Section 3.2.

- **RN**: Model minimizing expected cost with nominal probabilities
- **RN-PR**: Model minimizing expected cost with ambiguous probabilities
- **MMR**: Model minimizing α- reliable max regret with nominal probabilities
- **MMR-PR**: Model minimizing α- reliable max regret with ambiguous probabilities
- **MER**: Model minimizing α- reliable mean excess regret with nominal probabilities
- **MER-PR**: Model minimizing α- reliable mean excess regret with ambiguous probabilities

3.5.1 Computational Analysis of Heuristic Approaches

In this section, we solve the models in Section 3.2 and 3.3 using both commercial software and the two heuristics presented in Section 3.4. We compare the time to solve models using these approaches. Our goal is to determine when we can solve models to optimality using commercial software, when we need to use heuristics, and how effective those heuristics are when we do use them.

We solve the models presented in Section 3.2 and 3.3 using CPLEX version 12.6, as described above, with a 3,600 seconds time limit. We vary the planning horizon T in these tests, because the size of the problem depends on T. Recall that total number of scenarios is $|P| = 2^{T-1}$ and the number of variables and constraints increases as P increases. For each planning horizon T, we solve (or attempt to solve) 5 problem instances, randomly generated as follows: We consider one capacity type, and keep all parameters constant over time (and so drop the subscript t in the subsequent explanation). We draw an integer value for each parameter U_i, c_i, and f_i independently from $Unif(500, 1500), Unif(100, 300)$ and $Unif(50, 100)$ distributions, respectively, and we assume no construction lead time, so $l_i = 0$. We randomly select shortage penalty s from $\{0.1, 1, 2, 3\}$, where each possibility has equal probability. Each Bernoulli trial has a probability of success in the range $(0.25, 0.75)$, and for nominal models, we assume the center of that range, 0.5. The demand in the first period is generated from a $Unif(0, 1000)$ distribution. From the second period on, if the trial result is a success, demand equals $A + B$, where A is the demand from the last period if this is not the first success, A is randomly selected from $Unif(2000, 4000)$ distribution if this is the first success, and B is randomly selected from $Unif(500, 1500)$ distribution. If the trial result is not a success, demand remains the same as in the previous period.

The average computation time and the average optimality gap observed in these experiments are presented in Table 3.2. For each problem instance, we solve each model, denoted by m, to optimality or to the 3,600 second time limit, whichever comes first. We denote the objective function value by $z_{MIP}(m)$. Small instances, such as instances with $T < 10$ for **RN, RN-PR ,MER, MER-PR**, and **MMR-PR**, and with $T < 8$ for **MMR**, can be solved in less than a few seconds using CPLEX. Large instances cannot be solved optimally within the 1 hour time limit. We increase T until a feasible integer solution is not found within the time limit or an out-of-memory error is returned. If a feasible solution is found but an optimal solution is not, we report the optimality gap and the time that the best feasible solution is found.

As expected, we observe that these models become harder to solve as T increases. For **RN** and **RN-PR**, instances with $T > 13$ require at least $1,200$ seconds to find an optimal solution, and problem instances with $T = 16$ result in a greater than 20% optimality gap on average after an hour of computation. For **MMR**, **MMR-PR**, **MER**, and **MER-PR**, we observe that on average, the instances with at most 8, 10, 10, and 10 period planning horizons, respectively, can be solved in less than $1,200$ seconds with less than 10% gap. Unsurprisingly, we observe that incorporating ambiguous probabilities into our nominal models does not increase computation time significantly.

We refer to solution attempts using CPLEX as MILP. In Table 3.2, we compare the computational performance of MILP with that of our heuristics, simple rolling horizon (SRH), and delayed event spike with rolling horizon (DERH).

For the two heuristics, we must select values for r (the rolling horizon time) and T_{ES} (the starting time period where the scenario tree is collapsed to event spike). As r and T_{ES} are increased, there will be a trade-off between solution quality and computation time. For small instances ($T \leq 12$ for **RN**, $T \leq 9$ for **MMR**, and $T \leq 10$ for **MER**), we set $r = T - 1$ and $T_{ES} = T - 2$. For large instances, we set $r = 11, 8$, and 9 for **RN**, **MMR**, and **MER**, respectively. If suggested r is the time horizon for the problem instance, as shown in Table 3.2, it takes less than 80 seconds to solve models using MILP. The computation time to solve each sub-problem in both heuristics depends on the number of nodes in the problem. The time to solve each sub-problem with an r-period time horizon in SRH is similar to the time to solve a r-period time horizon problem using MILP. In DERH, if the number of event nodes in each sub-problem is close to the number of nodes in an r-period MILP problem, their computation times will be similar. We choose T_{ES} for which the total number of events in the first iteration is closest to the number of nodes in an $r + 1$-period MILP problem. (Note that since the number of event nodes in each sub-problem decreases as we move forward in the rolling horizon, we compare with an $r + 1$ rather than an r-period problem.)

We denote the objective values of the solutions found using SRH and DERH by $z_{SRH}(m)$ and $z_{DERH}(m)$, respectively. Each sub-problem in the heuristics is solved using CPLEX, but we set a 300 second time limit and a 3% relative MIP gap (when $r \geq 10$) rather than solving sub-problems to optimality.

To determine the effectiveness of the heuristic solutions, we compute the gap from the best bound, $z_{MIP}(m)(1 - (\text{MIP gap})/100)$, for $h =$ SRH, DERH:

$$\frac{z_h(m) - z_{MIP}(m)(1 - (\text{MIP gap})/100)}{z_h(m)}$$

Table 3.2. Average computation time and optimality gap using commercial software and heuristics

	(RN)						**(RN-PR)**					
	MILP		SRH		DERH		MILP		SRH		DERH	
T	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)
8	0.99	0	0.43	0	0.36	0	0.42	0	0.6	0.1	0.71	0.2
9	1.79	0	0.62	0.3	3.6	0	3.45	0	0.98	1.5	0.63	0.02
10	15.6	0	11.3	0.1	3.8	0.14	110.6	0	2.8	1.1	0.95	1.6
11	51.0	1.2	7.3	3.2	25.3	1.7	74.8	0.4	1.9	3.1	2.77	1.9
12	226.6	2.7	91.9	4.3	32.3	3.4	277.5	1.9	10	3.7	7.1	3.1
13	448.4	3.2	88.4	5.2	65.7	4.5	426.9	2.9	48.7	5.6	17.0	4.5
14	2,414	5.0	55.8	8.6	94.3	5.2	2,103	4.4	47.8	8.6	24.02	6.5
15	3,163	9.9	353	10.8	415.3	4.7	3,085	7	240	6.8	52.2	6.3
16	3,600	24.2	222	10	402.7	7.9	3,600	21.5	235.1	10.6	113.8	9.1
	(MMR)						**(MMR-PR)**					
	MILP		SRH		DERH		MILP		SRH		DERH	
T	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)
8	19.3	0	1.8	6	10.8	0	3.9	0	1.46	0	1.24	0
9	1,365	12.5	1.5	14.1	67.3	8.8	22.9	0	2.9	11.5	3.6	1.3
10	1,660	100+	33.2	4.9*	251.6	(1.8)*	264.3	0	18.1	8.2	27.6	2.2
11	3,600	100+	69.9	(27.4)*	335.2	(41.8)*	1,541	1.1	23.6	4.8	40	3.8
	(MER)						**(MER-PR)**					
	MILP		SRH		DERH		MILP		SRH		DERH	
T	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)
8	2.2	0	0.8	6.7	0.94	0	4.6	0	1.03	1.4	0.7	0
9	13.5	0	3.3	0	11.7	0	26.8	0	3.8	0	5.2	0
10	536.8	0	21.5	13.1	135.3	0.9	1,203	0	25.1	5.6	89.5	4.1
11	3,052	9.7	81	21.6	103.4	10.6	1,429	0.3	55.8	4.0	51.2	1.0
12	3,600	27.3	193	31	396.1	21.2	2,933	22.2	216.9	11.6	249.6	9.8

When the MIP gap is greater than 100% (**MMR** with $T = 10, 11$), we compute the gap from z_{MIP}, which is noted with $*$ in Table 3.2. In this case, z_{MIP} is not a lower bound on the optimal solution, so the gap can be negative.

We observe that the two heuristics perform better than the MILP in terms of computation time and optimality gap, especially for large problem instances. For example, for **RN**, when $T \leq 14$, both SRH and DERH find solutions to problem instances in less than 100 seconds, and the average gaps increase only at most 3.6% and 1.3% for SRH and DERH. When $T = 16$, we observe that the average gaps decrease by 14.2% and 16.3% for SRH and DERH compared to MILP. Note that in **MER**, although there are huge improvements in computation time using the heuristics, the average gaps using SRH increase by up to 13.1% in our examples. However, the average gaps using DERH only increase by at most 0.9% and improve by 6.1% when $T = 12$. Thus, we can conclude that DERH performs better than SRH in this case. Similarly, the average gaps are smaller for DERH than for SRH when we are solving other models, although the computation time is slightly greater for DERH than for SRH. This is because DERH considers all remaining periods, whereas SRH considers only truncated periods in the rolling horizon.

For instances that are too large to be solved using the MILP, we can use our heuristic approaches to find a feasible solution. We randomly generate 5 instances with $T = 20$ as described above for models **RN** and **RN-PR**, and solve them using SRH and DERH. For **RN**, the average computation times using SRH and DERH are $1,180$ and $1,350$ seconds, respectively, while the average expected cost is 10% lower using DERH than using SRH. For **RN-PR**, the average computation times using SRH and DERH are $2,600$ and 436 seconds, respectively. DERH requires significantly less computation time than SRH, especially when solving later periods in the rolling horizon. **RN-PR** tends to build more capacity in the early periods, and solutions of DERH also have a similar tendency, which makes the problem easier to solve later in the rolling horizon. Recall that the objective of **RN-PR** is to minimize worst-case expected cost. It is computationally challenging to find the exact worst-case value over all possible probabilities for these large instances, so we use simulation to compare the objective function values from SRH and DERH. We simulate 50 sample scenario trees with resolved probability uncertainty, and compute the expected costs of two heuristic solutions. We compute empirical CDFs of expected costs of solutions from the two heuristics over these 50 sample scenario trees, and observe that on average, the worst-case expected cost is 5% lower using DERH than using SRH. In other words, DERH can provide better solutions than SRH for large instances, consistent with what we saw in small instances.

Figure 3.8 shows an example of empirical cumulative distribution function (ECDF) of expected costs of solutions from our two heuristics. We observe that on average, the worst case expected cost is 5% lower using DERH than using SRH.

Overall, commercial software can effectively solve problems with shorter time horizon, that is, $13, 13, 8, 10, 10, 9$ for **RN**, **RN-PR**, **MMR**, **MMR-PR**, **MER**, and **MER-PR**, respectively. For problems that are too large to solve optimally using commercial software, our heuristic, DERH, is a very effective alternative.

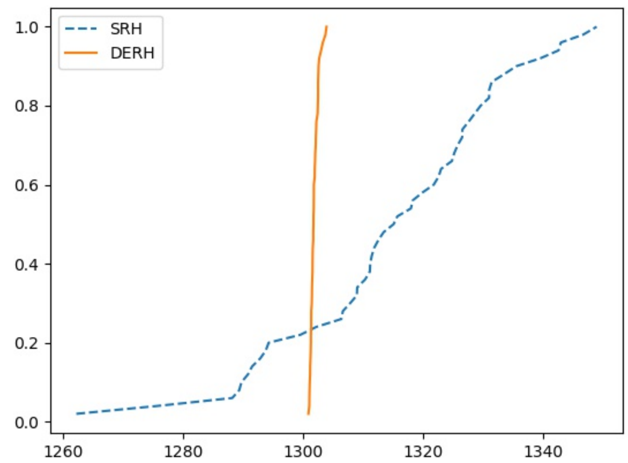


Figure 3.8. An example of ECDF of expected costs for two heuristic solutions of **RN-PR** ($T = 20$)

3.5.2 When Is It Worth Explicitly Modeling Ambiguous Success Probabilities?

To address the question in the title of this section, we compare solutions for a set of instances of each model presented in Section 3.2. The goal of these tests is to develop insight into the value of incorporating ambiguous probabilities into these models of capacity expansion, and also to explore the impact of alternative objective functions on capacity expansion decisions (note that we discuss this second goal in the subsequent section). The test instances we choose are intended to capture various computationally tractable decision-making scenarios in this setting.

We employ a 10 period of planning horizon for these test problems. We consider a primary facility type (denoted "Type A") with 3 periods of associated construction lead time. We describe demand uncertainty using a scenario tree with 512 paths. The demand in the first three periods is 0. From the 4th period on, if there have been at least four successful trials, demand equals $3000 + 1000 *$ (additional number of successes), and otherwise, demand remains the same as in the previous period. See Table 3.3 for details.

Table 3.3. Demand Structure of a test instance

	Time Period						
Path	1	2	3	4	5	...	10
1	0	0	0	3000	4000	...	9000
2	0	0	0	0	3000	...	8000
3	0	0	0	0	3000	...	8000
.	.	.	.	.	.	...	.
.	.	.	.	.	.	...	.
.	.	.	.	.	.	...	.
512	0	0	0	0	0	...	0

We set the probability of success of each node in the scenario tree, $\tilde{\pi}_n$, to fall in the range of $(0.25, 0.75)$ centered at $\bar{\pi} = 0.5$. We use $\bar{\pi}$ and $\tilde{\pi}_n$ to solve nominal and robust models, respectively.

To assess the sensitivity of results to parameters, we consider the following parameter settings. We keep all parameters constant over time, so we drop the subscript t. We vary incremental capacity unit size $U_A = \{1000, \mathbf{2000}, 3000, 4000\}$, construction cost $c_A = \{200, \mathbf{300}, 400, 500\}$ per capacity unit U_A, and fixed setup cost $f_A = \{\mathbf{0}, 100, 200, 300\}$. We also vary the shortage cost for every unit of unmet demand as follows: $s = \{0.1, 1, \mathbf{2}, 3\}$. Additionally, we consider a setting where the firm has both facility Type A and an additional facility (denoted "Type B") to choose from. This additional facility Type B has shorter construction lead time (l_B) and smaller incremental unit capacity (U_B), but higher construction cost (c_B). In our experiments, we set $f_B = 0$ and vary c_B, U_B and l_B as follows: $(C_B, U_B, l_B) = \{(160, 1000, 2), (90, 500, 1), (50, 250, 0)\}$

In our experiments, to generate test cases, we vary each parameter in turn, and keep other parameters at their **boldfaced** base setting from the lists above.

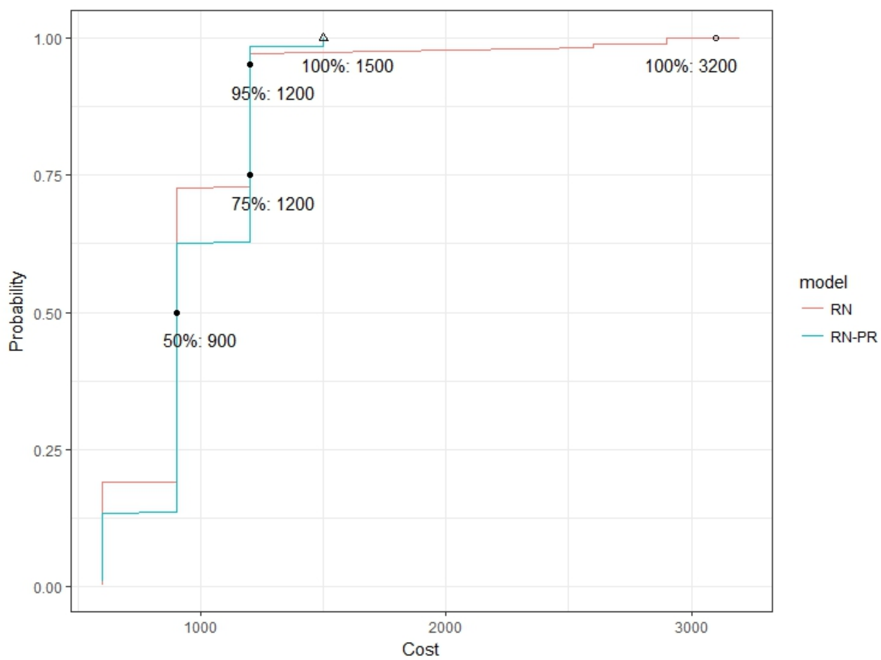
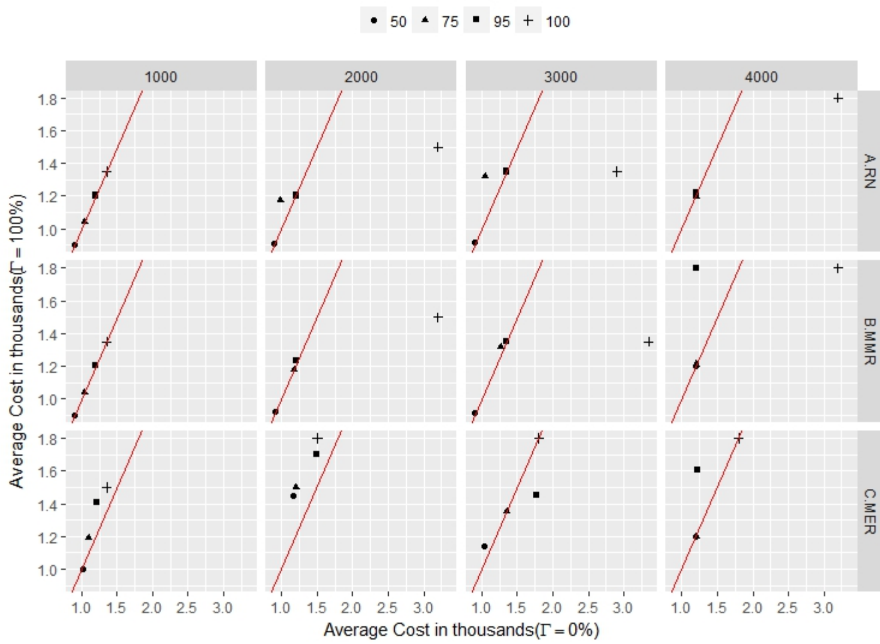


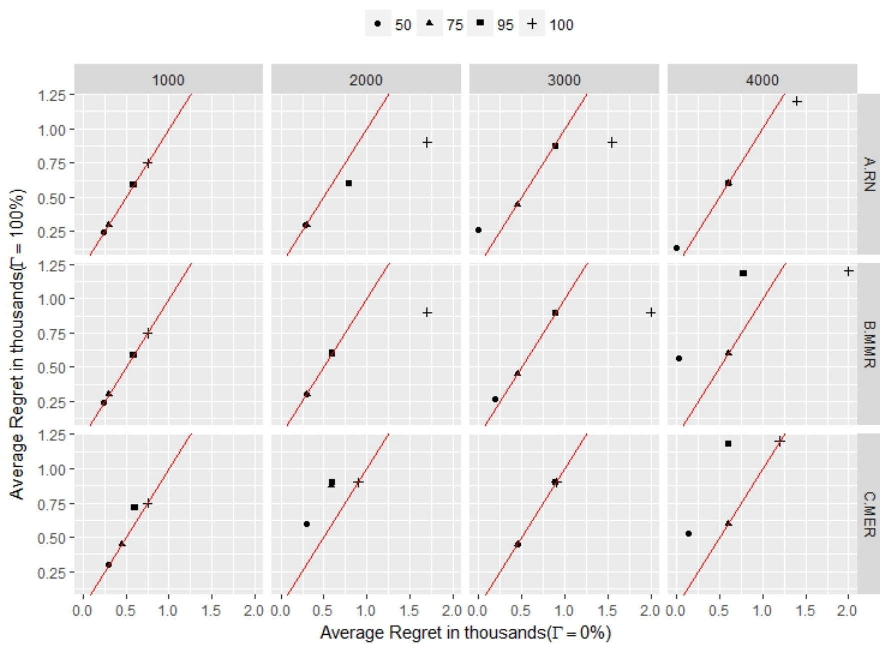
Figure 3.9. An example of Cumulative Distribution Function of costs

For each set of test parameters, we solve each of the models **(RN, RN-PR, MMR, MMR-PR**, **MER**, and **MER-PR)** to optimality, and then simulate 100 sample scenario trees with resolved probability uncertainty to find the distribution of outcomes. Specifically, for each simulation run, the (resolved) success rate of each node π_n is randomly generated from $Unif(0.25, 0.75)$ distribution in our tests. We then calculate the cumulative distribution function(CDF) of costs and regrets for each of the simulated scenario trees. For example, Figure 3.9 illustrates the CDF of costs for a sample simulated scenario tree given solutions to an instance of models **RN** and **RN-PR**. We then compute the average of the 50th, 75th, 95th, and 100th percentile of costs and regret over the 100 simulation runs.

In Figures 3.10-3.13, we compare the average percentiles for nominal versus robust models in our setting with ambiguous event probabilities, in an attempt to characterize settings where robust models add value.

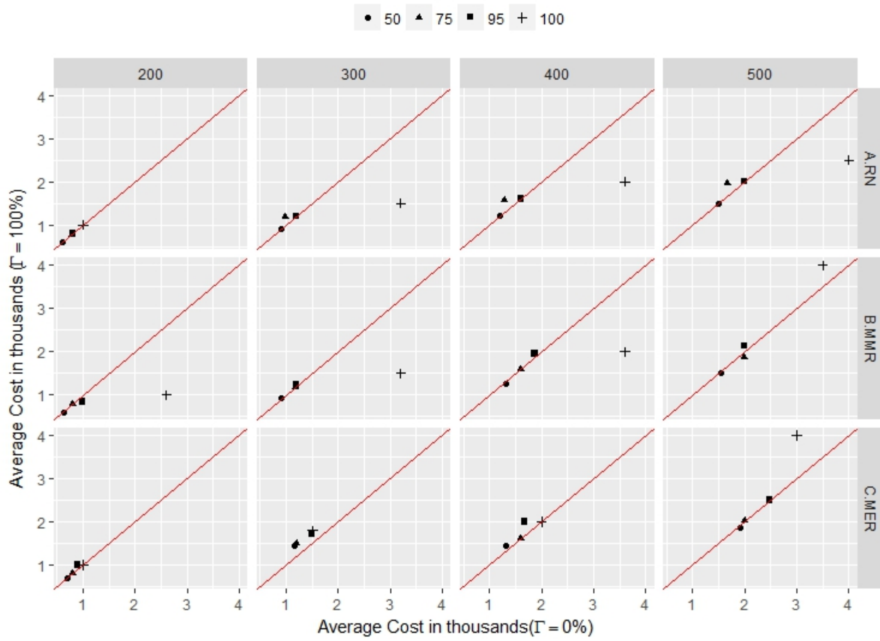


(a)

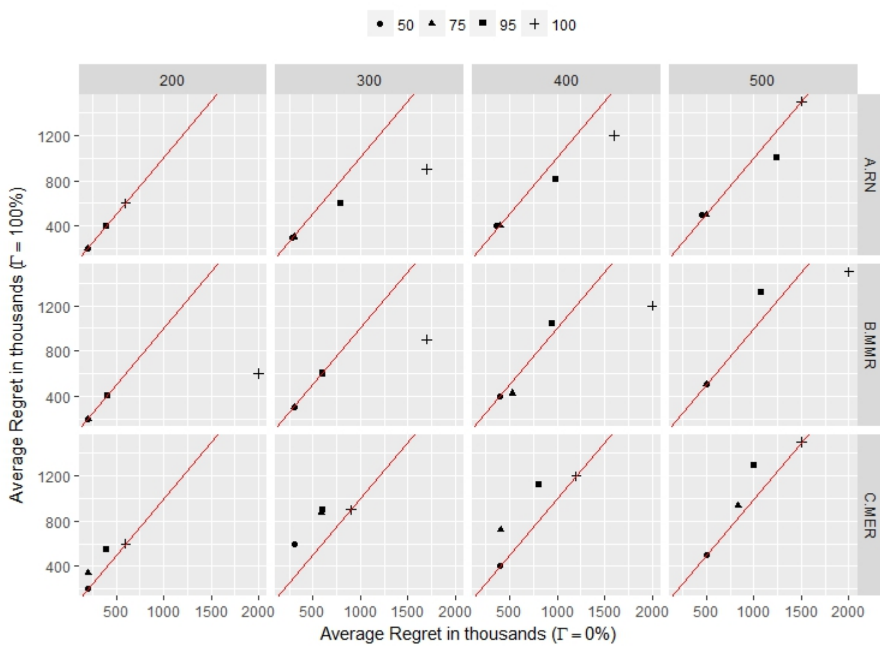


(b)

Figure 3.10. Cost and regret measures of nominal vs robust models with various incremental capacity unit



(a)



(b)

Figure 3.11. Cost and regret measures of nominal vs robust models with various construction costs

In Figure 3.10, we report the average 50th, 75th, 95th, and 100th percentile of cost (3.10a) and regret (3.10b) for each model as U_A varies. The X-axis and Y-axis represent average cost or regret for nominal models (mid point probabilities are used) and robust models (ambiguous probabilities are used with 100% budget), respectively. The red line represents $y = x$, so points below represent the robust model out-performing the nominal model, whereas points above the line represent the opposite. Figure 3.10a thus shows that the- worst case cost (100%) is significantly lower utilizing the solution of robust model **RN-PR** and **MMR-PR** than that of the nominal model **RN** and **MMR**. Figure 3.10b illustrates that most regrets over the 50th percentile are smaller when using solution to **RN-PR** and **MMR-PR** than when using solution to **RN** and **MMR**. When using solution to **MER** and **MER-PR** models, however, most points lie above the red line – in this case, cost and regret measures perform worse when robust models are employed. We observe similar tendencies when the construction cost C_A varies, in Figure 3.11.

The solution to model **RN-PR** tends to decrease the cost of high demand paths to minimize the worst-case expected cost, and thus, more capacity is built and fewer shortages are observed when the robust solution is used than when the nominal solution is used. **MMR-PR** and **MER-PR** both try to decrease the worst-case regret. We see that when the shortage cost is high relative to the investment cost, the penalty due to unmet demand on high demand paths leads to higher regret than the unused capacity on low demand paths.

When the shortage cost s is varied, as shown in Figure 3.12, similar observations can be made except when the shortage cost is very low ($s = 0.1$). In this case, in contrast to the previous cases, the unused capacity on low demand paths leads to higher regret than unmet demand on high demand paths, and as a result, **RN-PR** is not effective for minimizing regret.

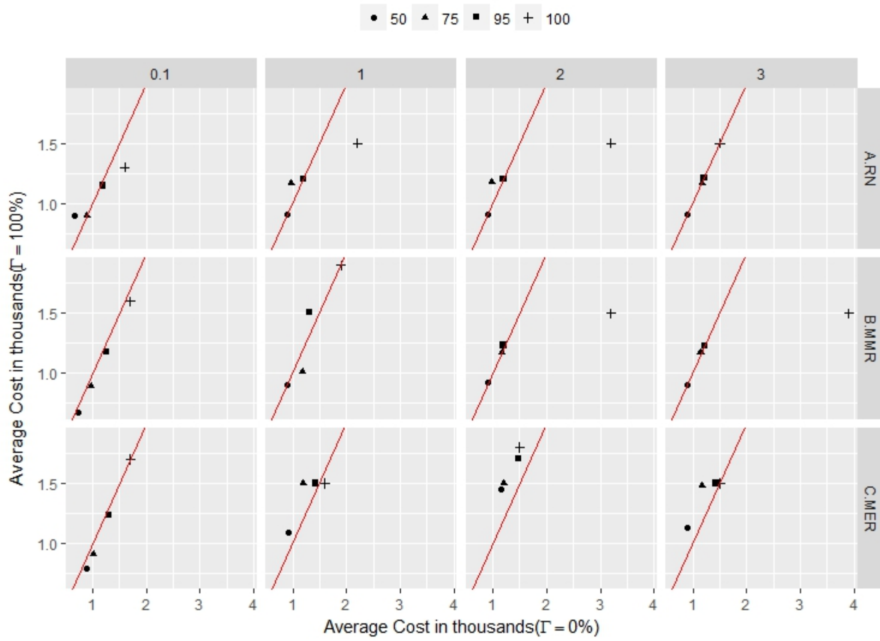
When the setup cost f_A has a positive value or an additional facility is introduced, the budget of uncertainty(Γ) plays a bigger role in the performance of solutions to these models, especially model **RN-PR**. We thus compare the solutions to robust model **RN-PR** with $\Gamma = 100\%$ and $\Gamma = 10\%$ with those to nominal model **RN** in Figure 3.13. To simplify exposition, we identify these robust models as **RN-PR**($\Gamma = 100\%$) and **RN-PR**($\Gamma = 10\%$).

When $f_A > 0$, as shown in Figure 3.13a and 3.13b, the worst-case cost and regret are smaller when using **RN-PR**($\Gamma = 10\%$) than when using **RN**, whereas 95% regret and mean excess regrets are larger when using **RN-PR**($\Gamma = 100\%$). Positive fixed cost increases the investment cost and encourages less frequent capacity construction, and thus may lead to more excess capacity on low demand paths while protecting the worst-case cost in the robust solution. The budget of uncertainty can prevent this overly conservative decision-making, and thus, the worst-case cost and regret perform better when the robust model is employed.

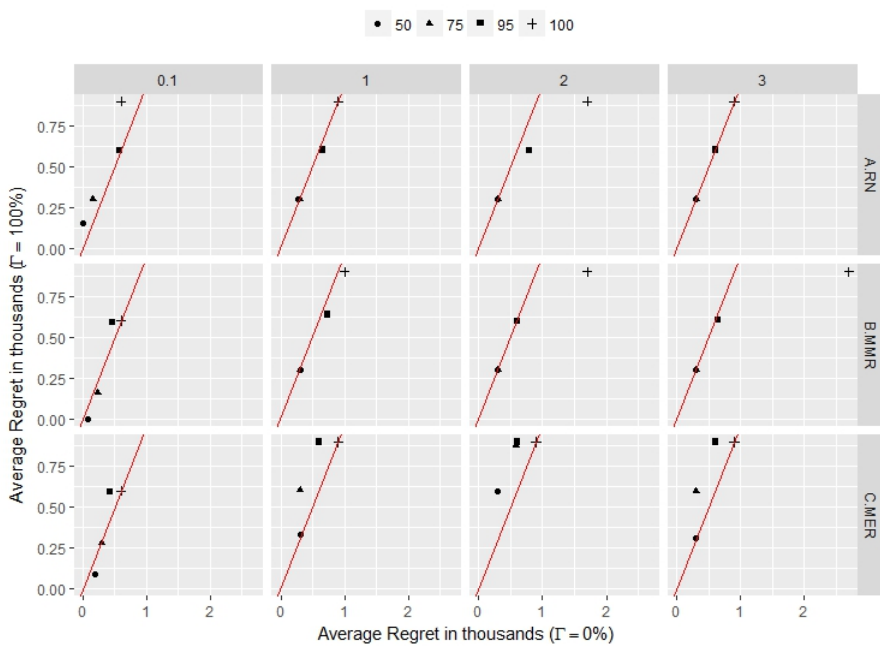
When an additional facility is introduced, all average cost measures are lower utilizing the solution to **RN-PR**($\Gamma = 10\%$) than when the solution to **RN** is employed, and some of cost measures are lower when the solution to **RN-PR**($\Gamma = 100\%$) is utilized. However regret measures are worse when using both **RN-PR**($\Gamma = 10\%$) and **RN-PR**($\Gamma = 100\%$) than when using **RN**.

When the shortage cost is very low, slightly more capacity is built with type A facilities (cheaper but with longer lead times) at the start of the planning horizon to decrease the cost of paths with high demand in the solution to **RN-PR**. This leads to wasted capacity in paths with low demand, so regret is worse, despite the decrease in construction costs.

Overall, we observe in our experiments that managers in this setting, with uncertainty around event probabilities, who are concerned about only minimizing the worst-case cost or worse case regret, should use the corresponding robust models. Managers concerned with simultaneously minimizing the worst-case cost, overall regret, and the worst-case regret



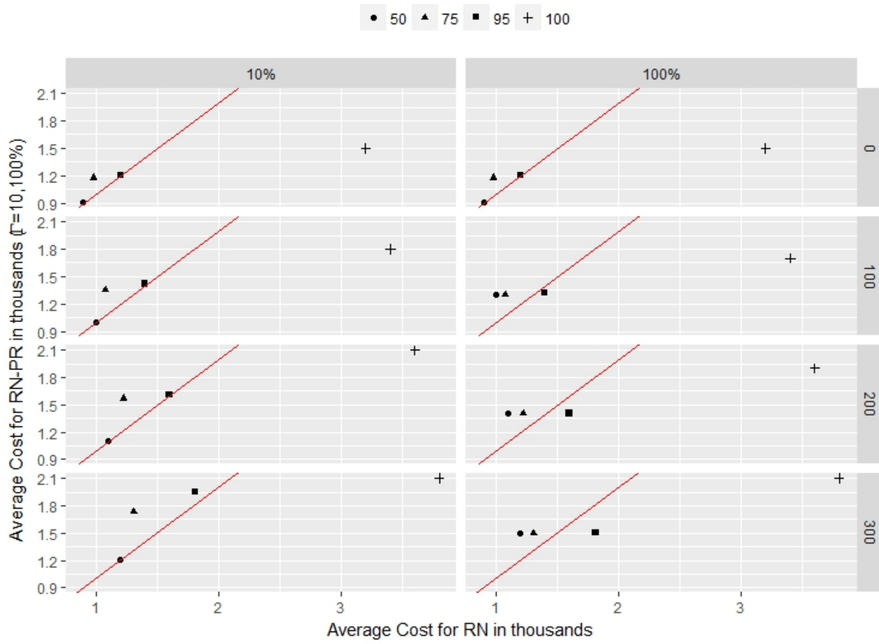
(a)



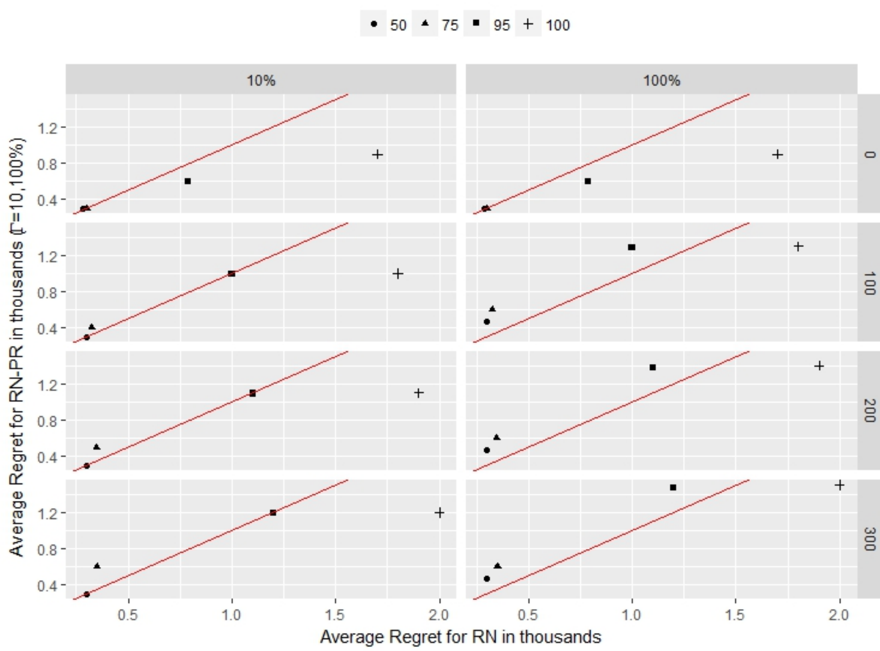
(b)

Figure 3.12. Cost and regret measures of nominal vs robust models with various shortage cost

should choose the robust model that minimizes expected cost, particularly when shortage cost is high relative to investment cost (in our experiments, for example, when the shortage cost per unit is at least double to the investment cost per unit). However, if minimizing expected overall cost is the primary concern, then it is better to choose the nominal model that minimizes expected cost. When shortage cost is low relative to investment cost, managers



(a)

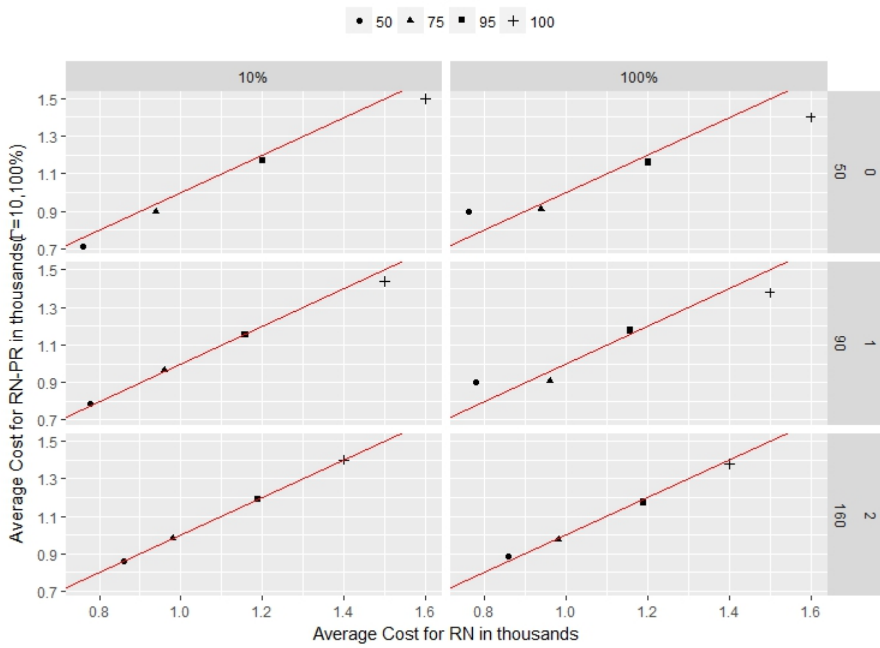


(b)

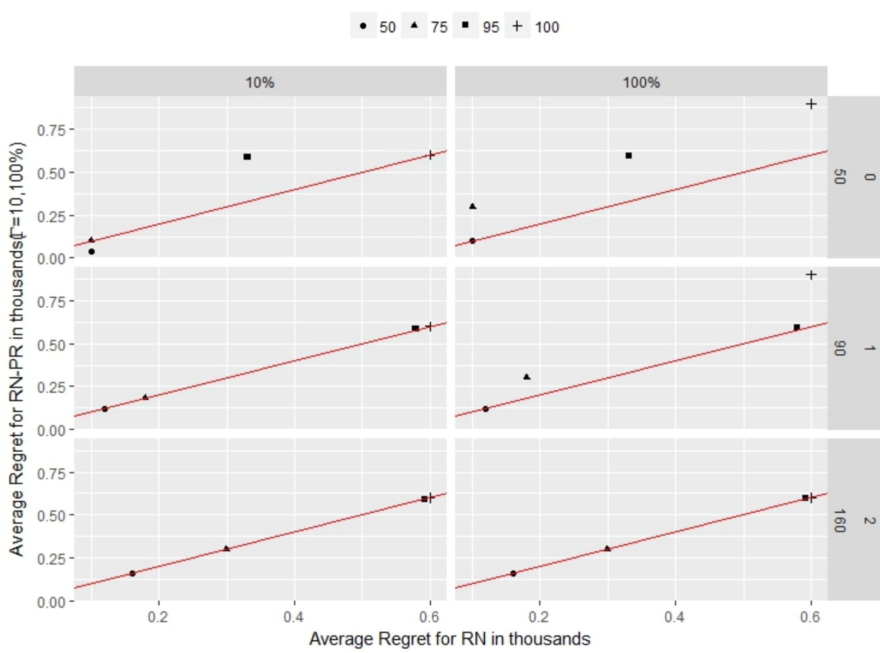
Figure 3.13. Cost and regret measures of **RN** vs **RN-PR** with various fixed cost

concerned with minimizing both overall and worst case regret should choose the nominal model that minimizes mean excess regret. Mangers concerned primarily with minimizing expected overall cost should choose the nominal model that minimizes expected cost.

In addition, when there are several types of capacity available, the robust solution with a correctly chosen budget of uncertainty reduces construction costs in many high demand



(a)



(b)

Figure 3.14. Cost and regret measures of **RN** vs **RN-PR** with various construction cost and lead time of additional facility

scenarios, but at the expense of higher regret. Thus, the robust model that minimizes expected cost will be useful if the firm is primarily concerned with minimizing expected overall cost.

Table 3.4. Average objective value difference to the optimal in percentages

Model	Objective		
	Expected Cost	95% Max Regret	95% Mean Excess Regret
RN	0.9	24.0	77.3
RN-PR	2.3	0	0.3
MMR	2.3	0	10.1
MMR-PR	3.3	0	0.3
MER	12.7	0	0.3
MER-PR	34.2	50.0	16.4

(a) $U_A = 2,000$, $c_A = 300$, s=2 (base case)

Model	Objective		
	Expected Cost	95% Max Regret	95% Mean Excess Regret
RN	0.6	46.8	15.7
RN-PR	10.4	48.2	29.2
MMR	3.5	14	9.4
MMR-PR	0.6	47	15.7
MER	10.8	7.4	2.6
MER-PR	5.6	48.0	15.8

(b) $U_A = 2,000$, $c_A = 300$, s=0.1

3.5.3 Impact of alternative objective functions

We also explore the impact of alternative objective functions on capacity expansion decisions. We choose two test instances- base case ($U_A = 2,000, c_A = 300, s = 2$) and the case with very low shortage cost ($U_A = 2,000, c_A = 300, s = 0.1$) - among various instances described in Section 3.5.2, since they can be two representative cases in terms of the investment and shortage cost.

In the following tests, we observe how each solution performs for each of the objective functions when the probability distribution is unknown.

In particular, for each set of test parameters, we solve six models ($m \in \{$**RN, RN-PR, MMR, MMR-PR, MER, MER-PR**$\}$) to optimality, and let X_m^* be the vector of optimal solution for model m. We simulate 100 sample scenario trees with resolved probability uncertainty as we described in Sub-Section 3.5.2. We denote the corresponding objective function of model m by $f(m) \in \{$*EC(Expected cost), MMR(95% Max regret), MER(95% Mean excess regret)*$\}$. For example, $f(\mathbf{RN})$ and $f(\mathbf{RN\text{-}PR})$, the corresponding objective functions of model **RN** and **RN-PR** are EC.

For each of sample scenario trees, denoted s, we solve each of the nominal models -**RN, MMR, MER**- to optimality, and define the optimal objective value of the model to be $z^*_{f(m),s}$. For each solution X_m^* of model $m \in \{$**RN, RN-PR, MMR, MMR-PR, MER, MER-PR**$\}$, given a simulated scenario tree s, we compute each of the objective function value $f(m')$ for $m' \in \{$ **RN, MMR, MER**$\}$ denoted $\mathcal{Z}_{f(m'),s}(X_m^*)$. We calculate the gap between $\mathcal{Z}_{f(m'),s}(X_m^*)$ and the optimal objective value, that is,

$$\frac{\mathcal{Z}_{f(m'),s}(X_m^*) - z^*_{f(m'),s}}{z^*_{f(m'),s}}$$

, and then average those over the 100 simulations runs.

In Table 3.4, we report average difference between computed objective function value and optimal objective function value of solutions of each model in percentages for the two test cases. We can compare the values in each row to observe how each solution performs for each objective function. We see that for the base case, the expected cost using the solutions to model **RN-PR, MMR-PR** performs only slightly worse (2.3% and 3.3%, respectively) than the optimal expected cost on average, while 95% max regret and 95% mean excess regret are close to optimal. However, when the shortage cost is low($s = 0.1$), performance is less consistent. As we discussed in the previous section, when the shortage cost is relatively higher than the investment cost, penalty due to unmet demand on high demand paths leads to high regret, and thus efforts to reducing cost on high demand paths are aligned for minimizing both the worst expected cost and regret.

Overall, the solutions to the robust model that minimizes expected cost perform well for all objectives with a slight loss in terms of optimal expected cost, particularly when the shortage cost is high relative high to the investment cost. Thus, in this case, if managers choose the robust model that minimizes expected cost, the decision can be robust to alternative objectives. When the shortage cost is low relative to the investment cost, choosing the most appropriate objective function becomes more critical.

3.6 Conclusion

In this chapter, we study a capacity planning model where demand is dependent on the outcome of a series of Bernoulli trials with unknown probabilities of success. We develop approaches to formulate and solve models that capture a variety of different objectives such as minimizing expected cost, minimizing α-reliable regret, and minimizing α-reliable mean excess regret. To find solutions that are robust to these ambiguities in probability of success, we formulate these models as robust stochastic integer programs and show that incorporating ambiguity in these probabilities does not increase the complexity of the problems. Our computational study suggests that when the shortage cost is relatively higher than the capacity investment cost, explicitly modeling ambiguous probabilities focusing on minimizing expected cost leads to solutions that actually perform quite well for all of the objective functions we consider in this chapter. For cases where these mixed integer linear programs are challenging to solve to optimality, we develop and test fast heuristic approaches. These include both a simple rolling horizon(SRH) approach, and a delayed event spike approach with rolling horizon (DERH). Through computational testing, we demonstrate that both SRH and DERH enable us to find good solutions to large problems that we are unable to solve effectively with commercial optimization software, and that DERH is computationally more expensive, but it outperforms SRH in terms of solution quality.

Chapter 4

Facility Location Planning

4.1 Introduction

In the capacity expansion planning model discussed in Chapter 3, we consider models optimizing timing, size and type of capacity investment – the building and expansion of facilities. In this chapter, we extend our capacity expansion models to optimize facility location and the allocation of customers to facilities, in addition to capacity expansion decisions. Thus, the problem we consider in this chapter is a version of the capacitated facility location problem, which is solved in a wide range of applications such as hub or supplier locations problems, and network design problems.

The facility location problems often involves uncertainties in input parameters such as demands, costs and among others. Our problem is related to models for facility location problems in the literature that include these characteristics. These uncertainties are often modeled using either stochastic programming or robust optimization, depending on whether probability distributions are known. In our setting, (stochastic) demand in each period is determined by the outcome of a series of binary events, where the likelihood of outcomes is unknown. We apply both robust optimization and stochastic programming to handle the uncertainty in this setting.

In many applications of the facility location problems, as mentioned above, there is a high degree of uncertainty surrounding future demands, costs, and other problem characteristics. To adapt to changing situations, firms may relocate facilities or expand the capacities of existing facilities, which may be expensive or inefficiently used. Sometimes, it may not be possible to relocate facilities – firms may be stuck with decisions that were made many years ago considering an entirely different set of circumstances. For example, power plants and hospital are often built anticipating many years of operation(Torres Soto, 2009). Therefore, it is often critical that firms properly account for future uncertainty in their decision-making processes in order to avoid expensive and inefficient investments.

As in Chapter 3, our focus here is on a class of discrete time planning models where demand in each period is determined by the outcome of a series of "binary events" in that period and in previous periods. In many ways, the setting in this chapter is similar: the outcome of events is uncertain, and the probability of outcomes cannot effectively be estimated. In contrast to Chapter 3, however, in this chapter, demand uncertainty is not completely resolved after binary event outcome. For example, in the biopharmaceutical industry, although demand is quite often well-understood given a series of binary events, this is not always true. There may still be a good deal of uncertainty once trial results are realized.

Our goal is to find solutions that are robust to the ambiguities described above. We formulate the models in this chapter as (stochastic) robust integer programs, and because these multistage stochastic programming problems are challenging to solve to optimality as the problem size becomes large, we propose heuristic approaches that extend the concepts of Nested Decomposition (ND) and Stochastic Dual Dynamic Programming (SDDP) to our setting in an effort to solve these models. These algorithms formulate the problem as a dynamic program (DP) and approximate the convex cost-to-go functions by adding Benders' cuts.

Since there are ambiguities in probabilities and demands, we modify standard ND and SDDP to work in a robust setting. In addition, to speed by convergence of those approaches, we consider splitting a large problem into moderate-sized problems rather than small-sized problems (problems with a single period). This concept can reduce the number of DP equations to iterate, and may reduce the number of iteration to converge although time to solve each sub-problem is increased. We call these approaches Blocked Nested Decomposition (BND) and Blocked Stochastic Dual Dynamic Programming (BSDDP).

Specifically, we consider the problem of selecting the locations of facilities, the capacity of these facilities, and the amounts shipped from these facilities to customers over a discrete and finite time horizon. The decision to open facilities is made once at the beginning of the time horizon, and for those opened facilities, the amount of available capacity can be expanded in any period. In each period, the amount of the product that can be shipped to retailers from a facility is constrained by available capacity of the facility. We optimize a function of sales revenue, product cost, facility construction costs, capacity expansion costs, and shipping costs. As in Chapter 3, we use a scenario tree to describe the uncertainty of demand which evolves based on trial outcomes.

For unresolved uncertainty of demand after the series of binary event outcomes, we may assume that we can sample using the Monte Carlo sampling approach given the true distribution of demand. However, the true distribution of demand is often unknown. Even if we assume that the true distribution is known, adding sampled realizations to the scenario tree increases the size of scenario tree significantly. The scenario tree that describes demand uncertainty based on binary test outcomes includes $\sum_{t=0}^{T} 2^t$ nodes in total. If, for example, 5 realizations are sampled for each node, the total number of nodes in the scenario tree becomes $\sum_{t=0}^{T} 10^t$. As an alternative approach, we employ robust optimization to capture remaining ambiguities in demand after the binary event outcomes. We also incorporate ambiguities in probabilities using robust optimization, which does not increase computational complexity significantly.

The size of the problem increases as the number of scenarios, the number of candidate facility sites, and number of demand sites increase. The event spike approach, which we used in the previous chapter, primarily focuses on reducing the number of the scenarios. Motivated by Yu et al. (2018), who introduce the use of SDDP for a multi-period stochastic facility location problem, we adapt concepts from ND and SDDP algorithms to develop algorithms that solve our problem efficiently.

The remainder of this chapter is organized as follows.

In Section 4.2, we formulate our facility location models, where the probability of Bernoulli trial success, is known, using a multistage stochastic mixed integer program. Then, we extend this model to incorporate ambiguous probabilities and demand. In Section 4.3, we develop the ND and SDDP-based approaches for efficiently solving these models. In Section 4.4, we present our preliminary computational testing, where we explore the effectiveness of the robust approach, and the performance of the algorithms we have developed.

4.2 Model

We consider a discrete time multi-period facility location problem with T-period time horizon, a set of potential facility sites, I, and a set of demand locations, J. We begin by introducing our initial model, which assumes that the probability of each trial success is known, and demand uncertainty is completely resolved after the binary event outcome is revealed. As described above, given this scenario, demand uncertainty can be modeled as a scenario tree, where each node tm in stage t and node counter m of the tree represents a realization of demand, and edges are labeled with the probability of trial outcomes. Each node of the tree from $t = 1$ to T has a unique parent node $A(tm)$, and each non-leaf node has a set of child nodes, denoted $C(tm)$. We denote the probability on the edge from $A(tm)$ to tm by π_{tm} – the conditional probability to reach node tm given $A(tm)$. The probability that scenario s is realized, θ^s, equals $\prod_{n \in \mathscr{P}(s)} \pi_n$, where $\mathscr{P}(s)$ is the set of nodes on the path of scenario s.

We denote by r^t the unit price of a product at time t, by f_i the fixed cost opening a facility at location i, by c_i^t the cost of expanding capacity by one unit at facility in location i at time t, by v_{ij}^t the cost of shipping one unit from facility i to customer j at time t, by U_i the unit capacity of facility i, and by d_j^{tm} the demand of customer j at node tm.

We define decision variables $x_i \in \{0, 1\}$ such that $x_i = 1$ if we open a facility at location i, and $x_i = 0$ otherwise. We also define decision variables y_i^{tm} to be the amount of capacity added to facility i at node tm, and variables z_{ij}^{tm} to equal the amount of product shipped from facility i to customer j at node tm.

We summarize notation used to formulate the model as follows.

Sets and indices:

- S: Set of scenarios ($s \in S$)
- I: Set of potential facility locations ($i \in I$)
- J: Set of customers ($j \in J$)
- N: Set of nodes ($tm \in N$ or $n \in N$)
- t: Index for time periods $(0, ..., T)$
- m: Index for nodes at time t, $(1, .., 2^t)$

Parameters:

- r^t: Unit price of a product at time t
- U_i: Unit capacity of a facility i

- f_i: Fixed cost opening a facility i
- c_i^t: Cost per unit of capacity adding to a facility at site i at time t
- v_{ij}^t: Shipping cost per unit from facility at site i to customer j at time t
- d_j^{tm}: Demand of customer j at time t at node tm
- π_{tm}: Conditional probability of reaching node tm
- θ^s: Probability of realization of a scenario s ($\theta^s = \prod_{n \in \mathcal{P}(s)} \pi_n$)

Decision Variables:

- x_i: 1 if opening a facility i, 0 otherwise
- y_i^{tm}: Amount of capacity added to facility i at time t at node tm
- z_{ij}^{tm}: Amount of products shipped from facility i to customer j at time t at node tm
- $F(tm)$: Profit at node tm
- $P(s)$: Total profit for scenario s

Given this notation, we formulate the capacitated facility location problem with the goal of maximizing the expected profit as follows. We present a node-based formulation to avoid the large number of non-anticipativity constraints.

FL

$$max \quad \sum_{s \in S} \theta^s P(s) \tag{4.1}$$

$$s.t. \quad F(01) = -\sum_{i \in I}(f_i x_i + c_i^0 y_i^{01}) + \sum_{i \in I}\sum_{j \in J}(r^0 - v_{ij}^t) z_{ij}^{01} \tag{4.2}$$

$$F(tm) = -\sum_{i \in I} c_i^t y_i^{tm} + \sum_{i \in I}\sum_{j \in J}(r^t - v_{ij}^t) z_{ij}^{tm} \quad \forall tm \in N, t \geq 1 \tag{4.3}$$

$$P(s) = \sum_{tm \in \mathscr{P}(s)} F(tm) \quad s \in S \tag{4.4}$$

$$\sum_{i \in I} z_{ij}^{tm} \leq d_j^{tm} \quad j \in J, tm \in N \tag{4.5}$$

$$y_i^{tm} \leq M x_i \quad i \in I, tm \in N \tag{4.6}$$

$$\sum_{j \in R} z_{ij}^{tm} \leq U_i \sum_{n \in \mathcal{P}(tm)} y_i^n \quad i \in I, tm \in N \tag{4.7}$$

$$x_i \in \{0, 1\} \quad i \in I \tag{4.8}$$

$$y_i^{tm} \geq 0, z_{ij}^{tm} \geq 0 \quad i \in I, j \in J, tm \in N \tag{4.9}$$

The objective function (4.1) maximizes the expected profit. Constraint (4.2) defines the profit of the root node (the beginning of planning horizon). The first term expresses the fixed facility opening and capacity building cost; the second term, revenue less shipping cost. Constraint (4.3) defines the profit of each node when $t \geq 1$. The first term expresses the capacity adding cost; the second term, revenue less shipping cost. Constraint (4.5) implies that amount fulfilled for customer j at a node is at most its demand quantity. Constraint(4.6) states that capacity is only added at open facilities. Constraint (4.7) ensures that the demand satisfied by facility i at node tm is less than its capacity. All remaining constraints are the non-negativity and binary constraints.

In the following two sections, we extend the model **FL** in order to model ambiguity in the probability of success of each Bernoulli trial and ambiguity in demand after the trial outcome is realized. As we point out in Chapter 3, although it may be possible to estimate these probabilities and demand using historical data, the model **FL** is sensitive to small changes in probability and demand estimates.

This leads us to consider a robust optimization(RO) approach, similar to the approach we applied in Chapter 3. Recall that in this approach, each ambiguous parameter lies in an interval and the number of parameters which can take their extreme values is limited by a budget.

We first describe the uncertainty sets for ambiguous probabilities and demand, and then formulate the robust version of **FL** that incorporates ambiguities in probability and demand.

4.2.1 Uncertainty Sets for Ambiguous Parameters

We begin by describing the uncertainty sets for the ambiguous parameters. Let $\tilde{\pi}_{tm}$ be the ambiguous probability, which belongs to an interval centered at $\bar{\pi}_{tm}$ and of half length $\hat{\pi}_{tm}$. Without a budget of uncertainty, the uncertainty set Ω for ambiguous parameter $\tilde{\pi}_{tm}$ is:

$$\Omega = \{\tilde{\pi} | \tilde{\pi}_{tm} \in [\bar{\pi}_{tm} - \hat{\pi}_{tm}, \bar{\pi}_{tm} + \hat{\pi}_{tm}] \quad \forall tm \in \cup_{t=1..T} N^t,$$
$$\sum_{n \in C(tm)} \tilde{\pi}_n = 1, \forall tm \in \cup_{t=0..T-1} N^t\}.$$

In general, considering all possible values of probabilities is too conservative, and so as in the previous chapter, we adjust the level of conservativeness using a budget of uncertainty as follows. We define the positive and negative scaled deviation of $\tilde{\theta}^s$ from its nominal value $\theta^s_{mid} = \prod_{tm \in \mathcal{P}(s)} \bar{\pi}_{tm}$ as

$$z^s_+ = (\tilde{\theta}^s - \theta^s_{mid})/\hat{\theta}^s_+$$

and

$$z^s_- = (\tilde{\theta}^s - \theta^s_{mid})/\hat{\theta}^s_-$$

where $\hat{\theta}^s_+ = \prod_{i \in P(s)} \pi^+_{tm} - \theta^s_{mid}$ and $\hat{\theta}^s_- = \theta^s_{mid} - \prod_{i \in P(s)} \pi^-_{tm}$, respectively. The scaled deviations take values in $[0, 1]$. Then, we force a budget of uncertainty as follows:
The total variation of scenario realization probabilities cannot exceed some threshold Γ, not necessarily integer:

$$\sum_{s \in S} z^s_+ + z^s_- \leq \Gamma$$

We transform Ω for $\tilde{\pi}_{tm}$ to the uncertainty set Θ for $\tilde{\theta}^s$ in order to remove nonlinear term $\prod_{i \in P(s)} \tilde{\pi}_i$, and we add the budget of uncertainty described above. Note that $\tilde{\pi}_{tm}$ can be represented as the sum of probabilities of scenario realizations that include node tm divided by the sum of probabilities of realized scenarios that include the parent of node tm, i.e., $\sum_{i\in S(tm)}\tilde{\theta}^i/\sum_{i\in S(A(tm))}\tilde{\theta}^i$. Then

$$\pi^-_{tm} \le \tilde{\pi}_{tm} \le \pi^+_{tm} \Leftrightarrow \pi^-_{tm}\sum_{i\in S(A(tm))}\tilde{\theta}^i \le \sum_{i\in S(tm)}\tilde{\theta}^i \le \pi^+_{tm}\sum_{i\in S(A(tm))}\tilde{\theta}^i$$

Also, the sum of scenario probabilities must satisfy $\sum_{s\in S}\tilde{\theta}^s = 1$. Under this transformation, the probability constraints $\sum_{i\in C(tm)}\tilde{\pi}_i = 1$ become trivial, as we show below:

$$\sum_{i\in C(tm)}\tilde{\pi}_i = 1 \Leftrightarrow \sum_{i\in C(tm)}\sum_{k\in S(i)}\tilde{\theta}^k/\sum_{j\in S(tm)}\tilde{\theta}^j = 1 \Leftrightarrow \sum_{j\in S(tm)}\tilde{\theta}^j/\sum_{j\in S(tm)}\tilde{\theta}^j = 1 \Leftrightarrow 1 = 1$$

Therefore, the uncertainty set Θ for $\tilde{\theta}^s$ is:

$$\begin{aligned}\Theta = \Big\{\tilde{\theta} \in R^{|S|} | \tilde{\theta}^s = \theta^s_{mid} - \hat{\theta}^s_- z^s_- + \hat{\theta}^s_+ z^s_+,& \\ & 0 \le z^s_+ \le 1, 0 \le z^s_- \le 1, \quad \forall s \in S, \\ & \pi^-_{tm}\sum_{i\in S(A(tm))}\tilde{\theta}^i \le \sum_{i\in S(n)}\tilde{\theta}^i \le \pi^+_{tm}\sum_{i\in S(A(tm))}\tilde{\theta}^i, \quad \forall tm \in N, t \ge 1 \\ & \sum_{s\in S}\tilde{\theta}^s = 1, \quad \sum_{s\in S} z^s_+ + z^s_- \le \Gamma\Big\}\end{aligned}$$

We also describe the uncertainty set for demand at each node tm, $\tilde{d}^{tm}_j$ as follows. We let $\tilde{d}^{tm}_j$ take values in $[\bar{d}^{tm}_j - \hat{d}^{tm}_j, \bar{d}^{tm}_j + \hat{d}^{tm}_j]$. The total variation of the scaled deviations of demand at node tm cannot exceed some threshold Γ^D_{tm}:

$$\sum_{j\in J}\frac{|\tilde{d}^{tm}_j - \bar{d}^{tm}_j|}{\hat{d}^{tm}_j} \le \Gamma^D_{tm}$$

Then, the uncertainty set $\mathscr{D}$ for $\tilde{d}^{tm}_j$ is:

$$\begin{aligned}\mathscr{D} = \Big\{\tilde{\mathbf{d}} \in \mathbb{R}^{I\times N\times T} | \tilde{d}^{tm}_j \in [\bar{d}^{tm}_j - \hat{d}^{tm}_j, \bar{d}^{tm}_j + \hat{d}^{tm}_j] \quad \forall j \in J, tm \in N,& \\ \sum_{j\in J}\frac{|\tilde{d}^{tm}_j - \bar{d}^{tm}_j|}{\hat{d}^{tm}_j} \le \Gamma^D_{tm} \quad \forall tm \in N\Big\}&\end{aligned}$$

4.2.2 Models using Robust Optimization

Given uncertainty sets Θ and $\mathscr{D}$, we formulate the robust version of **FL** as follows.

FL-R

$$max \min_{\theta \in \Theta} \sum_{s \in S} \tilde{\theta}^s P(s) \tag{4.10}$$

$$s.t. \quad (4.2) - (4.4), (4.6) - (4.9)$$

$$\sum_{i \in I} z_{ij}^{tm} \leq \tilde{d}_j^{tm} \quad j \in J, tm \in N, \tilde{d}_j^{tm} \in \mathscr{D} \tag{4.11}$$

Next, we derive the Mixed Integer Linear Program (MILP) for the uncertain MILP model **FL-R**. Given uncertainty set Ω, the objective function (4.10) which includes the ambiguous parameter $\tilde{\theta}^s$, is equivalent to solving the following auxiliary problem with variables z_+^s and z_-^s.

$$max \quad -\sum_{s \in S} (\theta_{mid}^s + \hat{\theta}_+^s z_+^s - \hat{\theta}_-^s z_-^s) P(s)$$

$$s.t. \quad \tilde{\theta}^s = \theta_{mid}^s + \hat{\theta}_+^s z_+^s - \hat{\theta}_-^s z_-^s \tag{4.12}$$

$$\sum_{s \in S} \tilde{\theta}^s = 1 \tag{4.13}$$

$$\pi_n^- \sum_{i \in S(A(n))} \tilde{\theta}^i \leq \sum_{i \in S(n)} \tilde{\theta}^i \qquad n \in N \setminus \{0\} \tag{4.14}$$

$$\sum_{i \in S(n)} \tilde{\theta}^i \leq \pi_n^+ \sum_{i \in S(A(n))} \tilde{\theta}^i \qquad n \in N \setminus \{0\} \tag{4.15}$$

$$\sum_{s \in S} z_+^s + z_-^s \leq \Gamma \tag{4.16}$$

$$0 \leq z_+^s \leq 1 \qquad s \in S \tag{4.17}$$

$$0 \leq z_-^s \leq 1 \qquad s \in S \tag{4.18}$$

We introduce dual variables $q, \omega_-^n, \omega_+^n, t, \gamma_+^s, \gamma_-^s$ for constraints (4.13), (4.14), (4.15), (4.16), (4.17) and (4.18). Then, the dual problem becomes:

$$\begin{aligned}
min \quad & C' \\
s.t. \quad & K^+(s) \geq -P(s)\hat{\theta}_+^s \quad s \in S \\
& K^-(s) \geq P(s)\hat{\theta}_-^s \quad s \in S \\
& t \geq 0, \\
& \gamma_+^s \geq 0, \gamma_-^s \geq 0, \quad s \in S \\
& \omega_+^n \geq 0, \omega_-^n \geq 0 \quad n \in N \setminus \{01\}
\end{aligned}$$

where

$$
\begin{aligned}
C' =& t\Gamma + \sum_{s\in S}(\gamma_+^s + \gamma_-^s) + \sum_{n\in C(0)} \hat{\pi}_n(\omega_+^n + \omega_-^n) + \sum_{n\in N^-} \hat{\pi}_n \sum_{i\in S(A(n))} \theta_{mid}^i(\omega_+^n + \omega_-^n) \\
K^+(s) =& q\hat{\theta}_+^s + t + r_+^s + Ka(\hat{\theta}_+^s) + Kb(\hat{\theta}_+^s) + Kc(\hat{\theta}_+^s) \\
K^-(s) =& -q\hat{\theta}_-^s + t + r_-^s + Ka(-\hat{\theta}_-^s) + Kb(-\hat{\theta}_-^s) + Kc(-\hat{\theta}_-^s) \\
Ka(a) =& \sum_{\substack{m\in P(s)\\ A(m)=0}} a(\omega_+^m - \omega_-^m), \quad Kb(b) = \sum_{\substack{n\in P(s)\\ A(n)\neq 0}} b\{(1-\pi_n^+)\omega_+^n - (1-\pi_n^-)\omega_-^n\} \\
Kc(c) =& \sum_{\substack{A(n)\in P(s)\\ n\notin P(s)}} \{c(\pi_n^-\omega_-^n - \pi_n^+\omega_+^n)\}
\end{aligned}
$$

The constraint (4.10) includes the ambiguous parameter $\tilde{d}_j^{tm}$, and it is equivalent to the following inequality:

$$\sum_{i\in I} z_{ij}^{tm} \leq \bar{d}_j^{tm} - \hat{d}_j^{tm} * \min(1, \Gamma_{tm}^D)$$

With this observation, **FL-R** can be reformulated as follows:

FL-RC

$$max \quad \sum_{s\in S} \theta_{mid}^s P(s) - C' \tag{4.19}$$

$$s.t. \quad (4.2) - (4.4), (4.6) - (4.9)$$

$$\sum_{i\in I} z_{ij}^{tm} \leq \bar{d}_j^{tm} - \hat{d}_j^{tm} * \min(1, \Gamma_{tm}) \quad j \in J, tm \in N \tag{4.20}$$

$$K^+(s) \geq -P(s)\hat{\theta}_+^s \quad s \in S \tag{4.21}$$

$$K^-(s) \geq P(s)\hat{\theta}_-^s \quad s \in S \tag{4.22}$$

$$t \geq 0, \gamma_+^s \geq 0, \gamma_-^s \geq 0, \omega_+^n \geq 0, \omega_-^n \geq 0 \tag{4.23}$$

We can make various choices when modeling ambiguities in probabilities and demands. We may estimate the values for both parameters and use the nominal model. We may estimate the one parameter, and incorporate ambiguities in another parameter or incorporate ambiguities in both parameters. We may consider a model, where demand is not based on binary outcomes.

In the following example, we can see that decisions vary depending on modeling choices. We also observe that when including the setting where demand depends on binary outcomes in the robust models, the solutions to the models are less conservative.

Example 2. *As an example, consider a problem with $T = 1$ and two customers, $j = 1, 2$. For each j, demand at each node is $\tilde{d}_j^{01} = 0$, $\tilde{d}_j^{11} \in [8000, 12000]$ and $\tilde{d}_j^{12} \in [1000, 5000]$. The locations of customers are the candidate locations of the two facilities. The open facility will cover demand, if possible, with $r^t = 1, f_i = 2,000, c_{i0} = 0.1, c_i^1 = 0.3$ for all i. The*

Table 4.1. Solutions under different modeling choices for Example 2

	FL	RC(P)	RC(D)	RD(PD)	RO
O.V.	7,000	4,900	3,400	1,580	0
t=0	(10,000,10,000)	(5,000,5,000)	(8,000,8,000)	(1,000,1,000)	(0,0)
t=1	(0,0)	(5,000,5,000)	(0,0)	(7,000,7,000)	(0,0)
	(0,0)	(0,0)	(0,0)	(0,0)	

*transportation cost is $v_{ij}^t = 1$ if $i \neq j$, and $v_{ij}^t = 0$ otherwise. We assume a box uncertainty set, where $\Gamma_{tm}^D = 2$. The probability of success $\tilde{p}_{11}$ falls in the interval $[0.3, 0.7]$. We compare solutions to the nominal model **FL** and the robust model **FL-RC**. For **FL**, we use the mid-point values of intervals for all uncertain parameters. For **FL-RC**, we consider ambiguities in probability only, demand only, and both probability and demand, and we denote these by RC(P), RC(D), and RC(PD), respectively. We also consider a model where demand is not based on binary outcomes, RO, where for each j, demand is defined as $\tilde{d}_j^0 = 0$ and $\tilde{d}_j^1 \in [1000, 12000]$. In Table 4.1, we present the expected profit and capacity built at each period and facility in the solutions to models described above. As shown in Table 4.1, solutions vary from model to model. Note that if we ignore the setting where demand depends on binary outcomes, the solution becomes very conservative resulting in not building any facilities.*

In many location-allocation models, shipment quantity is often presented as a fraction of total demand. In our model, we can substitute $\zeta_{ij}^{tm}\tilde{d}_j^{tm}$ for z_{ij}^{tm} where ζ_{ij}^{tm} is the proportion of demand j at node tm. The optimal solution remains same in nominal model, but this is not true for robust model. Specifically, Ardestani-Jaafari and Delage (2018), argue that this may lead to over-conservative solution when we consider ambiguities in demand, and the following example supports this observation. In Example 3, we use the same problem instance described in Example 2, and focus on ambiguities in demand. We compare the solution to **FL-RC** with the solution to a version of the model where we replace z_{ij}^{tm} by $\zeta_{ij}^{tm}\tilde{d}_j^{tm}$. We call this model FL-FRV.

Example 3. *The optimal objective value of **FL-RC** is $3,400$ as shown the column of RC(D) in Table 4.1. However, when we use the fractional variable, the optimal objective value of FL-FRV is $2,600$. In the optimal solution to FL-FRV $12,000$ units of capacity for each facility is built at time 0 to satisfy the largest possible demand, because the following constraint needs to be satisfied for all possible demand:*

$$\sum_{j \in R} \zeta_{ij}^{tm}\tilde{d}_j^{tm} \leq U_i \sum_{n \in \mathcal{P}(tm)} y_i^n \tag{4.24}$$

However, in the following objective function, the worst-case profit is computed, where we account for the smallest possible demand.

$$\begin{aligned} maxmin_{d \in \mathscr{D}} \quad & \sum_{s \in S} \theta^s((-\sum_{i \in I}(f_i x_i + c_i^0 y_i^{01}) + \sum_{i \in I}\sum_{j \in J}(r^0 - v_{ij}^t)\sum_{j \in R}\zeta_{ij}^{01}\tilde{d}_j^{01}) \\ & + (\sum_{tm \in \mathscr{P}(s)} - \sum_{i \in I} c_i^t y_i^{tm} + \sum_{i \in I}\sum_{j \in J}(r^t - v_{ij}^t)\sum_{j \in R}\zeta_{ij}^{tm}\tilde{d}_j^{tm})) \end{aligned}$$

This leads $4,000$ *units of capacity in each facility is not used in the optimal solution to FL-FVB of this example, and its worst-case profit is* 800 *less than the worst-case profit of* ***FL-RC****. In the following, we show the case that this possibility to waste large capacity may lead to overly conservative decision when using FL-FRV.*

Suppose we let $f_i = 3,000$*,* $\tilde{d}_j^{11} \in [8,000, 15,000]$ *and keep other parameters unchanged. The optimal objective value of* ***FL-RC*** *is equal to* 600*, but the optimal objective value of FL-FRV is equal to* 0*. The solution to FL-FRV becomes very conservative resulting in not building any facilities, where we may lost an opportunity to make a profit.*

Ben-Tal et al. (2004) introduce the adjustable robust optimization (ARO) methodology, where decision variables can have one component that cannot be adjusted due to the uncertain data, and another component that can be adjusted based on the uncertain data. They point out that introducing these variables often makes solutions less conservative at the cost of additional computational time. In FL-FRV, z_{ij}^{tm} is a decision variable that can be written in terms of ambiguous demand, and adapting a concept from ARO, we use the following substitution:

$$z_{ij}^{tm} = w_{ij}^{tm} + \zeta_{ij}^{tm}\tilde{d}_j^{tm}$$

This substitution guarantees that the following revised model always provides a solution that is at most as conservative as the solution to **FL-RC**. In other words, the objective value of the revised model is always higher than the objective value of **FL-RC**.

Next, we formulate the robust version of **FL** with the two additional variables w_{ij}^{tm} and ζ_{ij}^{tm} as follows:

FLFV-R

$$max \min_{\theta\in\Theta} \sum_{s\in S} \tilde{\theta}^s P(s) \tag{4.25}$$

$$s.t. \quad F(01) = -\sum_{i\in I}(f_i x_i + c_i^0 y_i^{01}) + \sum_{i\in I}\sum_{j\in J}(r - v_{ij}^t)(w_{ij}^{01} + \zeta_{ij}^{tm}\tilde{d}_j^{01}) \tag{4.26}$$

$$F(tm) = -\sum_{i\in I} c_i^t y_i^{tm} + \sum_{i\in I}\sum_{j\in J}(r - v_{ij}^t)(w_{ij}^{tm} + \zeta_{ij}^{tm}\tilde{d}_j^{tm}) \quad \forall tm \in N, t \geq 1 \tag{4.27}$$

$$P(s) = \sum_{tm\in\mathcal{P}(s)} F(tm) \quad s \in S \tag{4.28}$$

$$\sum_{i\in I} w_{ij}^{tm} + \zeta_{ij}^{tm}\tilde{d}_j^{tm} \leq \tilde{d}_j^{tm} \quad j \in J, tm \in N, d \in \mathscr{D} \tag{4.29}$$

$$y_i^{tm} \leq M x_i \quad i \in I, tm \in N \tag{4.30}$$

$$\sum_{j\in R}(w_{ij}^{tm} + \zeta_{ij}^{tm}\tilde{d}_j^{tm}) \leq U_i \sum_{n\in\mathcal{P}(tm)} y_i^n \quad i \in I, tm \in N \tag{4.31}$$

$$x_i \in \{0, 1\} \quad i \in I \tag{4.32}$$

$$y_i^{tm} \geq 0, w_{ij}^{tm} \geq 0, \zeta_{ij}^{tm} \geq 0 \quad i \in I, j \in J, tm \in N \tag{4.33}$$

Lemma 3. ***FLFV-R*** *can be formulated as a MILP.*

Proof. We note that (4.26)- (4.27) are equality constraints with ambiguous parameters. Thus, we can rewrite objective function (4.25) and constraints (4.26)- (4.28) as follows:

$$\begin{aligned} & max \min_{\theta\in\Theta} \sum_{s\in S} \tilde{\theta}^s P(s) && (4.34) \\ = & max \min_{\theta\in\Theta} \sum_{s\in S} \tilde{\theta}^s \sum_{tm\in\mathcal{P}(s)} F(tm) \\ = & max \min_{\theta\in\Theta} \sum_{s\in S} \tilde{\theta}^s \sum_{tm\in\mathcal{P}(s)} \max_{d\in\mathscr{D}} F(tm) \end{aligned}$$

where

$$F(tm) = -\sum_{i\in I} c_i^t y_i^{tm} + \sum_{i\in I}\sum_{j\in J}(r - v_{ij}^t)(w_{ij}^{tm} + \zeta_{ij}^{tm}(\bar{d}_j^{tm} + \hat{d}_j^{tm} e_j^{+tm} - \hat{d}_j^{tm} e_j^{-tm}))$$

Then, $\max_{d\in\mathscr{D}} F(tm)$ is equivalent to solving the following auxiliary linear program:

$$\begin{aligned} & max \sum_{i\in I}\sum_{j\in J}(r - v_{ij}^t)\zeta_{ij}^{tm}\hat{d}_j^{tm} e_j^{tm} \\ & s.t \quad 0 \le e_j^{tm} \le 1 \quad j\in J && (4.35) \\ & \sum_{j\in J} e_j^{tm} \le \Gamma_{tm}^D && (4.36) \end{aligned}$$

We can apply strong duality, since the feasible set is nonempty and bounded. Let λ_{jtm} and μ_{tm} be dual variables for constraints (4.35) and (4.36), respectively. Then, the dual problem becomes:

$$\begin{aligned} min \quad & \sum_{j\in J} \lambda_{jtm} + \Gamma_{tm}^D \mu^{tm} \\ s.t. \quad & \lambda_{jtm} + \mu_{tm} \ge \sum_{i\in I} \hat{d}_j^{tm} \zeta_{ij}^{tm}(r - v_{ij}^t) \\ & \lambda_{jtm} \ge 0, \mu_{tm} \ge 0 \end{aligned}$$

Now, $\max_{d\in\mathscr{D}} F(tm)$ can be rewritten as:

$$F'(tm) = -\sum_{i\in I} c_i^t y_i^{tm} + \sum_{i\in I}\sum_{j\in J}(r - v_{ij}^t)(w_{ij}^{tm} + \zeta_{ij}^{tm}\bar{d}_j^{tm}) + \sum_{j\in J}\lambda_{jtm} + \Gamma_{tm}^D \mu^{tm}.$$

Objective (4.34) now can be transformed to

$$max \min_{\theta\in\Theta} \sum_{s\in S} \tilde{\theta}^s \sum_{\mathcal{P}(s)} F'(tm) \tag{4.37}$$

Let $P'(s) = \sum_{\mathcal{P}(s)} F'(tm)$, and the objective (4.37) is

$$max \min_{\theta\in\Theta} \sum_{s\in S} \tilde{\theta}^s P'(s) \tag{4.38}$$

. As we have seen in **FL-RC**, this objective function can be rewritten as follows.

$$
\begin{aligned}
max \quad & \sum_{s \in S} \theta^s_{mid} P(s) - C' \\
s.t. \quad & K^+(s) \geq -P'(s)\hat{\theta}^s_+ \quad s \in S \\
& K^-(s) \geq P'(s)\hat{\theta}^s_- \quad s \in S \\
& t \geq 0, \gamma^s_+ \geq 0, \gamma^s_- \geq 0, \omega^n_+ \geq 0, \omega^n_- \geq 0
\end{aligned}
$$

In addition, constraints (4.29) and (4.31) also include ambiguous parameter $\tilde{d}$, and can be rewritten as follows:

For each $j \in J, tm \in N$, constraint (4.29) is equivalent to:

$$
\sum_{i \in I} w^{tm}_{ij} - (1 - \zeta^{tm}_{ij})(\bar{d}^{tm}_j - \hat{d}^{tm}_j * \min(1, \Gamma_{tm})) \leq 0
$$

For each $i \in I, tm \in N$, Constraint (4.31) is equivalent to solving following problem.

$$
\begin{aligned}
max & \sum_{i \in I} \zeta^{tm}_{ij} \hat{d}^{tm}_j e^{tm}_j \\
s.t \quad & 0 \leq e^{tm}_j \leq 1 \quad j \in J \\
& \sum_{j \in J} e^{tm}_j \leq \Gamma^D_{tm}
\end{aligned}
$$

Applying strong duality,

$$
\begin{aligned}
min \quad & \sum_{j \in J} \theta^i_{jtm} + \Gamma^D_{tm} \nu^i_{tm} \\
s.t. \quad & \theta^i_{jtm} + \nu^i_{tm} \geq \zeta^{tm}_{ij} \hat{d}^{tm}_j \\
& \theta^i_{jtm} \geq 0, \nu^i_{tm} \geq 0
\end{aligned}
$$

Reinjecting all of these equivalent constraints into the original problem, we arrive at the following MILP:

FLFV-RC

$$
\begin{aligned}
max \quad & \sum_{s \in S} \theta^s_{mid} P(s) - C' \\
s.t. \quad & F(01) = -\sum_{i \in I}(f_i x_i + c^0_i y^{01}_i) + \sum_{i \in I}\sum_{j \in J}(r - v^t_{ij})(w^{01}_{ij} + \zeta^{tm}_{ij}\bar{d}^{01}_j) + \sum_{j \in J}\lambda_{j01} + \Gamma^D_{01}\mu^{01} \\
& F(tm) = -\sum_{i \in I} c^t_i y^{tm}_i + \sum_{i \in I}\sum_{j \in J}(r - v^t_{ij})(w^{tm}_{ij} + \zeta^{tm}_{ij}\bar{d}^{tm}_j) + \sum_{j \in J}\lambda_{jtm} + \Gamma^D_{tm}\mu^{tm} \\
& \qquad \forall tm \in N, t \geq 1 \\
& P(s) = \sum_{tm \in \mathcal{P}(s)} F(tm) \quad s \in S \\
& \lambda_{jtm} + \mu_{tm} \geq \sum_{i \in I} \hat{d}^{tm}_j \zeta^{tm}_{ij}(r - v^t_{ij}) \quad \forall tm \in N \\
& \sum_{i \in I} w^{tm}_{ij} - (1 - \zeta^{tm}_{ij})(\bar{d}^{tm}_j - \hat{d}^{tm}_j * \min(1, \Gamma_{tm})) \leq 0 \quad j \in J, tm \in N \\
& y^{tm}_i \leq M x_i \quad i \in I, tm \in N \\
& \sum_{j \in R}(w^{tm}_{ij} + \zeta^{tm}_{ij}\bar{d}^{tm}_j) + \sum_{j \in J}\theta^i_{jtm} + \Gamma^D_{tm}\nu^i_{tm} \leq U_i \sum_{n \in \mathcal{P}(tm)} y^n_i \quad i \in I, tm \in N \\
& \theta^i_{jtm} + \nu^i_{tm} \geq \zeta^{tm}_{ij}\hat{d}^{tm}_j \quad i \in I, tm \in N \\
& x_i \in \{0, 1\} \quad i \in I \\
& K^+(s) \geq -P(s)\hat{\theta}^s_+ \quad s \in S \\
& K^-(s) \geq P(s)\hat{\theta}^s_- \quad s \in S \\
& t \geq 0, \gamma^s_+ \geq 0, \gamma^n_- \geq 0, \quad s \in S \\
& \omega^n_+ \geq 0, \omega^n_- \geq 0, \lambda_{jtm} \geq 0, \mu_{tm} \geq 0, \theta^i_{jtm} \geq 0, \nu^i_{tm} \geq 0 \quad tm \in N
\end{aligned}
$$

□

Lemma 4. *For a given problem instance L, let $f^*_{(FL-RC)}(L)$ and $f^*_{(FLFV-RC)}(L)$ be the optimal objective values to model **FL-RC** and **FLFV-RC**, respectively. Then,*

$$f^*_{(FL-RC)}(L) \leq f^*_{(FLFV-RC)}(L)$$

Proof. Suppose we constrain $\zeta^{tm}_{ij} = 0$ in **FLFV-R**. In this case, **FLFV-R** is equivalent to **FL-R**. Therefore, **FLFV-R** is an upper bound on **FL-R**.

We have seen that that **FL-RC** and **FLFV-RC** are deterministic robust counterparts of **FLFV-R** and **FL-R**, respectively. Therefore, we can conclude that **FLFV-RC** is also upper bound on **FL-RC**, and as a result,

$$f^*_{(FL-RC)}(L) \leq f^*_{(FLFV-RC)}(L)$$

□

4.3 Heuristic Algorithms

As we will observe in Section 4.4, we can solve moderate-sized problem instances of our model optimally using commercial optimization software such as CPLEX. However, we need alternative approaches to solve large instances.

Multistage stochastic linear programming problems (MSLP) can be equivalently formulated as recursive dynamic programs (DP). In these DPs, the cost-to-go function is piece-wise linear and convex. Thus, these cost-to-go functions can be approximated by linear cuts as in nested Benders decomposition, also known as L-shaped decomposition. This nested Benders decomposition algorithm approximates the convex cost-to-go functions by adding Benders' cuts, and it converges in finite steps to an optimal solution (Birge, 1985). However, when the size of problem is large, nested Benders decomposition becomes computationally intractable. Pereira and Pinto (1991) develop a sampling-based nested decomposition (ND) method, Stochastic Dual Dynamic Programming (SDDP), which utilizes stage-wise independence of stochastic process, so that the number of equations in DP reduces significantly. Z.-L. Chen and Powell (1999) prove that given finite scenarios and linear programs in every stage, SDDP converges in finite iterations. SDDP also has been applied to stochastic processes with stage-wise dependence, for example, using auto-regressive processes (Shapiro et al., 2013).

Motivated by Yu et al. (2018), who introduce the use of SDDP for multi-period stochastic facility location problems, we propose heuristic approaches that extend the concepts of ND and SDDP to our setting. Although our model is formulated as a MILP, integer variables only relate to the initial period, and thus our profit-to-go function (our model maximizes expected profit, and thus we use the term profit-to-go instead of cost-to-go) is convex. However, the approaches in the literature do not directly apply to our setting, since our model includes ambiguities in probabilities and demand. We incorporate these ambiguities by adopting the machinery of robust optimization. In both our ND and SDDP-based approaches, we express the value functions of the DP for each time period t given state variables and a realization of random variables at t. This implies that each value function is formulated as a single-period linear program. In our models, the number of scenarios only doubles in each period, and thus the size of scenario tree expands relatively slowly.

This motivates us to consider decomposing our large problem into a smaller number of sub-problems with longer horizons, and we call this sub-problem a "block" in what follows. In this case, each value function is formulated as a multi-period linear program. If the size of sub-problems is moderate, the computational time to solve them does not increase significantly. In addition, the total number of DP equations is reduced, and the number of iterations to converge may decrease. This approach accelerates convergence in most cases. We call these approaches Blocked ND (BND) and Blocked SDDP (BSDDP).

4.3.1 Dynamic Program

Recall the nominal model **FL**. We originally formulated this as multistage stochastic programming with binary variables, and we present equivalent dynamic program in this section.

We first introduce a new state variable s_i^{tm} to represent the cumulative amount of capacity built up to time t, that is,

$$s_i^{tm} = \sum_{n \in \mathcal{P}(tm)} y_i^n.$$

We modify the facility opening decision variable x_i as follows. Recall that for each node, the constraint (4.6)

$$y_i^{tm} \leq M x_i \quad i \in I$$

needs to be satisfied. We introduce a continuous variable $\hat{x}_i^{tm}$ for each node, the following constraint for the root node 01,

$$\hat{x}_i^{01} = x_i \quad i \in I$$

and the following constraints for each node $tm \in N$.

$$\hat{x}_i^{tm} = \hat{x}_i^{A(tm)} \quad i \in I$$

In addition, we modify constraint (4.6) using the new variable $\hat{x}_i^{tm}$ as follows.

$$y_i^{tm} \leq M \hat{x}_i^{tm} \quad i \in I.$$

This modification ensures that the integer variable only appears in the initial period in the following DP equations.

We use vector $x_{01} \in \mathbb{Z}_+^I$ to denote $\{x_i^{01}\}_{i \in I}$. For each $tm \in N$, we use vector $\hat{x}_{tm} \in \mathbb{R}_+^I$ to denote $\{\hat{x}_i^{tm}\}_{i \in I}$, vector $s_{tm} \in \mathbb{R}_+^I$ to denote $\{s_i^{tm}\}_{i \in I}$, and vector $d_{tm} \in \mathbb{Z}_+^J$ to denote demand $\{d_j^{tm}\}_{j \in J}$. Let $w_{tm} := (s_{tm}, \hat{x}_{tm})$ be a state variable vector for a node tm.

We denote the objective function at tm as follows:

$$g_0(\hat{x}_{01}, s_{01}, z_{01}) = -\sum_{i \in I} (f_i \hat{x}_i + c_i^{01} s_i^{01}) + \sum_{i \in I} \sum_{j \in J} (r^0 - v_{ij}^{01}) z_{ij}^{01}$$

$$g_t(s_{A(tm)}, s_{tm}, z_{tm}) = -\sum_{i \in I} c_t (s_i^{tm} - s_i^{A(tm)}) + \sum_{i \in I} \sum_{j \in J} (r^t - v_{ij}^t) z_{ij}^{tm}$$

.

The DP equations for the optimal value function of **FL** at node $tm \in N$ can then be written as follows:

$$(P_{01}) \quad max \quad \sum_{i \in I} g_0(\hat{x}_{01}, s_{01}, z_{01}) + \sum_{n \in C(0)} \pi_n Q_n(w_{01}) \tag{4.40}$$

$$s.t. \quad \sum_{i \in I} z_{ij}^{01} \leq d_j^{01} \quad j \in J \tag{4.41}$$

$$s_i^{01} \leq M x_i \quad i \in I \tag{4.42}$$

$$\hat{x}_i^{01} = x_i \quad i \in I \tag{4.43}$$

$$\sum_{j \in R} z_{ij}^{01} \leq U_i s_i^{01} \quad i \in I \tag{4.44}$$

$$s_i^{01} \geq 0, x_i \in [0,1], \hat{x}_i^{01} \geq 0, z_{ij}^{01} \geq 0 \quad i \in I, j \in J \tag{4.45}$$

where $w_0 = (0, 0)$, and for each node $tm \in N \setminus \{01\}$:

$$(P_{tm}) \quad Q_{tm}(w_{A(tm)}) :=$$

$$max \quad g_t(s_{A(tm)}, s_{tm}, z_{tm}) + \sum_{n \in C(tm)} \pi_n Q_n(w_{tm}) \tag{4.46}$$

$$s.t. \quad \sum_{i \in I} z_{ij}^{tm} \leq d_j^{tm} \quad j \in J \tag{4.47}$$

$$s_i^{tm} \leq M x_i^{tm} \quad i \in I \tag{4.48}$$

$$\hat{x}_i^{tm} = \hat{x}_i^{A(tm)} \quad i \in I \tag{4.49}$$

$$s_i^{A(tm)} \leq s_i^{tm} \quad i \in I \tag{4.50}$$

$$\sum_{j \in R} z_{ij}^{tm} \leq U_i s_i^{tm} \quad i \in I \tag{4.51}$$

$$s_i^{tm} \geq 0, \hat{x}_i^{tm} \geq 0, z_{ij}^{tm} \geq 0 \quad i \in I, j \in J \tag{4.52}$$

with $Q_{Tm}(w_{(T+1)m}) = 0$.

In the sections below, we use the following notation. We let $X_{tm}^A(w_{A(tm)}, d_{tm})$ be a feasible set defined by constraints (4.47)-(4.52), and $X_t^B(w_0, d_{01})$ be a feasible set defined by constraints (4.41)-(4.45). We also denote the expected profit to-go function by $\mathcal{Q}_{tm}(w_{tm}) := \sum_{n \in C(tm)} \pi_n Q_n(w_n)$.

4.3.2 Nested Decomposition

In this section, we describe the how the nested decomposition (ND) algorithm (Birge and Louveaux, 2011) can be applied to our nominal model **FL**, and in the next section, we extend and modify this approach so that it can be applied to our robust model. In this algorithm, the problem for each node tm ((4.53)) is solved in a forward and backward step. In the forward step, the solution for stage $t+1$ is updated, and in the backward step, cuts for stage $t-1$ are generated. This procedure iterates these forward and backward steps until the stopping criteria is met. We outline algorithm details below:

In each iteration l, the ND algorithm consists of a forward step and a backward step. In iteration l, the forward step considers all nodes $tm \in N_t$ from $t = 0$ to T, where N_t denotes a set of nodes at t. In particular, in iteration l, at node tm, given $s_{A(tm)}^l$ and $x_{A(tm)}^l$ obtained from solving the problem of its parent node $A(tm)$, we solve the problem $ND_{tm}^l(w_{A(tm)}^l, \psi_{tm}^l)$ defined as:

$$(ND_{tm}^l(w_{A(tm)}^l, \psi_{tm}^l)) \tag{4.53}$$

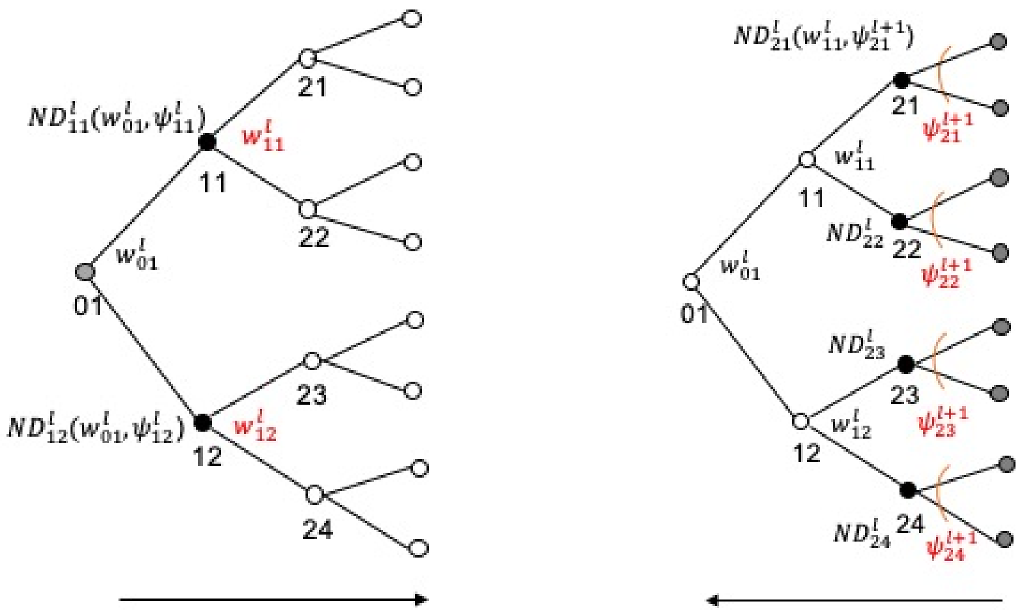
$$\underline{Q}_{tm}^l(w_{A(tm)}^l, \psi_{tm}^l) := max \quad g_t(s_{A(tm)}^l, s_{tm}^l, z_{tm}^l) + \psi_{tm}^l(s_{tm}^l, x_{tm}^l)$$

$$s.t. \quad (s_{tm}^l, \hat{x}_{tm}^l, z_{tm}^l) \in X_{tm}^A(w_{A(tm)}, d_{tm})$$

where the approximate expected profit to-go function is:

$$\psi_{tm}^l(s_{tm}^l, \hat{x}_{tm}^l) := max\{\phi_{tm} : \phi_{tm} \leq B_{tm}$$

$$\phi_{tm} \leq \sum_{n \in C(tm)} \pi_n(\beta_n^q + \alpha_n^{Aq} s_{tm}^l + \alpha_n^{Bq} \hat{x}_{tm}^l), \forall q = 1..l-1\}. \tag{4.54}$$



(a) Forward **(b)** Backward

Figure 4.1. Forward and backward steps of the nested decomposition algorithm

$\psi^l_{tm}(s^l_{tm}, \hat{x}^l_{tm})$ provides a piece-wise linear convex under-approximation of the expected profit-to-go function $\mathcal{Q}_{tm}(w_{tm})$, and we assume there is an upper bound B_{tm} on ϕ_{tm} to avoid unboundedness.

$(s^l_{tm}, \hat{x}^l_{tm})$, an optimal solution to $ND^l_{tm}(w^l_{A(tm)}, \psi^l_{tm})$ is passed as a state variable w^l_{tm} to the problems $ND^l_k(w^l_{tm}, \psi^l_k)$ of child nodes $k \in C(tm)$ of node tm. When all forward problems are solved in iteration l, we can obtain a lower bound (**LB**) to the optimal value of **FL** as follows.

$$\mathbf{LB} := g_0(x^l_{01}, s^l_{01}, z^l_{01}) + \sum_{s \in S} \theta_s \sum_{tm \in \mathcal{P}} g_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm}). \tag{4.55}$$

Figure 4.1 illustrates the forward and backward steps of the ND algorithm. In this graphical example, at time stage 1 in iteration l, the forward step proceeds as follows. Given the state variable $w^l_{01} = (s^l_{01}, \hat{x}^l_{01})$, which is obtained by solving ND^l_{01}, ND^l_{11} and ND^l_{12} are solved independently. The solutions $(s^l_{11}, \hat{x}^l_{11})$ and $(s^l_{12}, \hat{x}^l_{11})$ to ND^l_{11} and ND^l_{12} are passed as a state variable to solve the problems at their child nodes at $t = 2$.

The backward step starts from the final stage T. In the final stage T, $\psi^l_{Tm} = 0$ since there is no expected profit-to-go function. The goal of the backward step is to update the expected profit to-go function at each node by adding a new cut. In particular, at a node Tm of the final stage T, we solve $ND^l_{Tm}(w^l_{A(Tm)}, \psi^l_{Tm})$ given a candidate solution obtained from the forward step, and collect cut coefficients $(\beta^l_{Tm}, \alpha^{Al}_{Tm}, \alpha^{Bl}_{Tm})$. The cut coefficients are obtained from a linear inequality that approximates the true value function $Q_{Tm}(w_{A(tm)})$ based on weak duality.

When cut coefficients are collected for all nodes $Tm \in N_T$, we proceed to stage $T - 1$. At node $(T-1)m$, a cut is added by using cut coefficients generated from its child nodes and inequality (4.54), and thus $\psi^l_{(T-1)m}$ is updated to $\psi^{l+1}_{(T-1)m}$. In other words, we generate a cut, which is an inequality that approximate true value of $Q_{Tm}(.)$, by taking the average of inequalities of all children of node $(T-1)m$. Then we solve the updated problem $ND^l_{Tm}(w^l_{A((T-1)m)}, \psi^{l+1}_{(T-1)m})$ at node $(T-1)m$ in the backward step to collect cut coefficients. This set of cut coefficients is aggregated with those of its sibling node in stage $T-2$ to generate a cut for its parent node. The backward step continues in this way until it reaches the root node (01) of the tree.

In Figure 4.1, the backward step is proceeded as follows. At node 21 in iteration l, given cut coefficients obtained from solving ND^l_{31} and ND^l_{32}, ψ^l_{21} is updated to ψ^{l+1}_{21}. Then we solve ND^l_{21} to collect a set of cut coefficients, which is aggregated with cut coefficients from ND^l_{21} to update ψ_{11}.

Since the cuts are approximations of the true expected to-go profits, the optimal objective value obtained by solving

$$(ND^l_{01}(w_0, \psi^l_{01})) \quad \underline{Q}^l_{01}(w_0, \psi^{l+1}_{01}) := max \quad g_0(x^l_{01}, s^l_{01}, z^l_{01}) + \psi^l_{01}(s^l_{01}, \hat{x}^l_{01})$$
$$s.t. \quad (s^l_{01}, x^l_{01}, z^l_{01}) \in X^B_0(w_0, d_0)$$

provides an upper bound, **UB**, to the optimal expected profit of the original problem **FL**. These forward and backward steps are repeated until the stopping criterion, $\mathbf{UB} - \mathbf{LB} \leq \epsilon$, is met. We summarize the ND algorithm below.

Step 1: Initialization. $LB \leftarrow -\infty$, $UB \leftarrow \infty$, $l \leftarrow 1$
For all $tm \in N$, initialize ψ^1_{tm}, $\psi^1_{tm} = \{\phi_{tm} : \quad \phi_{tm} \leq B_{tm}\}$

Step 2: Forward Step.
For $t = 0..T$ and $tm \in N_t$, solve $ND^l_{tm}(w_{tm}, \psi^l_{tm})$, collect solution $(\hat{x}^l_{tm}, s^l_{tm}, z^l_{tm})$, update $g_0(x^l_{01}, s^l_{01}, z^l_{01})$ and $g_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm})$.

Step 3:

$$\mathbf{LB} \leftarrow g_0(x^l_{01}, s^l_{01}, z^l_{01}) + \sum_{s \in S} \theta_s \sum_{tm \in \mathcal{P}} g_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm})$$

Step 4: Backward Step.
For $t = T..1$ and $tm \in N_t$, do followings.
If $n \in C(tm)$ exists,
add a cut (4.59) using cut coefficients $(\beta^l_n, \alpha^{Al}_n, \alpha^{Bl}_n)$ for all $n \in C(tm)$, then $\psi^{l+1}_{tm} \leftarrow \psi^l_{tm}$
Otherwise, $\psi^{l+1}_{tm} \leftarrow \psi^l_{tm}$.
Solve $ND^l_{tm}(w_{tm}, \psi^{l+1}_{tm})$, and collect cut coefficients $(\beta^l_{tm}, \alpha^{Al}_{tm}, \alpha^{Bl}_{tm})$

Step 5: Solve $ND^l_{01}(w_0, \psi^{l+1}_{01})$ and set **UB** be the optimal value.

Step 6: If some stopping criterion is met, then stop.
Otherwise, $l \leftarrow l + 1$ and go to Step 2.

4.3.3 The Nested Decomposition Approach for a Model with Ambiguities in Probabilities and Demand

We now extend the ND approach to the problem with ambiguities in probabilities and demand. In the robust setting with respect to ambiguous parameters $\tilde{d}_{tm}$ and $\tilde{\pi}_n$, the Bellman equations can be written as follows:

$$\begin{aligned} Q_{tm}(w_{A(tm)}) := max \min_{\tilde{d}_{tm} \in \mathcal{D}} \quad & g_t(s_{A(tm)}, s_{tm}, z_{tm}) + \max_{\pi_n \in \Omega, n \in C(tm)} \sum_{n \in C(tm)} \pi_n Q_n(w_{tm}) \\ s.t. \quad & (s_{tm}, \hat{x}_{tm}, z_{tm}) \in X^A_{tm}(w_{A(tm)}, \tilde{d}_{tm}) \end{aligned} \tag{4.57}$$

$$\begin{aligned} Q_{01} := max \min_{d_{tm} \in \mathcal{D}} \quad & g_0(x_{01}, s_{01}, z_{01}) + \max_{\pi^n \in \Omega, n \in C(01)} \sum_{n \in C(01)} \pi_n Q_n(w_{01}) \\ s.t. \quad & (s_{01}, x_{01}, z_{01}) \in X^B_0(w_0, \tilde{d}_{01}) \end{aligned} \tag{4.58}$$

In the nominal setting, to find the optimal value of Q_{01}, we solve the problem ND^l_{tm} (4.53), which is formulated as a linear program, at node tm in each iteration l in forward and backward steps of the ND algorithm. In order to show that we can use a ND-based approach in our robust setting, we need to show that the problem ND^l_{tm} can still be formulated as a linear program in this setting.

We first focus on the inner maximization problem in the objective function (4.57) of $Q_{tm}(w_{A(tm)})$, which is,

$$\max_{\pi^n \in \Omega, n \in C(tm)} \sum_{n \in C(tm)} \pi_n Q_n(w_{tm}) \tag{4.59}$$

Recall that in our setting, each node tm has only two child nodes, and we denote these nodes by $c(tm)_s$ and $c(tm)_f$. Since $\pi_{c(tm)_s} + \pi_{c(tm)_f} = 1$, given w_{tm}, the problem (4.59) can be written as a linear program with respect to $\pi_{c(tm)_s}$:

$$\begin{aligned} \max \quad & \pi_{c(tm)_s} Q_{c(tm)_s}(w_{tm}) + (1 - \pi_{c(tm)_s}) Q_{c(tm)_f}(w_{tm}) \\ s.t. \quad & \pi_{c(tm)_s} \leq \pi^+_{c(tm)_s} && (4.60) \\ & - \pi_{c(tm)_s} \leq -\pi^-_{c(tm)_s} && (4.61) \end{aligned}$$

This auxiliary problem is feasible (since $\pi_{c(tm)_s} = \bar{\pi}_{c(tm)_s}$ is a feasible solution) and bounded. Let γ^+_{tm} and γ^-_{tm} be the dual variables corresponding to constraints (4.60) and (4.61). Then, the dual of the auxiliary problem is:

$$\begin{aligned} \min \quad & Q_{c(tm)_f}(w_{tm}) + \pi^+_{c(tm)_s} \gamma^+_{tm} - \pi^-_{c(tm)_s} \gamma^-_{tm} \\ s.t. \quad & \gamma^+_{tm} - \gamma^-_{tm} \geq Q_{c(tm)_s}(w_{tm}) - Q_{c(tm)_f}(w_{tm}) \end{aligned}$$

By reinjecting this into the DP equations (4.57) and (4.58), we can rewrite these DP equations as follows:

$$
\begin{aligned}
&Q_{tm}(w_{A(tm)}) := \\
&\quad max \min_{\tilde{d}_{tm}\in\mathcal{D}} \quad g_t(s_{A(tm)}, s_{tm}, z_{tm}) + Q_{c(tm)_f}(w_{tm}) + \pi^+_{c(tm)_s}\gamma^+_{tm} - \pi^-_{c(tm)_s}\gamma^-_{tm} \\
&\qquad\qquad s.t. \quad (s_{tm}, \hat{x}_{tm}, z_{tm}) \in X^A_{tm}(w_{A(tm)}, \tilde{d}_{tm}) \\
&\qquad\qquad\qquad \gamma^+_{tm} - \gamma^-_{tm} \geq Q_{c(tm)_s}(w_{tm}) - Q_{c(tm)_f}(w_{tm})
\end{aligned}
\tag{4.62}
$$

$$
\begin{aligned}
Q_{01} := \quad & max \min_{\tilde{d}_{tm}\in\mathcal{D}} \quad g_t(x_{01}, s_{01}, z_{01}) + Q_{12}(w_{tm}) + \pi^+_{11}\gamma^+_{01} - \pi^-_{11}\gamma^-_{01} \\
& \qquad s.t. \quad (s_{01}, x_{01}, z_{01}) \in X^B_0(w_0, \tilde{d}_{01}) \\
& \qquad\qquad \gamma^+_{01} - \gamma^-_{01} \geq Q_{11}(w_{01}) - Q_{12}(w_{01}).
\end{aligned}
$$

In the inner minimization problem, ambiguous parameter $\tilde{d}_{tm}$ is only found in the constraint (4.47), which is,

$$\sum_{i\in I} z^{tm}_{ij} \leq \tilde{d}^{tm}_j \quad j \in J, \tilde{d}_{tm} \in \mathcal{D}.$$

This constraint is equivalent to:

$$\sum_{i\in I} z^{tm}_{ij} \leq \bar{d}^{tm}_j - \hat{d}^{tm}_j \times min(1, \Gamma^D_{tm}). \tag{4.63}$$

We denote by $\tilde{X}^A_{tm}(w_{A(tm)}, \tilde{d}_{tm})$ a feasible set defined by constraints (4.48)-(4.52) and (4.63), all of which are linear.

Then, in the robust setting, our ND-based algorithm solves the following NDR^l_{tm} problem, which is formulated as a linear program:

$$
\begin{aligned}
&\underline{Q}^l_{tm}(w^l_{A(tm)}, \hat{\psi}^l_{C(tm)}) := \\
&\quad max \min_{\tilde{d}_{tm}\in\mathcal{D}} \quad g_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm}) + \psi^l_{c(tm)_f}(w^l_{tm}) + \pi^+_{c(tm)_s}\gamma^+_{tm} - \pi^-_{c(tm)_s}\gamma^-_{tm} \\
&\qquad\qquad s.t. \quad (s^l_{tm}, \hat{x}^l_{tm}, z^l_{tm}) \in X^A_{tm}(w^l_{A(tm)}, \tilde{d}_{tm}) \\
&\qquad\qquad\qquad \gamma^+_{tm} - \gamma^-_{tm} \geq \hat{\psi}^l_{c(tm)_s}(w^l_{tm}) - \hat{\psi}^l_{c(tm)_f}(w^l_{tm})
\end{aligned}
$$

$$
\begin{aligned}
&\underline{Q}^l_{01}(w_0, \hat{\psi}^l_{C(tm)}) := \\
&\quad max \min_{\tilde{d}_{01}\in\mathcal{D}} \quad g_0(x^l, s^l_{01}, z^l_{01}) + \psi^l_{12}(w^l_{01}) + \pi^+_{c(01)_s}\gamma^+_{01} - \pi^-_{11}\gamma^-_{01} \\
&\qquad\qquad s.t. \quad (s^l_{01}, \hat{x}^l_{01}, z^l_{01}) \in X^B_0(w_0, \tilde{d}_{01}) \\
&\qquad\qquad\qquad \gamma^+_{01} - \gamma^-_{01} \geq \hat{\psi}^l_{11}(w^l_{01}) - \hat{\psi}^l_{12}(w^l_{01})
\end{aligned}
$$

where $\hat{\psi}^l(w^l_{A(tm)})$, the approximation of value function $Q_{tm}(w_{A(tm)})$, is defined as:

$$
\begin{aligned}
\hat{\psi}^l_{tm}(w^l_{A(tm)}) := max\{\phi_{tm} : \quad & \phi_{tm} \leq B_{tm} \\
& \phi_{tm} \leq (\beta^q_{tm} + \alpha^{Aq}_{tm} s_{A(tm)} + \alpha^{Bq}_{tm} \hat{x}_{A(tm)}), \forall q = 1..l-1\}
\end{aligned}
\tag{4.64}
$$

The corresponding NDR^l_{tm} for model **FLFV-RC** can be found in the Appendix A.1.

Note that we add two cuts here instead of an aggregated single cut which was the case in the ND algorithm presented in the previous section. While our ND-based algorithm has similar steps to the original ND algorithm, the backward step needs some modification due to these multi-cuts.

The detailed algorithm is as follows. Our ND-based algorithm for the robust model also consists of a forward step and a backward step in each iteration. The forward step is the same as in the traditional ND algorithm. We solve NDR^l_{tm} for each node tm, and for each iteration l, and update a proposed solution. We compute the **LB** to the optimal value of the robust model as follows. For $n \in N$, we let π'_n be π^-_n if the trial result is success at node n, and π^+_n otherwise. Then we compute the worst-case scenario probability $\theta'_s = \prod_{n\in\mathcal{P}} \pi'_n$, and using this the **LB** becomes

$$\mathbf{LB} := g_0(x^l_{01}, s^l_{01}, z^l_{01}) + \sum_{s\in S} \theta'_s \sum_{tm\in\mathcal{P}} g_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm}).$$

In the backward step, we solve NDR^l_{tm}, and collect cut coefficients obtained from a linear inequality that approximates the true value function Q_{tm} based on weak duality. In contrast to the traditional ND algorithm, where a cut is added to its parent node $A(tm)$ after being aggregated with its sibling node, a cut is added to self node tm. Thus, at node tm, $\hat{\psi}^l_{tm}$ is updated to $\hat{\psi}^{l+1}_{tm}$. When all problems are solved for all nodes tm at stage t, we proceed to stage $t-1$. At a node $(t-1)m$, given updated cuts from its child nodes, we solve the updated problem $NDR^l_{(t-1)m}$ at node $(t-1)m$, and generate a cut for node $(t-1)m$. The backward step continues in this way until it reaches the root node (01) of the tree. We summarize the modified ND algorithm for our robust model below.

Step 1: Initialization. $LB \leftarrow -\infty$, $UB \leftarrow \infty$, $l \leftarrow 1$
For all $tm \in N$, initialize ψ^1_{tm}, $\psi^1_{tm} = \{\phi_{tm} : \quad \phi_{tm} \le B_{tm}\}$

Step 2: Forward Step.
For $t = 0..T$ and $tm \in N_t$,
solve $NDR^l_{tm}(w_{tm}, \hat{\psi}^l_{C(tm)})$, collect solution $(\hat{x}^l_{tm}, s^l_{tm}, z^l_{tm})$,
and update $g_0(x^l_{01}, s^l_{01}, z^l_{01})$ and $g_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm})$.

Step 3:

$$\mathbf{LB} \leftarrow g_0(x^l_{01}, s^l_{01}, z^l_{01}) + \sum_{s\in S} \theta'_s \sum_{tm\in\mathcal{P}} g_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm})$$

Step 4: Backward Step.
For $t = T..1$ and $tm \in N_t$, do followings.
Solve $NDR^l_{tm}(w_{tm}, \psi^{l+1}_{C(tm)})$, collect cut coefficients $(\beta^l_{tm}, \alpha^{Al}_{tm}, \alpha^{Bl}_{tm})$, and
add a cut (4.64) to ψ^l_{tm}, and set this as ψ^{l+1}_{tm}.

Step 5: Solve $NDR^l_{01}(w_0, \psi^{l+1}_{C(01)})$ and set **UB** be the optimal value.

Step 6: If some stopping criterion is satisfied, then stop.
Otherwise, $l \leftarrow l+1$ and go to Step 2.

4.3.4 Blocked Nested Decomposition

In the ND algorithm, we solve the problem ND^l_{tm} for all nodes $tm \in N$ in each iteration l. Each ND^l_{tm} is a single-period problem, and thus the computation time to solve each problem is very short. This algorithm generates as many DP equations as the number of nodes in the scenario tree of a multistage stochastic program, and requires many iterations to converge, and thus it often takes a long time, even though each sub-problem can be solved quickly.

In our setting, each node in the scenario tree only has two child nodes, and the size of the scenario tree expands relatively slowly as the planning horizon increases. Thus, moderate size problem instances can be solved optimally in a short time using commercial software such as CPLEX. For example, as we will see in Section 4.4, a problem instance with $T = 5$, 10 potential facility sites, and 20 demand locations takes less than 50 seconds to solve optimally.

In our ND-based algorithm, we propose splitting our one large problem into a smaller number of sub-problems by increasing individual sub-problem horizons. This implies that each value function will be formulated as a multi-period linear program. If the size of sub-problems is moderate, the computational time to solve them does not increase significantly as described above. In addition, the total number of DP equations is reduced, and the number of iterations to converge may decrease. Thus, we can expect relatively faster convergence in this approach.

In this section, we first formulate our models using recursive blocked dynamic programming equations, and then introduce the Blocked Nested Decomposition algorithm.

Blocked Dynamic Programming Equations

We divide a planning horizon into B separate time intervals, $[0, t_1], [t_1 + 1, t_2]$... and $[t_{B-1} + 1, T]$. In each division, sub-trees are created rooted from each node in the first period of the corresponding time interval . We call a sub-tree a *block*. For example, in Figure 4.2, we divide a scenario tree into two $[0, 2]$ and $[3, 3]$ time intervals. In division $[0, 2]$, there is 1 sub-tree (block) rooted from node 01, and in division $[3, 3]$, there are 8 sub-trees (blocks) rooted from all nodes tm at $t = 3$. We denote a set of all blocks by $\mathcal{B}$.

Each block, denoted by b, is also a scenario tree with a root node $r(b)$, and $d_{[r(b),t_b]}$ denotes a demand vector from the root node $r(b)$ to all descendant nodes through stage t_b. The set of nodes in the block b is denoted by N_b. The set of nodes on the path from root node $r(b)$ to node n, including node n, is denoted by $\mathcal{P}(r(b), n)$. $\mathcal{S}(n)$ denotes the set of scenarios in which node n is included. $\mathcal{L}_b$ denotes all leaf nodes at stage t_h in the scenario tree of block b. Let p^k be the probability of reaching node $k \in \mathcal{L}_b$, where $p^k = \prod_{n\in\mathcal{P}(r(b),k)} \pi_n$.

We use vector $\mathbf{x}_{N(b)}$ for each block b to denote $\{\hat{x}^n_i\}_{i\in I,n\in N_b}$, vector $\mathbf{s}_{N(b)}$ to denote $\{s^n_i\}_{i\in I,n\in N_b}$, and vector $\mathbf{z}_{N(b)}$ to denote $\{z^n_{ij}\}_{i\in I,j\in J,n\in N_b}$.

We denote the objective function at block 1 and other blocks $b > 1$ as follows:

$$g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) = g_0(x_{01}, s_{01}, z_{01}) + \sum_{k\in\mathcal{L}_b} p^k \sum_{n\in\mathcal{P}(01,k)} g_t(s_{A(n)}, s_n, z_n)$$

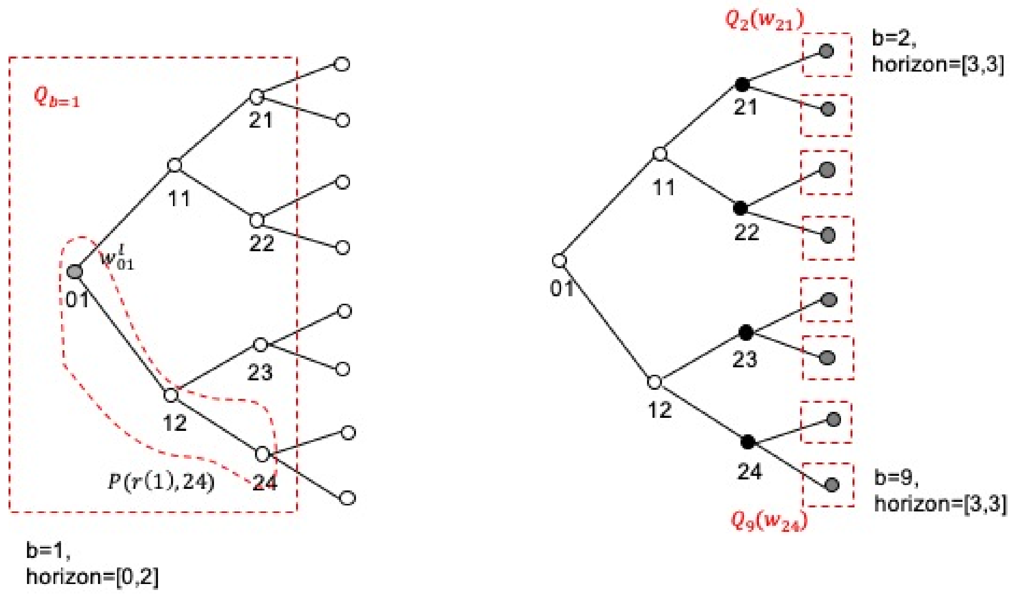


Figure 4.2. Blocked Nested Decomposition: Notation

$$g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) = \sum_{k \in \mathcal{L}_b} p^k \sum_{n \in \mathcal{P}(r(b),k)} g_t(s_{A(n)}, s_n, z_n)$$

.

Given this notation, we can write the *blocked DP equations* for the optimal value function of **FL** at the first block $(r(b=1) = 01)$ as follows:

$$Q_{b=1} := max \quad g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) + \sum_{k \in \mathcal{L}_b} p^k \sum_{\substack{n \in C(k), \\ r(b')=n}} \pi_k Q_{b'}(w_k)$$

$$s.t. \quad \sum_{i \in I} z_{ij}^n \leq d_j^n \quad j \in J, n \in N_b \tag{4.65}$$

$$s_i^n \leq M\hat{x}_i^n \quad i \in I, n \in N_b \tag{4.66}$$

$$\hat{x}_i^n = \hat{x}_i^{A(n)} \quad i \in I, tm \in N_b \tag{4.67}$$

$$\sum_{j \in R} z_{ij}^n \leq U_i s_i^n \quad i \in I, n \in N_b \tag{4.68}$$

$$x_i^{01} \in [0,1] \tag{4.69}$$

$$\hat{x}_i^n, s_i^n \geq 0, z_{ij}^n \geq 0 \quad i \in I, j \in J, n \in N_b \tag{4.70}$$

where for other blocks $b > 1$,

$$Q_b(w_{A(r(b))}) := max \quad g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) + \sum_{k \in \mathcal{L}_b} p^k \sum_{\substack{n \in C(k), \\ r(b')=n}} \pi_k Q_{b'}(w_k)$$

$$s.t. \quad (4.65) - (4.68), (4.70)$$

with $Q_b(w_{Tm}) = 0$ for $r(b) \in C(Tm)$.

We let $X_b^A(w_{A(r(b))}, d_{[(r(b),t_b]})$ be a feasible set defined by constraints (4.65)-(4.68), and (4.70). We also let $X_1^B(w_0, d_{[01,t_1]})$ be a feasible set defined by constraints (4.65)-(4.70).

Blocked Dynamic Programming Equations with Ambiguous Probabilities and Demand

In this subsection, we formulate the robust model **FL-RC** using recursive blocked dynamic programming equations. Recall that we use the probability p^k of reaching node $k \in \mathcal{L}_b$ and $\pi_{k'}$ for $k' \in C(k)$ in the blocked DP equation $Q_b(w_{A(r(b))})$ for block b. Let $\theta'_{k'}$ be the probability of reaching node k', that is, $\theta'_{k'} = p^k \pi_{k'}$. Then, we need to define the uncertainty set in terms of $\theta'_{k'}$ for the robust version of $Q_b(w_{A(r(b))})$. Let $\mathcal{C}(\mathcal{L}_b)$ be a set of child nodes of all nodes in $\mathcal{L}_b$.

We define the positive and negative scaled deviation of $\theta'_{k'}$ from its nominal value $\theta'^{mid}_{k'} = p^{A(k')}\bar{\pi}_{k'}$ to be

$$z^+_{k'} = (\theta'_{k'} - \theta'^{mid}_{k'})/\hat{\theta}^+_{k'}$$

and

$$z^-_{k'} = (\theta'_{k'} - \theta'^{mid}_{k'})/\hat{\theta}^-_{k'}$$

where $\hat{\theta}^+_{k'} = \prod_{n\in\mathcal{P}(r(b),k')} \pi^+_n - \theta'^{mid}_{k'}$ and $\hat{\theta}^-_{k'} = \theta'^{mid}_{k'} - \prod_{n\in\mathcal{P}(r(b),k')} \pi^-_n$, and the scaled deviations take values in $[0, 1]$. Then, $\theta'_{k'}$ can be written as:

$$\theta'_{k'} = \theta'^{mid}_{k'} + \hat{\theta}'^+_{k'} z^+_{k'} - \hat{\theta}'^-_{k'} z^-_{k'}.$$

Recall that given a budget of uncertainty, we assume that the total variation of scenario realization probabilities cannot exceed Γ. For each block b, we define the budget of uncertainty as described below.

Let S_b be the number of the child nodes of the all leaf nodes in the tree of block b. If block b includes the last period of nodes from the original scenario tree, we let S_b be the total number of scenarios in the tree of block b. Then, for the tree in each block b, the total variation of the probabilities $\theta'_{k'}$ for all $k' \in C(k), k \in \mathcal{L}_b$ is forced not to exceed $\frac{\Gamma}{|S|} * S_b$. More formally, we can write this as follows:

$$\sum_{k\in\mathcal{L}_b} \sum_{k'\in C(k)} z^+_{k'} + z^-_{k'} \leq \frac{\Gamma}{|S|} * S_b$$

Then, the uncertainty set Θ_2 for θ' is:

$$\begin{aligned}\Theta_2 = \Bigg\{ \theta' \in R^{|\mathcal{C}(\mathcal{L}_b)|} | \theta'_{k'} = \theta'^{mid}_{k'} + \hat{\theta}'^+_{k'} z^+_{k'} - \hat{\theta}'^-_{k'} z^-_{k'},& \\ 0 \leq z^+_{k'} \leq 1, 0 \leq z^-_{k'} \leq 1, \quad \forall k' \in \mathcal{C}(\mathcal{L}_b),& \\ \pi^-_n \sum_{i\in S(A(n))} \theta'_i \leq \sum_{i\in S(n)} \theta'_i \leq \pi^+_n \sum_{i\in S(A(n))} \theta'_i, \quad \forall n \in N_b \cup \mathcal{C}(\mathcal{L}_b)& \\ \sum_{k'\in\mathcal{C}(\mathcal{L}_b)} \theta'_{k'} = 1, \quad \sum_{k'\in\mathcal{C}(\mathcal{L}_b)} z^+_{k'} + z^-_{k'} \leq \frac{\Gamma}{|S|} * S_b \Bigg\}&\end{aligned}$$

Given the uncertainty set Θ_2 and $\mathcal{D}$, the *blocked DP equation* for the value function of **FL-RC** at the first block is:

$$\begin{aligned}\tilde{Q}_{b=1} := max \min_{\substack{\tilde{d}_{[(r(1),t_1]}\in\mathcal{D},\\ \theta'\in\Theta_2}} \quad & g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) + \sum_{k\in\mathcal{L}_b}\sum_{\substack{n\in C(k),\\ r(b')=n}} \theta'_n Q_{b'}(w_k)\\ s.t. \quad & (\mathbf{s}_{N_1}, \mathbf{x}_{N_1}, \mathbf{z}_{N_1}) \in X_1^B(w_0, d_{[(r(1),t_h]})\end{aligned}$$

where for other blocks $b > 1$,

$$\begin{aligned}\tilde{Q}_b(w_{A(r(b))}) := max \min_{\substack{\tilde{d}_{[(r(b),t_b]}\in\mathcal{D},\\ \theta'\in\Theta_2}} \quad & g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) + \sum_{k\in\mathcal{L}_b}\sum_{\substack{k'\in C(k),\\ r(b')=k'}} \theta'_{k'} Q_{b'}(w_k)\\ s.t. \quad & (\mathbf{s}_{N_b}, \mathbf{x}_{N_b}, \mathbf{z}_{N_b}) \in X_b^A(w_{A(r(b))}, \tilde{d}_{[(r(b),t_h]}).\end{aligned} \tag{4.71}$$

In the following lemma, we show that $\tilde{Q}_b$ can be rewritten as ${Q'}_b$. ${Q'}_b$ allows us to apply the our ND-based algorithm - Blocked Nested Decomposition, which will be described in the following subsection.

Lemma 5. *The DP equation $\tilde{Q}_b$ for block b can be equivalently formulated as the following ${Q'}_b$:*

$$\begin{aligned}max \quad & \sum_{k\in\mathcal{L}(b)}\sum_{n\in\mathcal{P}(r(b),k)} \theta_{k'}^{mid} g_t(s_{A(n)}, s_n, z_n) + \sum_{\substack{k'\in\mathcal{C}(\mathcal{L}_b),\\ r(b')=k'}} \theta_{k'}^{mid} Q_{b'} - C'\\ s.t. \quad & (\mathbf{s}_{N_b}, \mathbf{x}_{N_b}, \mathbf{z}_{N_b}) \in \tilde{X}_b^A(w_{A(r(b))}, d_{[(r(b),t_h]})\\ & K^+(k') \geq -\sum_{n\in\mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) - Q_{b'}\hat{\theta}_{k'}^+ \quad \forall k' \in \mathcal{C}(\mathcal{L}_b)\\ & K^-(k') \geq \sum_{n\in\mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) + Q_{b'}\hat{\theta}_{k'}^- \quad \forall k' \in \mathcal{C}(\mathcal{L}_b)\\ & t \geq 0, r_+^{k'} \geq 0, r_-^{k'} \geq 0, w_+^n \geq 0, w_-^n \geq 0\end{aligned}$$

where

$$C' = t\frac{\Gamma}{|S|} * S_b + \sum_{k'\in\mathcal{C}(\mathcal{L}_b)} (r_+^{k'} + r_-^{k'}) + \sum_{n\in C(r(b))} \hat{\pi}_n(w_+^n + w_-^n) + \sum_{n\in N^-} \hat{\pi}_n \sum_{i\in S(A(n))} \theta_i'^{mid}(w_+^n + w_-^n)$$

$$K^+(k') = q\hat{\theta}_+^{k'} + t + r_+^{k'} + Ka(\hat{\theta}_+^{k'}) + Kb(\hat{\theta}_+^{k'}) + Kc(\hat{\theta}_+^{k'})$$

$$K^-(k') = -q\hat{\theta}_-^{k'} + t + r_-^{k'} + Ka(-\hat{\theta}_-^{k'}) + Kb(-\hat{\theta}_-^{k'}) + Kc(-\hat{\theta}_-^{k'})$$

$$Ka(a) = \sum_{\substack{m\in\mathcal{P}(r(b),k')\\ A(m)=0}} a(w_+^m - w_-^m), \quad Kb(b) = \sum_{\substack{n\in\mathcal{P}(r(b),k')\\ A(n)\neq 0}} b\{(1-\pi_n^+)w_+^n - (1-\pi_n^-)w_-^n\}$$

$$Kc(c) = \sum_{\substack{A(n)\in\mathcal{P}(r(b),k)\\ n\notin\mathcal{P}(r(b),k)}} \{c(\pi_n^- w_-^n - \pi_n^+ p w_+^n)\}$$

Proof. Recall that

$$g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) = \sum_{k\in\mathcal{L}_b} p^k \sum_{n\in\mathcal{P}(r(b),k)} g_t(s_{A(n)}, s_n, z_n).$$

Since p^k can be replaced by

$$p^k = \sum_{k'\in C(k)} \theta'_{k'},$$

the inner minimization problem of (4.71) is equivalent to solving following linear program:

$$\begin{aligned}
&\text{AUX}:\\
&\max \quad -\sum_{k'\in\mathcal{C}(\mathcal{L}_b)} \theta'_{k'} \sum_{n\in\mathcal{P}(r(b),k)} g_t(s_{A(n)}, s_n, z_n) - \sum_{k'\in\mathcal{C}(\mathcal{L}_b)} \theta'_{k'} Q_{k'}(w^l_{k'})\\
&s.t. \quad \theta'_{k'} = \theta'^{mid}_{k'} + \hat{\theta}'^{+}_{k'} z^+_{k'} - \hat{\theta}'^{-}_{k'} z^-_{k'} \quad k'\in\mathcal{C}(\mathcal{L}_b)\\
&\qquad \sum_{k'\in\mathcal{C}(\mathcal{L}_b)} \theta'_{k'} = 1\\
&\qquad \pi^-_n \sum_{i\in S(A(n))} \theta'_i \le \sum_{i\in S(n)} \theta'_i \quad \forall n\in N_b\cup C(\mathcal{L}_b)\setminus\{r(b)\}\\
&\qquad \sum_{i\in S(n)} \theta'_i \le \pi^+_n \sum_{i\in S(A(n))} \theta'_i \quad \forall n\in N_b\cup C(\mathcal{L}_b)\setminus\{r(b)\}\\
&\qquad \sum_{k'\in\mathcal{C}(\mathcal{L}_b)} z^+_{k'} + z^-_{k'} \le \frac{\Gamma}{|S|} * S_b\\
&\qquad 0\le z^+_{k'} \le 1 \quad k'\in\mathcal{C}(\mathcal{L}_b)\\
&\qquad 0\le z^-_{k'} \le 1 \quad k'\in\mathcal{C}(\mathcal{L}_b)
\end{aligned}$$

Then, the dual of AUX is

$$\begin{aligned}
&min \quad C'\\
&s.t. \quad K^+(k') \ge -\sum_{n\in\mathcal{P}(r(b),A(k'))} g_t(s^l_{A(n)}, s^l_n, z^l_n) - Q_{k'}\hat{\theta}^+_{k'} \quad \forall k\in\mathcal{L}_b, k'\in C(k)\\
&\qquad K^-(k') \ge \sum_{n\in\mathcal{P}(r(b),A(k'))} g_t(s^l_{A(n)}, s^l_n, z^l_n) + Q_{k'}\hat{\theta}^-_{k'} \quad \forall k\in\mathcal{L}_b, k'\in C(k)\\
&\qquad t\ge 0, r^{k'}_+ \ge 0, r^{k'}_- \ge 0, w^n_+ \ge 0, w^n_- \ge 0
\end{aligned}$$

We reinject this into the original objective function.
With ambiguous parameter $\tilde{d}$, the constraint (4.65) is equivalent to solving:

$$\sum_{i\in I} z^{tm}_{ij} \le \bar{d}^{tm}_j - \hat{d}^{tm}_j \times min(1, \Gamma^D_{tm}) \quad j\in J, tm\in N_b. \tag{4.72}$$

We denote a feasible set defined by constraints (4.66)-(4.70) and (4.72), all of which are linear constraints by $\tilde{X}^A_b(w_{A(r(b))}, d_{[(r(b),t_h]})$, which will replace $X^A_b(w_{A(r(b))}, d_{[(r(b),t_h]})$.
Finally, the resulting formulation is Q'_b as in the statement of the lemma. □

Next, we describe a modified version of nested decomposition which we use to solve blocked dynamic programming equations. We call this the Blocked Nested Decomposition algorithm. We first describe the algorithm for the nominal model, and extend it to the robust model.

Blocked Nested Decomposition

Similar to the ND algorithm, the Blocked Nested Decomposition (BND) algorithm consists of a forward and a backward step in each iteration. In contrast to the ND, which solves the problem for each node, in the BND we solve the following problem for each block b in each iteration l:

$$
\begin{aligned}
&(BND_b^l(w^l_{A(r(b))}, \psi^l_{C(\mathcal{L}_b)})\\
&\underline{Q}_b^l(w^l_{A(r(b))}, \psi^l_{C(\mathcal{L}_b)}) :=\\
&\qquad\qquad max \quad g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) + \sum_{k\in\mathcal{L}_b} p^k \sum_{\substack{k'\in C(k),\\ r(b')=k'}} \pi_{k'}\psi^l_{b'}(w^l_k)\\
&\qquad\qquad s.t. \quad (\mathbf{s}^l_{N_b}, \mathbf{x}^l_{N_b}, \mathbf{z}^l_{N_b}) \in X_b^A(w^l_{A(r(b))}, d_{[(r(b),t_b]})
\end{aligned}
$$

$$
\begin{aligned}
&\underline{Q}_1^l(w_0, \psi^l_{C(\mathcal{L}_b)}) :=\\
&\qquad\qquad max \quad g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) + \sum_{k\in\mathcal{L}_1} p^k \sum_{\substack{k'\in C(k),\\ r(b')=k'}} \pi_k\psi^l_{b'}(w^l_k)\\
&\qquad\qquad s.t. \quad (\mathbf{s}^l_{N_1}, \mathbf{x}^l_{N_1}, \mathbf{z}^l_{N_1}) \in X_1^B(w^l_{A(r(1))}, d_{[(r(1),t_1]})
\end{aligned}
$$

where the value function of block b' is approximated by:

$$
\begin{aligned}
\psi^l_{b'}(w^l_{A(r(b'))}) := max\{\phi_{b'} : \quad & \phi_{b'} \le B_{tm}\\
& \phi_{b'} \le (\beta^q_{tm} + \alpha^{Aq}_{tm} s_{A(r(b'))} + \alpha^{Bq}_{tm}\hat{x}_{A(r(b'))}), \forall q = 1..l-1\}
\end{aligned}
\tag{4.73}
$$

and $\psi^l_{C(\mathcal{L}_b)}$ is a set of $\psi^l_{b'}(w^l_{A(r(b'))})$ for all $k \in C(\mathcal{L}_b)$ and $r(b') = k$.

Then, the BND algorithm can be described as follows:
Step 1: Initialization. $LB \leftarrow -\infty$, $UB \leftarrow \infty$, $l \leftarrow 1$
For all $b \in \mathcal{B}$, initialize ψ^1_b, $\psi^1_b = \{\phi_b : \quad \phi_b \le B_{tm}\}$

Step 2: Forward Step.
For $b \in \mathcal{B}$, in ascending order,
solve $BND_b^l(w^l_{A(r(b))}, \psi^l_{C(\mathcal{L}_b)})$, collect solution $(\mathbf{s}^l_{N_b}, \mathbf{x}^l_{N_b}, \mathbf{z}^l_{N_b})$,
and update $g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)})$ and $g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)})$.

Step 3:

$$
\mathbf{LB} \leftarrow g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) + \sum_{b\in\mathcal{B}\setminus\{1\}} (\prod_{n\in\mathcal{P}(01,r(b))} \pi_n) g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)})
$$

Step 4: Backward Step.
For $b \in \mathcal{B}$, in descending order, do followings.
Solve $BND_b^l(w_{A(r(b))}^l, \psi_{C(\mathcal{L}_b)}^l)$, collect cut coefficients $(\beta_b^l, \alpha_b^{Al}, \alpha_b^{Bl})$, and add a cut (4.74) to ψ_b^l, and set this as ψ_b^{l+1}.

Step 5: Solve $BND_1^l(w_0, \psi_{C(\mathcal{L}_1)}^{l+1})$ and set **UB** be the optimal value.

Step 6: If some stopping criterion is satisfied, then stop.
Otherwise, $l \leftarrow l+1$ and go to Step 2.

For the robust model, we solve the following problem for each block b in each iteration l in the BND algorithm, and we denote this problem by $(BNDR_{tm}^l(w_{A(r(b))}^l, \psi_{C(\mathcal{L}_b)}^l))$.

$$\begin{aligned}
&\underline{Q}_b^l(w_{A(r(b))}^l, \psi_{C(\mathcal{L}_b)}^l) := \\
&\quad max \quad g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) + \sum_{\substack{k' \in \mathcal{C}(\mathcal{L}_b),\\ r(b')=k'}} \theta_{k'}^{mid} \psi_{b'}^l(w_{A(k)}^l) \\
&\quad s.t. \quad (\mathbf{s}_{N_b}, \mathbf{x}_{N_b}, \mathbf{z}_{N_b}) \in \tilde{X}_b^A(w_{A(r(b))}, d_{[(r(b),t_h]}) \\
&\qquad\quad K^+(k') \geq - \sum_{n \in \mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) - \psi_{b'}^l(w_{A(k)}^l)\hat{\theta}_{k'}^+ \quad \forall k' \in \mathcal{C}(\mathcal{L}_b) \\
&\qquad\quad K^-(k') \geq \sum_{n \in \mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) + \psi_{b'}^l(w_{A(k)}^l)\hat{\theta}_{k'}^- \quad \forall k' \in \mathcal{C}(\mathcal{L}_b) \\
&\qquad\quad t \geq 0, r_+^{k'} \geq 0, r_-^{k'} \geq 0, w_+^n \geq 0, w_-^n \geq 0
\end{aligned}$$

$$\begin{aligned}
&\underline{Q}_1^l(w_0, \psi_{C(\mathcal{L}_b)}^l) := \\
&\quad max \quad g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) + \sum_{\substack{k' \in \mathcal{C}(\mathcal{L}_b),\\ r(b')=k'}} \theta_{k'}^{mid} \psi_{b'}^l(w_{A(k)}^l) \\
&\quad s.t. \quad (\mathbf{s}_{N_1}^l, \mathbf{x}_{N_1}^l, \mathbf{z}_{N_1}^l) \in X_1^B(w_{A(r(1))}^l, d_{[(r(1),t_1]}) \\
&\qquad\quad K^+(k') \geq - \sum_{n \in \mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) - \psi_{b'}^l(w_{A(k)}^l)\hat{\theta}_{k'}^+ \quad \forall k' \in \mathcal{C}(\mathcal{L}_b) \\
&\qquad\quad K^-(k') \geq \sum_{n \in \mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) + \psi_{b'}^l(w_{A(k)}^l)\hat{\theta}_{k'}^- \quad \forall k' \in \mathcal{C}(\mathcal{L}_b) \\
&\qquad\quad t \geq 0, r_+^{k'} \geq 0, r_-^{k'} \geq 0, w_+^n \geq 0, w_-^n \geq 0
\end{aligned}$$

where $g_b(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) = \sum_{k \in \mathcal{L}(b)} \sum_{n \in \mathcal{P}(r(b),k)} \theta_{k'}^{mid} g_t(s_{A(n)}, s_n, z_n) - C'$, the value function of block b' is approximated by:

$$\begin{aligned}
\psi_{b'}^l(w_{A(r(b'))}^l) := max\{\phi_{b'} : \quad & \phi_{b'} \leq B_{tm} \\
& \phi_{b'} \leq (\beta_{tm}^q + \alpha_{tm}^{Aq} s_{A(r(b'))} + \alpha_{tm}^{Bq} \hat{x}_{A(r(b'))}), \forall q = 1..l-1\} \qquad (4.74)
\end{aligned}$$

and $\psi_{C(\mathcal{L}_b)}^l$ is a set of $\psi_{b'}^l(w_{A(r(b'))}^l)$ for all $k \in C(\mathcal{L}_b)$ and $r(b') = k$.

We follow the BND algorithm described above, but $BNDR^l_{tm}(w^l_{A(r(b))}, \psi^l_{C(\mathcal{L}_b)})$ is solved in each step instead of $BND^l_{tm}(w^l_{A(r(b))}, \psi^l_{C(\mathcal{L}_b)})$.

4.3.5 Stochastic Dual Dynamic Programming

Stochastic Dual Dynamic Programming (SDDP) is a sampling based nested decomposition method proposed in Pereira and Pinto (1991) that takes advantage of stage-wise independence. The scenario tree is stage-wise independent when any two nodes at the same stage share an identical set of child nodes. Under this assumption, the value functions and the expected profit-to-go functions depend only on the stage rather than the nodes, and thus there is a significant reduction in the number of DP equations. If the stochastic process is stage-wise dependent, it can typically be modeled using an auto-regressive process, and as a result, additional state variables are created to keep track of previous stochastic outcomes. If the stochastic variable is on the right-hand sides of constraints, a standard SDDP is valid because the value function is still convex with respect to state variables.

In our problem, we assume that demand is dependent on the outcomes of a series of Bernoulli trials. We restrict this setting in order to apply SDDP. Specifically, we require demand at time t to be related to the demand at time $t-1$ as well as the outcome of current trial. Then, demand can be modeled as follows.

$$d_t = \delta_t d_{t-1} + X_t h_t + \epsilon_t \tag{4.75}$$

where $X_t \sim Bernoulli(\pi_t)$, and $P(X_t = 1|X_{[t-1]}) = P(X_t = 1)$, and error vectors $\epsilon_1, ...\epsilon_t$ are independent of each other.

Although this demand model has its limitation, it is significantly more flexible than insisting on stage-wise independence. It also captures the basic assumption in our setting – in the scenario tree of our problem, when more positive outcomes are observed, more demand is expected. Also, the stochastic variable, the demand, only appears on the right-hand side of constraint (4.5) in **FL**, and thus we can use the standard SDDP algorithm for our nominal model. In particular, the Bellman equation for each node tm with demand model (4.75) is:

$$\begin{aligned}
Q_{tm}(w_{A(tm)}, d_{A(tm)}, X_{tm}, \epsilon_{tm}) :=& \\
max \quad & g_t(s_{A(tm)}, s_{tm}, z_{tm}) + \mathcal{Q}_{tm}(s_{tm}, x_{tm}, d_{tm}) \\
s.t. \quad & \sum_{i \in I} z^{tm}_{ij} \le \delta_t d_{A(tm)} + X_{tm} h_t + \epsilon_{tm} \quad j \in J \\
& d_{tm} = \delta_t d_{A(tm)} + X_{tm} h_t + \epsilon_{tm} \\
& (s_{tm}, x_{tm}, z_{tm}) \in X^A_{tm}(w_{A(tm)}, d_{tm})
\end{aligned}$$

Let $\xi_t := (X_t, \epsilon_t)$. Recall that in demand model (4.75), $P(X_t = 1|X_{[t-1]}) = P(X_t = 1)$, and $\epsilon_t \quad \forall t \le T$ are independent, and thus, $\xi_t \quad \forall t \le T$ are independent. Let E_t be a set of realizations of ξ_t (in our case, there are two realizations), and the probability of each outcome defined as π_{tn} where $n = 1..|E_t|$.

We define state variable $\acute{w}_t := (s_t, x_t, d_t)$. Then, the DP equations and the expected profit-to-go functions depend on the stage rather than the nodes, and a DP equation per stage is:

$$
\begin{aligned}
&Q_t(\acute{w}_{t-1}, \xi_t) := \\
&\qquad max \quad g_t(s_{t-1}, s_t, z_t) + \mathcal{Q}_{t+1}(\acute{w}_t) \\
&\qquad s.t. \quad \sum_{i \in I} z_{ij}^t \leq \delta_t d_{t-1} + X_t h_t + \epsilon_t \quad j \in J \\
&\qquad\qquad d_t = \delta_t d_{t-1} + X_t h_t + \epsilon_t \\
&\qquad\qquad (s_t, \hat{x}_t, z_t) \in X_t^A(w_{t-1}, d_t).
\end{aligned}
\tag{4.76}
$$

Now, we describe the standard SDDP proposed in Pereira and Pinto (1991) when applied to solve **FL**.

The SDDP algorithm consists of a forward step, where sample paths are generated from the stochastic process and solutions are updated, and a backward step, where we improve the approximation of the profit-to-go functions in each iteration. In each iteration l, in the forward step, we start by sampling K scenarios from the tree. Next, from $t = 0$ to T, we solve the following problem for each k. In particular, for each k at stage t given a state variable $\acute{w}_{t-1}^{kl}$, the problem $SDDP_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_t^l)$ for a particular realization of ξ_t (denoted by ξ_t^k where $\xi_t^k \in E_t$) is defined as:

$$
\begin{aligned}
&(SDDP_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_t^l)) \\
&\quad \underline{Q}_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_t^l) := \\
&\qquad max \quad g_t(s_{t-1}^{kl}, s_t^{kl} z_t^{kl}) + \psi_t^l(\acute{w}_t^l) \\
&\qquad s.t. \quad \sum_{i \in I} z_{ijt}^{kl} \leq \delta_t d_{t-1}^k + X_t^k h_t + \epsilon_t^k \quad j \in J \\
&\qquad\qquad d_t^k = \delta_t d_{t-1}^k + X_t^k h_t + \epsilon_t^k \\
&\qquad\qquad (s_t^{kl}, x_t^{kl}, z_t^{kl}) \in X_t^A(w_{t-1}^{kl}, d_t^k)
\end{aligned}
$$

where expected profit-to-go function ψ_t^l is defined as

$$
\begin{aligned}
\psi_t^l(\acute{w}_t) := \max\{\phi_t : &\phi_t \leq B_t \\
&\phi_t \leq \sum_{m \in E_t} \pi_{(t+1)m}(\beta_{t+1}^{mq} + (\alpha_{t+1}^{mq})^T \acute{w}_t) \quad \forall q = 1, .., l-1\}
\end{aligned}
$$

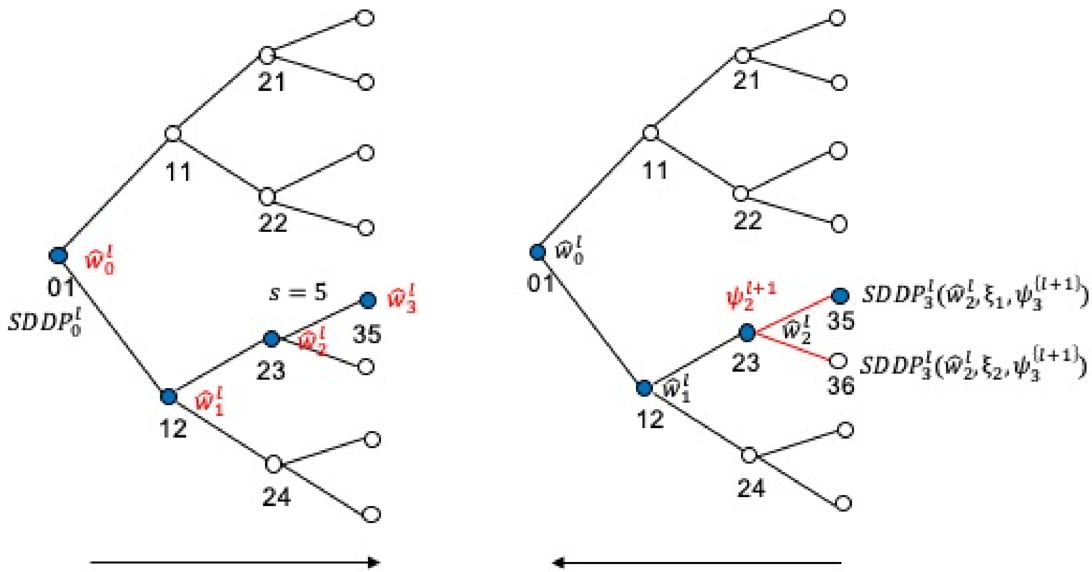
$(s_t^{kl}, x_t^{kl}, z_t^{kl})$, which is an optimal solution to the problem $SDDP_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_t^l)$, is given as a state variable $\acute{w}_t^{kl}$ to the problem $SDDP_{t+1}^l(\acute{w}_t^{kl}, \xi_t^k, \psi_t^l)$ of stage $t+1$ for scenario k.

After all the forward problems on the sampled paths are solved in iteration l, we construct a probabilistic lower bound on the optimal value to the problem **FL** with confidence $1 - \alpha$ as

$$
\mathbf{LB:=}\hat{\mu} - z_{\alpha/2} \frac{\hat{\sigma}}{\sqrt{K}}
$$

where $o^k = \sum_{t=0}^T g_t(s_{t-1}^{lk}, s_t^{lk}, z_t^{lk})$, $\hat{\mu} = 1/K \sum_{k=1}^K o^k$, $\hat{\sigma^2} = 1/(K-1) \sum_{k=1}^K (o^k - \hat{\mu})^2$ as proposed in Zou et al. (2018).

The backward step starts from the stage T. The goal of the backward step is to update the expected profit to-go function at each stage by adding a new cut. In particular, in stage t



(a) Forward **(b)** Backward

Figure 4.3. Illustration of Stochastic Dual Dynamic Programming

for scenario k, given a candidate solution $\acute{w}_{t-1}^{kl}$, we solve $SDDP_t^l(\acute{w}_{t-1}^{kl}, \xi_t^n, \psi_t^{l+1})$ for all $n \in E_t$ and collect cut coefficients $(\beta_t^{nl}, \alpha_t^{nl})$. Then the following cut is added to ψ_{t-1}^l to get ψ_{t-1}^{l+1}.

$$\phi_{t-1} \leq \sum_{m \in E_t} \pi_{tm}(\beta_t^{nl} + (\alpha_t^{nl})^T \acute{w}_{t-1}$$

Figure 4.3 illustrates the forward and backward steps in the l^{th} iteration of SDDP. In the forward step, scenario 5 is sampled, and $SDDP_t^l$ problems at nodes 01 to 35 in scenario 5 are solved in iteration l and solutions are updated. In the backward step of this example, $SDDP_3^l$ problems at 35 and 36 are solved to generate a cut which adds to ψ_2^l.

The linear cuts added in backward steps approximate the true expected profit-to-go function, and thus the optimal value of the problem at the root node provides an upper bound, **UB**. However, we can only get a probabilistic lower bound, the validity of which is guaranteed with a certain probability given that K is not too small, and thus it is possible that **LB** $\geq$ **UB** even for big K. As a result, careful consideration is required when choosing a stopping criterion. As suggested in the literature, we stop the algorithm when the upper bound becomes stable, and the probabilistic lower bound using large sample size K is close to the upper bound.

We summarize the SDDP algorithm below:

Step 1: Initialization. $LB \leftarrow -\infty$, $UB \leftarrow \infty$, $l \leftarrow 1$
For all $t \leq T$, initialize ψ_t^1, $\psi_t^1 = \{\phi_t : \quad \phi_t \leq B_t\}$

Step 2: Sample K scenarios $\Sigma^l = \{\xi_0^l, ...\xi_T^l\}$

Step 3: Forward Step.

For $t = 0..T$ and $k = 1..K$,
Solve $(SDDP_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_t^l))$, collect solution $(\hat{x}_t^{kl}, s_t^{kl}, z_t^{kl})$,
and update $g_0(x_0^{kl}, s_0^{kl}, z_0^{kl})$ and $g_t(s_{t-1}^{kl}, s_t^{kl}, z_t^{kl})$.

Step 4:
$o^k \leftarrow \sum_{t=0}^{T} g_t(s_{t-1}^{lk}, s_t^{lk}, z_t^{lk})$, $\hat{\mu} \leftarrow 1/K \sum_{k=1}^{K} o^k$, $\hat{\sigma^2} \leftarrow 1/(K-1) \sum_{k=1}^{K} (o^k - \hat{\mu})^2$

$$\mathbf{LB} \leftarrow \hat{\mu} - z_{\alpha/2} \frac{\hat{\sigma}}{\sqrt{K}}$$

Step 5: Backward Step.
For $t = T..1$, and for $k = 1..K$ repeat followings.
For $n = 1..2$, solve $(SDDP_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_t^l))$, and collect cut coefficients $(\beta_t^{nl}, \alpha_t^{nl})$
Add a following cut to ψ_{t-1}^l, and set this as ψ_{t-1}^{l+1}.

$$\phi_{t-1} \leq \sum_{m \in E_t} \pi_{tm}(\beta_t^{nl} + (\alpha_t^{nl})^T \acute{w}_{t-1}$$

Step 6: Solve $SDDP_0^l(w_0, \xi_0^1, \psi_0^{l+1})$ and set **UB** be the optimal value.

Step 7: If some stopping criterion is satisfied, then stop.
Otherwise, $l \leftarrow l + 1$ and go to Step 2.

Blocked Stochastic Dual Dynamic Programming for a Model with Ambiguities in Probabilities and Demand

We now extend the SDDP approach to our problem with ambiguities in probabilities and demand. In addition, we borrow a concept from Blocked ND for SDDP. We call this Blocked Stochastic Dual Dynamic Programming (BSDDP).

Specifically, we divide the planning horizon into $[0, t_1]$, $[t_1 + 1, t_1 + 2]$, ... and $[T - 1, T]$, where a sub-tree (block) in the first time interval is the only one with multiple periods. Recall that the DP equation for the first block with ambiguous parameters is as follows:

$$\begin{aligned} \tilde{Q}_{b=1} := max \min_{\substack{\tilde{d}_{[(r(1),t_1]} \in \mathcal{D}, \\ \theta' \in \Theta_2}} \quad & g_1(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) + \sum_{k \in \mathcal{L}_b} \sum_{\substack{n \in C(k), \\ r(b')=n}} \theta'_n Q_{b'}(w_k) \\ s.t. \quad & (\mathbf{s}_{N_1}, \mathbf{x}_{N_1}, \mathbf{z}_{N_1}) \in X_1^B(w_0, d_{[(r(1),t_h]}) \end{aligned}$$

For $b > 1$, instead of considering Q_b, we consider a DP equation per stage as shown in previous section:

$$Q_t(\acute{w}_{t-1}, \xi_t) := \tag{4.77}$$
$$\begin{aligned} max \quad & g_t(s_{t-1}, s_t, z_t) + \mathcal{Q}_{t+1}(\acute{w}_t) \\ s.t. \quad & \sum_{i \in I} z^t_{ij} \leq \delta_t d_{t-1} + X_t h_t + \epsilon_t \quad j \in J \\ & d_t = \delta_t d_{t-1} + X_t h_t + \epsilon_t \\ & (s_t, \hat{x}_t, z_t) \in X^A_t(w_{t-1}, d_t). \end{aligned}$$

We incorporate ambiguous parameters to Q_t, the value function (4.77), as we did in section 4.3.3, and then the DP equation with ambiguous parameters for stage t is:

$$Q_t(\acute{w}_{t-1}, \xi_t) :=$$
$$\begin{aligned} max \min_{\tilde{d}_t \in \mathcal{D}} \quad & g_t(s_{t-1}, s_t, z_t) + Q_{t+1}(\acute{w}_t, \xi^2_t) + \pi^+_{(t+1)1}\gamma^+_t - \pi^-_{(t+1)1}\gamma^-_t \\ s.t. \quad & \sum_{i \in I} z^t_{ij} \leq \delta_t d_{t-1} + X_t h_t + \epsilon_t \quad j \in J \\ & d_t = \delta_t d_{t-1} + X_t h_t + \epsilon_t \qquad (4.78) \\ & (s_t, \hat{x}_t, z_t) \in X^A_t(w_{t-1}, d_t) \\ & \gamma^+_{tm} - \gamma^-_{tm} \geq Q_{t+1}(\acute{w}_t, \xi^1_t) - Q_{t+1}(\acute{w}_t, \xi^2_t). \end{aligned}$$

When we incorporate $\tilde{d}_t$, the equality constraint (4.78) becomes infeasible. Instead, in the BSDDP algorithm below, we sample some $j \in J$ at time stage t, and let the demand be equal to its worst values while demand for other j is at its nominal values.

We describe the changes for our BSDDP algorithm for the robust model from SDDP algorithm below. We first present the problem, denoted by $BSDDP^l_t(\acute{w}^{kl}_{t-1}, \xi^k_t, \psi^l_{t+1})$, which will be solved in each forward and backward step in each iteration.

$$\underline{Q}^l_1(w_0, \psi^l_{t_1+1}) :=$$
$$\begin{aligned} max \quad & g_0(\mathbf{x}_{N(1)}, \mathbf{s}_{N(1)}, \mathbf{z}_{N(1)}) + \sum_{k' \in \mathcal{C}(\mathcal{L}_b)} \theta^{mid}_{k'} \psi^l_{t_1+1,n,e(k')}(w^l_{A(k')}) \\ s.t. \quad & (\mathbf{s}^l_{N_1}, \mathbf{x}^l_{N_1}, \mathbf{z}^l_{N_1}) \in X^B_1(w^l_{A(r(1))}, d_{[(r(1),t_1]}) \\ & K^+(k') \geq - \sum_{n \in \mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) - \psi^l_{t_1+1,e(k')}(w^l_{A(k)})\hat{\theta}^+_{k'}, \forall k' \in \mathcal{C}(\mathcal{L}_b) \\ & K^-(k') \geq \sum_{n \in \mathcal{P}(r(b),A(k'))} g_t(s_{A(n)}, s_n, z_n) + \psi^l_{t_1+1,e(k')}(w^l_{A(k)})\hat{\theta}^-_{k'} \quad \forall k' \in \mathcal{C}(\mathcal{L}_b) \\ & t \geq 0, r^{k'}_+ \geq 0, r^{k'}_- \geq 0, w^n_+ \geq 0, w^n_- \geq 0 \end{aligned}$$

$$
\begin{aligned}
&\underline{Q}_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_{t+1}^l) := \\
&\qquad max \quad g_t(s_{t-1}^{kl}, s_t^{kl} z_t^{kl}) + \psi_{t+1,2}^l(\acute{w}_t^{kl}) + \pi_{(t+1)1}^+ \gamma_{t+1}^{+kl} - \pi_{(t+1)1}^- \gamma_{t+1}^{-kl} \\
&\qquad s.t. \quad \sum_{i \in I} z_{ijt}^{kl} \leq \phi_t d_{t-1}^k + X_t^k h_t + \epsilon_t^k \quad j \in J \\
&\qquad\qquad d_t^k = \phi_t d_{t-1}^k + X_t^k h_t + \epsilon_t^k \\
&\qquad\qquad \gamma_{tm}^+ - \gamma_{tm}^- \geq \psi_{t+1,1}^l(\acute{w}_t^{kl}) - \psi_{t+1,2}^l(\acute{w}_t^{kl}) \\
&\qquad\qquad (s_t^{kl}, x_t^{kl}, z_t^{kl}) \in X_t^A(w_{t-1}^{kl}, d_t^k)
\end{aligned}
$$

where $g_0(\mathbf{x}_{N(b)}, \mathbf{s}_{N(b)}, \mathbf{z}_{N(b)}) = \sum_{k \in \mathcal{L}(1)} \sum_{n \in \mathcal{P}(r(b),k)} \theta_{k'}^{mid} g_t(s_{A(n)}, s_n, z_n) - C'$, and the profit to-go function of stage t and a realization $n \in E_t$ is approximated by:

$$
\begin{aligned}
\psi_{t,n}^l(\acute{w}_{t-1}) := \max\{\phi_{t,n} :&\phi_{t,n} \leq B_t \\
&\phi_{t,n} \leq \beta_{t,n}^q + (\alpha_{t,n}^q)^T \acute{w}_{t-1} \quad \forall q = 1, .., l-1\}
\end{aligned}
$$

and ψ_t^l is a set of $\psi_{t,n}^l(\acute{w}_{t-1})$ for all $n \in E_t$.

Note that for $t \leq t_1$, we solve a single multi-period problem instead of several single period problems. When we solve $BSDDP_t^l$ for $t \geq t_1 + 1$, we incorporate ambiguous parameter d into our algorithm in following way. Recall that $\tilde{d}_j$ belongs to an interval $[d_j^-, d_j^+]$, and demand for all $j \in J$ at node tm is constrained by a budget of uncertainty Γ_{tm}^D. In iteration l, at time stage t, scenario k, we sample Γ_t^D values of j from J. We set the demand of all sampled j to their worst values ($d_{jk}^t = d_{jk}^{t-}$), and for other j we set the demand at their nominal values ($d_{jk}^t = \bar{d}_{jk}^t$). Then $BSDDP_t^l$ for $t \geq t_1 + 1$ becomes a linear program.

We summarize the BSDDP algorithm below:

Step 1: Initialization. $LB \leftarrow -\infty$, $UB \leftarrow \infty$, $l \leftarrow 1$
For all $t \leq T$, $n \in E_t$ initialize $\psi_{t,n}^1$, $\psi_{t,n}^1 = \{\phi_{t,n} : \quad \phi_{t,n} \leq B_t\}$

Step 2: Sample K scenarios $\Sigma^l = \{\xi_0^l, ...\xi_T^l\}$
Sample $M = \Gamma_t$ customers for each t and k, $\Delta_{tk} = \{j_1^t, ...j_M^t\}$

Step 3: Forward Step.
For $t = 0$,
Solve $(BSDDP_0^l(w_0, \psi_{t_1+1}^l))$, collect solution, and update $g_1(\mathbf{x}_{N(1)}^l, \mathbf{s}_{N(1)}^l, \mathbf{z}_{N(1)}^l)$
For $t = t_1 + 1, .., T$ and $k = 1..K$,
For $j \in J$,
If $j \in \Delta_{tk}$, set $d_{jk}^t = d_{jk}^{t-}$. Else, set $d_{jk}^t = \bar{d}_{jk}^t$
Solve $(BSDDP_t^l(\acute{w}_{t-1}^{kl}, \xi_t^k, \psi_{t+1}^l))$, collect solution $(\hat{x}_t^{kl}, s_t^{kl}, z_t^{kl})$,
and update $g_t(s_{t-1}^{kl}, s_t^{kl}, z_t^{kl})$.

Step 4:
$o^k \leftarrow \sum_{t=0}^T g_t(s_{t-1}^{lk}, s_t^{lk}, z_t^{lk})$, $\hat{\mu} \leftarrow 1/K \sum_{k=1}^K o^k$, $\hat{\sigma^2} \leftarrow 1/(K-1) \sum_{k=1}^K (o^k - \hat{\mu})^2$

$$\mathbf{LB} \leftarrow g_1(\mathbf{x}^l_{N(1)}, \mathbf{s}^l_{N(1)}, \mathbf{z}^l_{N(1)}) + \hat{\mu} - z_{\alpha/2}\frac{\hat{\sigma}}{\sqrt{K}}$$

Step 5: Backward Step.
For $t = T..t_1+1$, and for $k = 1..K$ repeat the following:
For $n = 1..2$, do the following:
For $j \in J$,
If $j \in \Delta_{tk}$, set $d^t_{jk} = d^{t-}_{jk}$. Else, set $d^t_{jk} = \bar{d}^t_{jk}$,
Solve $(BSDDP^l_t(\acute{w}^{kl}_{t-1}, \xi^n_t, \psi^{l+1}_{t+1}))$, and collect cut coefficients $(\beta^l_{t,n}, \alpha^l_{t,n})$
Add a following cut to ψ^l_t, and set this as ψ^{l+1}_t.

$$\phi_{t,n} \leq (\beta^l_{t,n} + (\alpha^l_{t,n})^T \acute{w}_{t-1}$$

Step 6: Solve $(BSDDP^l_0(w_0, \psi^{l+1}_{t_1+1}))$ and set **UB** be the optimal value.

Step 7: If some stopping criterion is satisfied, then stop.
Otherwise, $l \leftarrow l+1$ and go to Step 2.

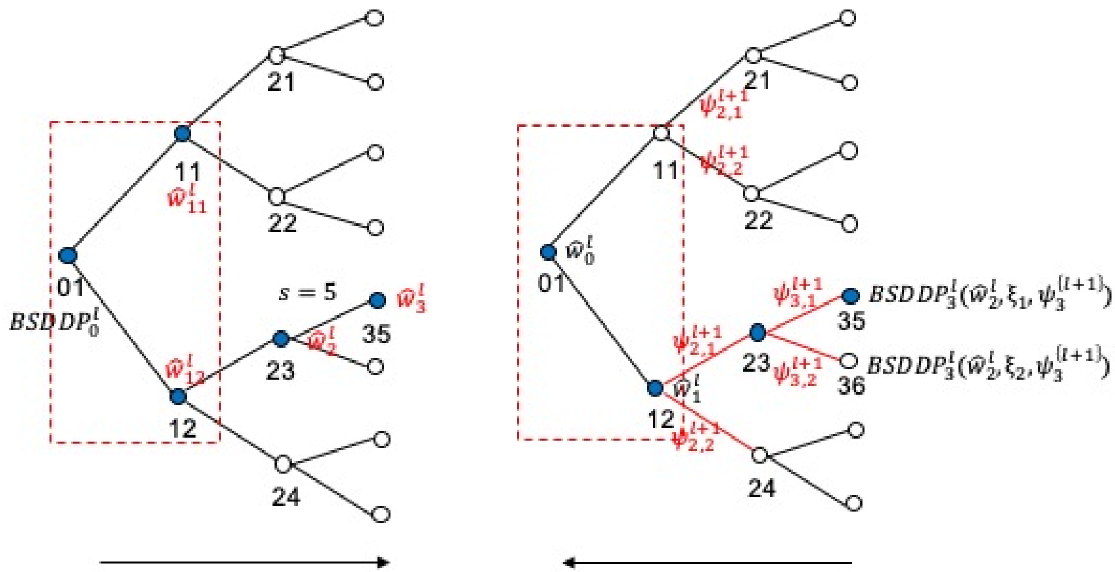
Figure 4.4 illustrates the forward and backward steps in the l^{th} iteration of BSDDP. In the forward step, the problem $BSDDP^l_{b=1}$ from $t = 0$ to 1 is solved, then for sampled scenario 5, $SDDP^l_t$ problems at nodes 23 and 35 are solved, and then solutions are updated. In the backward step of this example, $BSDDP^l_3$ problems at 35 and 36 are solved to generate cuts which are added to $\psi^l_{3,1}$ and $\psi^l_{3,2}$. Then these are set as $\psi^{l+1}_{3,1}$ and $\psi^{l+1}_{3,2}$. At $t = 2$, cuts for $\psi^l_{2,1}$ and $\psi^l_{2,2}$ are added to get $\psi^{l+1}_{2,1}$ and $\psi^{l+1}_{2,2}$. For problem $BSDDP^l_0$, $\psi^{l+1}_{2,1}$ and $\psi^{l+1}_{2,2}$ are used for all child nodes 11 and 12.

4.4 Computational Study

In the previous sections, we propose a MILP model with ambiguous success probabilities and demand, as well as heuristic approaches to solve this problem. Here, we conduct a computational study to answer the following questions:

- What is the impact of explicitly modeling ambiguous probabilities of successes and demand in this class of planning models? When is it worth doing so? Are there any insights about the decisions made by robust model in terms of the number of facilities, the capacity of open facilities, amount of covered demand, and amount of unused capacity?

- How large are the models that we can solve to optimality using commercial optimization software? How effective are the proposed heuristics for instances that are too large to solve to optimality?



(a) Forward

(b) Backward

Figure 4.4. Blocked SDDP

We implement and test our models on a desktop computer with a Core i7 3.4 GHz processor and 16 GB of RAM. We code all models using Python and solve them using CPLEX, version 12.6, from IBM.

4.4.1 When Is It Worth Explicitly Modeling Ambiguities in Probability and Demand?

In this section, we analyze the decisions made by our model in Section 4.2 to explore the impact of modeling ambiguous successes probabilities and demand on decision strategy. To perform this analysis, we run the following experiments. The test instances are chosen to capture various decision-making scenarios in this setting.

We employ a 5 period of planning horizon, 15 facility locations being 15, and 15 demand locations for these test problems. We randomly generate 15 and 15 points on a square representing the candidate facility locations and customer locations, respectively. More specifically, let (x, y) be a (longitude, latitude) pair indicating a point on a square with a side length of 100. For each point, we draw a value for x and y from $Unif(0, 100)$ and $Unif(0, 100)$ distributions, respectively. The distance between a facility i and a customer j, denoted by $dist_{ij}$ is computed as Euclidean distance.

For each demand location, the nominal demand for the first period ($t = 0$) is 0. The demand evolves based on the following model,

$$d_t = \delta_t d_{t-1} + X_t h_t + \epsilon_t,$$

and we draw a value for h_t and ϵ_t from $Unif(1000, 20000)$ and $Unif(0, 50)$ distributions, respectively. Each Bernoulli trial has a probability of success in the range $(0.1, 0.9)$, and for nominal models, we assume the center of that range, 0.5.

To assess the sensitivity of results to parameters, we consider the following parameter settings: we draw a value for fixed facility construction (f_i) from various interval of uniform distribution as follows: For each facility i, $f_i \sim \{Unif(20000, 30000)$, $\mathbf{Unif(30000, 40000)}$, $Unif(40000, 50000)$, $Unif(50000, 60000)\}$. Capacity expansion cost per unit in the initial period (c_i^0) is randomly generated from uniform distributions as follows: For each i, $c_i^0 \sim \{Unif(0.1, 0.2)$, $\mathbf{Unif(0.5, 0.6)}$, $Unif(1, 1.1)$, $Unif(1.5, 1.6)\}$. Cost per unit of adding capacity (c_i^t) for $t >= 1$ is increased by 10%. For the product price per unit (r^t) and the transportation cost (v_{ij}^t), we keep parameters constant over time, so we drop the subscript t in the subsequent explanation. r is randomly generated from various interval of uniform distribution as follows: $r \sim \{Unif(1, 1.5)$, $\mathbf{Unif(1.5, 2)}$, $Unif(2, 2.5)$, $Unif(2.5, 3)\}$. Transportation cost v_{ij} is computed by $\nu \times dist_{ij}$, and we vary $\nu = \{0.005$, $\mathbf{0.01}$, 0.02, $0.03\}$.

In our experiments, to generate test cases, we vary each parameter in turn, and keep other parameters at their **boldfaced** base setting from the lists above. In addition, we vary the level of budget of uncertainty. The threshold value for the budget of uncertainty of probabilities Γ varies as follows: $\Gamma(\%) = 0, 10, 20, 30, 40, 50$, where $\Gamma = |S| \times \Gamma(\%)/100$, and S is a set of scenarios. The threshold value for the budget of uncertainty of demand Γ_{tm}^D varies as follows: $\Gamma_{tm}^D(\%) = 0, 20, 60, 100$, where $\Gamma_{tm}^D = |J| \times \Gamma_{tm}^D(\%)/100$, and J is a set of customers. In what follows, when we refer Γ and Γ_{tm}^D, we are referring to $\Gamma(\%)$ and $\Gamma_{tm}^D(\%)$, respectively.

For each set of test parameters, we generate 30 problem instances, and we solve the model to optimality with various level of budget of uncertainty. We then simulate 30 sample scenario trees with resolved probability uncertainty to find the distribution of outcomes. Thus, 900 simulation runs are considered in total for each test set.

We run separate experiments for ambiguities in probabilities and demand to observe the impact each has on decisions more clearly. Specifically, when we consider ambiguities in probabilities, for each simulation run, the (resolved) success rate of each node π_n is randomly generated from $Unif(0.1, 0.9)$ in our tests. We then calculate the cumulative distribution function (CDF) of profits for each of the simulated scenario trees, and then the average of the 10th, 20th, 30th, 50th, 70th and 90th percentile of profits over all simulation runs.

Similarly, when we consider ambiguities in demand, for each simulation run, the resolved demand at each node in the scenario tree is randomly generated from a uniform distribution described above for each node in each problem instance. To evaluate the solutions of models using various ranges of Γ_{tm}^D, we fix investment decisions (opening facility and amount of capacity expansion) for all nodes using each of the solutions, and resolve the nominal model for each of the simulated scenario trees. We then calculate the average expected profit over all simulation runs and the CDF of the profits for each of the simulated scenario trees, and then 10th, 20th, 30th, 50th, 70th and 90th percentile over all simulation runs.

We report the number of open facilities and the expected total capacity in the last period averaged over all simulation runs for various level of uncertainty and parameters. We also report the demand coverage and the amount of wasted capacity.

In Figure 4.5-4.7, we compare the average percentiles for nominal ($\Gamma = 0$) versus robust ($\Gamma > 0$) models in our setting with ambiguous event probabilities, in an attempt to characterize settings where robust models add value. Each percentile profit for the model with various Γ is expressed as a percentage of nominal model profit.

In Figure 4.5, we report the average percentage of 10th, 20th, 30th, 50th, 70th and 90th percentile profit for the model with various Γ as c_i^0 varies.

As shown in Figure 4.5, the 10th percentile of profit increases as Γ increases, and the increase in rate is larger when c_i^0 is higher. Since our model maximizes the profit, the 10th percentile of profit captures "bad" scenarios. Thus, if we observe higher profit in a low percentile, the solution is better protected against worse cases. However, the $50, 70$, and 90th percentile profits decrease as Γ increases. One interesting observation is that the 10th percentile profit is slightly better in solutions with a large Γ than in solution with a small Γ. However, higher percentile profits ($50, 70$th and 90th) are much worse in the same comparison. For example, the average 10th percentile profit is only 0.2% worse in the solutions with $\Gamma = 40$ than in the solutions with $\Gamma = 50$. However, the average 90th percentile profit is 12% better when using the solutions with $\Gamma = 40$ than when using the solutions with $\Gamma = 50$. As expected, the observation implies that there is a trade-off between overall profit and the worst-case (or worse case) profit depending on the value of the budget of uncertainty (Γ). It also shows that the small increase of Γ when Γ is large, may lead to a much conservative solution, and thus a value between 10 and 40 is better to be chosen for Γ based on what the primary concern (overall profit or the worst-case) is.

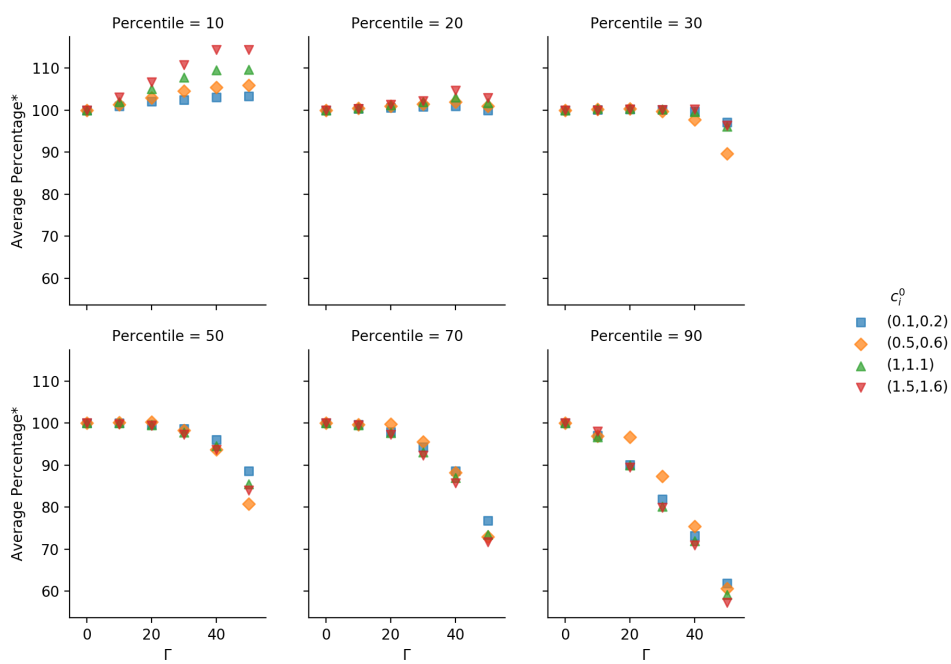


Figure 4.5. Average percentage of robust model to nominal model in $10, 20, 30, 50, 70$ and 90th percentile profit when varying Γ and capacity expansion cost(c_i^0)

The solutions to robust models tend to increase the profit of low demand paths to maximize the worst-case expected profit, and this is achieved by reducing excess capacity on low demand paths. When the capacity expansion cost is higher, the excess capacity impacts the profitability more significantly, and we observe that it is worth using a robust model in settings with high capacity expansion cost in order to achieve lower risk. On the other hand, overly conservative solutions hurt the profitability of high demand paths significantly by expanding capacity in higher cost or not fulfilling more demand. A budget of uncertainty

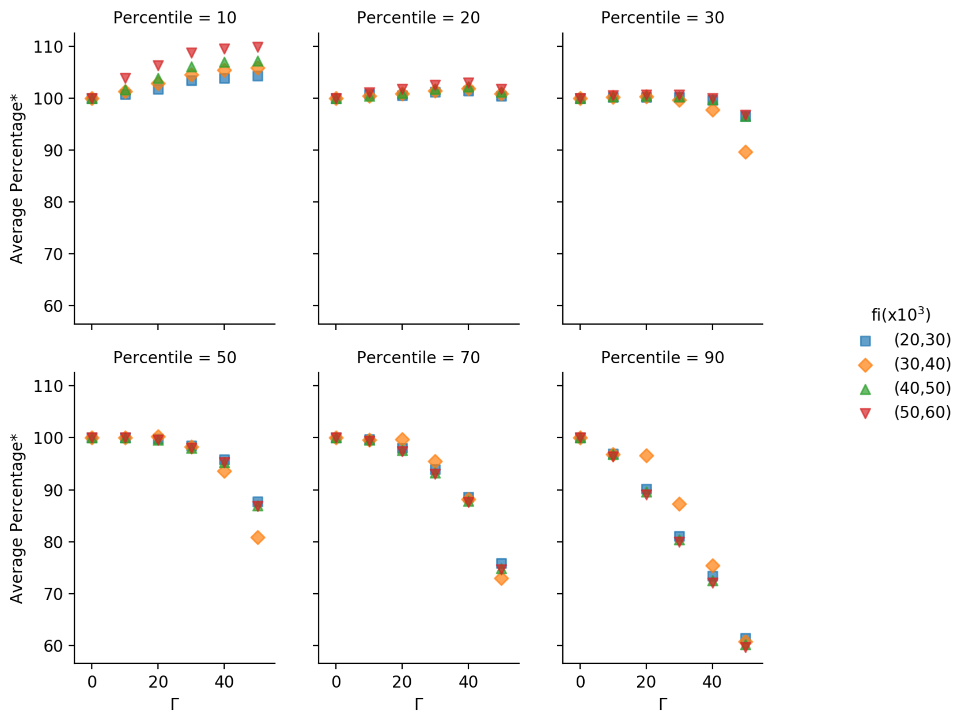


Figure 4.6. Average percentage of robust model to nominal model in 10, 20, 30, 50, 70 and 90th percentile profit when varying Γ and fixed facility opening cost(f_i)

can mitigate this. We observe the similar tendencies when varying the fixed facility opening cost, in Figure 4.6.

This is also supported by the results shown in Table 4.2, where we report the average number of open facilities, total capacity, proportion of covered demand, and proportion of unused capacity when varying Γ when varying c_i^0 and f_i. As expected, as investment cost increases, the number of open facilities and the total capacity decrease. As f_i varies, the number of open facilities changes significantly, but the total capacity built, the covered demand, and the unused capacity portion only change slightly. On the other hand, when c_i^0 varies, the number of open facilities changes slightly, but the total capacity built, the covered demand portion, and the unused capacity portion are changed more significantly.

In Table 4.2, we also observe that the average number of open facilities and total capacity decreases as Γ increases. This is not surprising because more conservative solutions are expected for large Γ, and thus less capacity is installed. Notice that more demand is covered in solutions with small Γ, but capacity is more effectively used in solutions with large Γ.

When the price of product r is varied, as shown in Figure 4.7, the 10th percentile of profit increases as Γ increases, and the increase in rate is larger when r is lower. However, the 50, 70, and 90th percentile profits decrease as Γ increases. When the unit price of product is low, marginal profit is low. All demand is more likely fulfilled on low demand paths for many possible decisions, and thus the unused capacity impacts the profitability more significantly on these low demand paths. Therefore, as observed in results, the robust model is more effective for lowering the risk when the price is low.

Table 4.2. Number of open facilities, total capacity, proportion of covered demand, and proportion of unused capacity, averaged over 30 problem instances and over a 30 sample scenario trees when varying Γ and capacity expansion cost(c_i^0) and opening facility cost(f_i)

	Γ	c_i^0				$f_i(\times 10^3)$			
	(%)	[.1,.2]	[.5,.6]	[1,1.1]	[1.5,1.6]	(20,30)	(30,40)	(40,50)	(50,60)
Average Number of Open Facilities	0	3.6	3.63	3.47	3.4	4.2	3.63	3.2	3
	10	3.4	3.4	3.23	3.2	4	3.4	3.03	2.73
	20	3.13	3.1	3.03	3.0	3.8	3.13	3.17	2.8
	30	3.1	3.06	2.87	2.87	3.63	3.06	2.97	2.67
	40	2.9	2.9	2.73	2.7	3.53	2.9	2.87	2.63
	50	2.8	2.83	2.7	2.6	3.37	2.83	2.83	2.57
Average Total Capacity	0	408	408.1	357.6	331.7	408.1	408.1	407.4	407.4
	10	398.8	394.3	340.6	318.8	395.8	394.3	393.6	392.5
	20	380.9	376.3	328.5	306	377.1	376.3	374.4	373.6
	30	359.6	345.6	300.1	284.2	349.7	345.6	346.6	342.9
	40	340	333.3	290.7	272.2	333.7	333.3	329.8	326.8
	50	318.8	307.5	262.8	235.9	310	307.5	304.6	297.6
Average Covered Demand (%)	0	83.3	83.3	81.5	80.7	83.3	83.3	83.3	83.3
	10	82.9	82.8	80.8	80.1	82.9	82.8	82.8	82.8
	20	82.1	81.8	79.8	78.7	81.9	81.8	81.8	81.7
	30	80.5	79.9	77.8	76.7	80.1	79.9	79.9	79.7
	40	79.1	79.1	77	75.7	79.1	79.1	78.9	78.7
	50	76.6	75.7	72.7	70	75.9	75.7	75.6	75.1
Average Unused Capacity (%)	0	30.2	30.3	28.5	24.6	30.3	30.3	30.3	30.3
	10	28.2	28.2	26.9	23.8	28.2	28.2	28.2	28.2
	20	26.7	27.3	25.8	23.7	27.3	27.4	27.3	27.2
	30	22.6	23.6	23.5	22.3	23.5	23.4	23.6	23.4
	40	19.7	20	19.4	17.4	20.7	20.5	19.9	19.8
	50	21.5	19.8	19.3	17.9	20.7	20.3	19.9	19.9

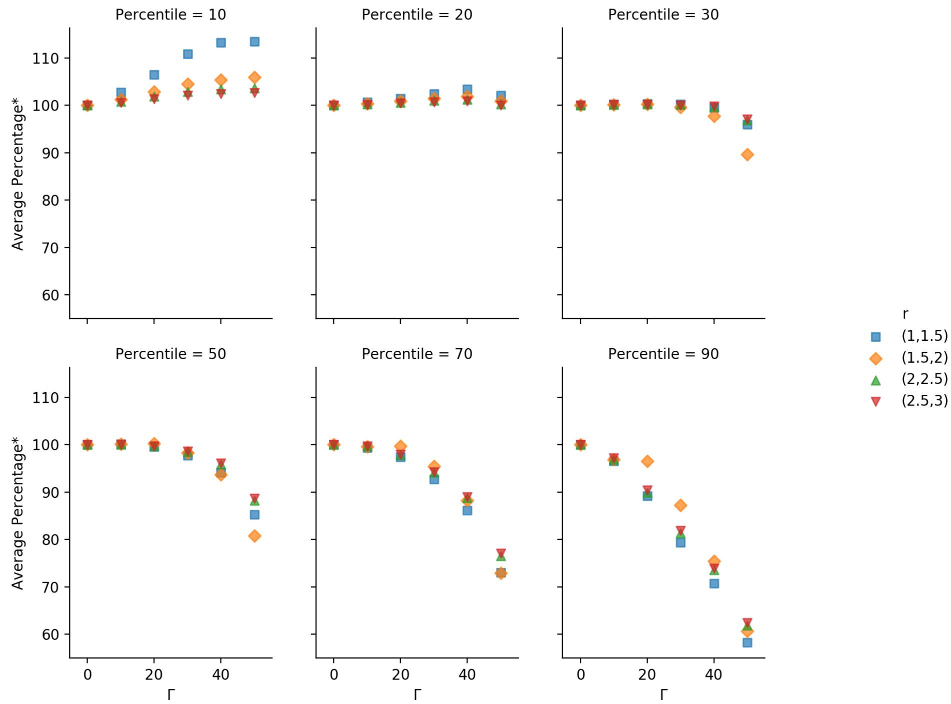


Figure 4.7. Average percentage of robust model to nominal model in $10, 20, 30, 50, 70$ and 90th percentile profit when varying Γ and price(r)

Figure 4.8 also show that the 10th percentile profit increases more sharply as the multiplier of transportation cost, ν, increases. Increased ν encourages opening more facilities, which are located closer to customers, and it may lead to excessive investment on low demand paths while the robust model attempts to reduce this risk. Also, in both figures, we observe that employing the budget of uncertainty can mitigate over-conservatism as shown in previous cases.

Table 4.3 supports the observations above when varying the price and transportation cost. As r increases, both the number of open facilities and the total capacities increase. However, as ν increases, the number of open facilities increases, but the total capacities decrease. We also observe that the average number of open facilities and total capacities decrease as Γ increases.

Next, we report observation for our model in the setting with ambiguous demand, in an attempt to characterize where the robust model adds value and impacts decision-making.

In Table 4.4, we compare the average expected profit for the nominal versus the robust model with various budgets of uncertainty in the setting with ambiguous demand for each node.

We compare the solutions resulting from varying capacity expansion costs(c_i^0) and the multiplier of transportation costs(ν). The solutions of the nominal model are obtained by setting $\Gamma_{tm}^D = 0$. One interesting observation is that when budget of uncertainty is employed ($\Gamma_{tm}^D = 20, 60$), the robust model can provide solutions that outperform the solutions of the nominal model in terms of average expected profit in some cases, for example, when

Table 4.3. Number of open facilities, total capacity, proportion of covered demand, and proportion of unused capacity, averaged over 30 problem instances and over a 30 sample scenario trees when varying Γ, and r and ν

		r				ν			
	Γ(%)	[1,1.5]	[1.5,2]	[2,2.5]	[2.5,3]	0.005	0.01	0.02	0.03
Average Number of Open Facilities	0	3.53	3.63	3.63	3.63	2.43	3.63	5	5.7
	10	3.27	3.4	3.4	3.4	2.3	3.4	4.7	5.3
	20	3.1	3.13	3.13	3.13	2.07	3.13	4.3	5.03
	30	2.97	3.06	3.07	3.07	1.99	3.06	4.13	4.9
	40	2.76	2.9	2.9	2.9	1.95	2.9	3.93	4.57
	50	2.73	2.83	2.83	2.83	1.9	2.83	3.86	4.43
Average Total Capacity	0	384.5	408.1	408.1	408.1	408.4	408.1	403.4	378.2
	10	367.2	394.3	397.2	397.3	395.5	394.3	387.4	361.1
	20	350.5	376.3	377.8	381.6	376.5	376.3	366.4	340.3
	30	320.3	345.6	356.3	357.2	347.4	345.6	337.9	315.7
	40	305.6	333.3	336	337.6	325.8	333.3	323.4	298
	50	277.5	307.5	313.1	321.8	298.2	307.5	295.3	273
Average Covered Demand(%)	0	82.5	83.3	83.3	83.3	83.3	83.3	83.2	81.5
	10	81.8	82.8	82.9	82.9	82.9	82.8	82.6	80.5
	20	80.6	81.8	82	82	81.8	81.8	81.4	78.8
	30	78.5	79.9	80.3	80.5	79.9	79.9	79.4	77.2
	40	77.4	79.1	79.1	79	78.5	79.1	77.8	75.1
	50	73.4	75.7	76.1	76.2	74.6	75.7	74.1	71.5
Average Unused Capacity(%)	0	29.6	30.3	30.3	30.3	30.4	30.3	30	29.1
	10	27.6	28.2	28.2	28.2	28.3	28.2	28	27.4
	20	26.8	27.3	27.2	27.2	27.4	27.3	27	26.5
	30	23.5	23.6	23.3	22.8	23.4	23.6	23.4	23.1
	40	19.7	20	20.4	20.2	19.8	20	20.8	20.3
	50	19.8	19.8	21	21.6	20.2	19.8	21.2	20.8

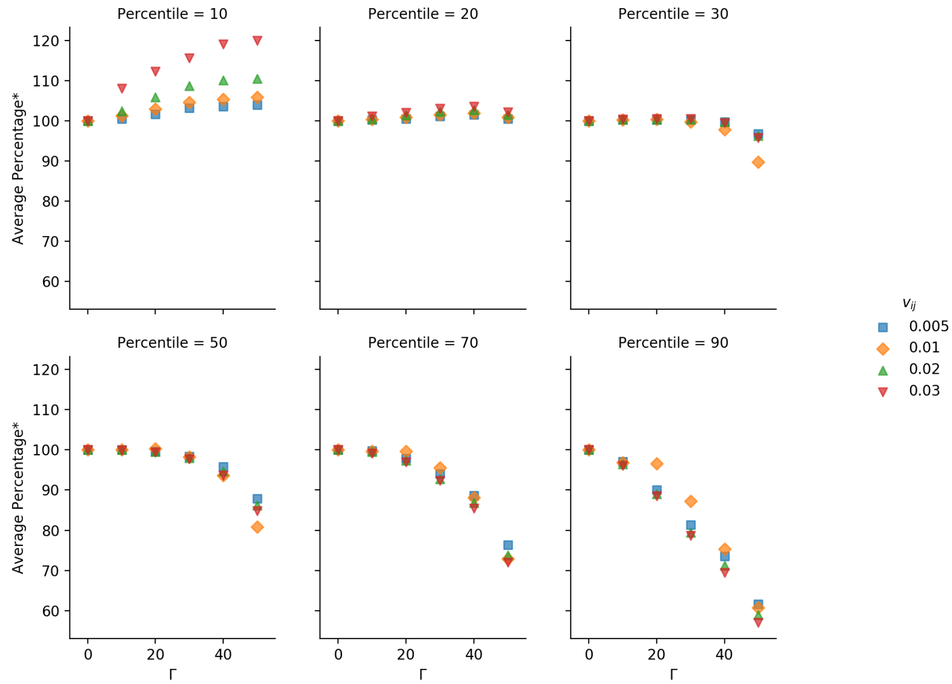


Figure 4.8. Average percentage of robust model to nominal model in 10, 20, 30, 50, 70 and 90th percentile profit when varying Γ and multiplier of transportation cost(ν)

Table 4.4. Average expected profit

	c_i^0				$\nu(\times 10^3)$			
$\Gamma_{tm}^D(\%)$	[.1,.2]	[.5,.6]	[1,1.1]	[1.5,1.6]	0.005	0.01	0.02	0.03
0	100	100	100	100	100	100	100	100
20	100.6	99.8	99.6	98	100.6	99.8	99.8	100.4
60	100.5	100.03	93.4	91.9	99.0	100.03	96.6	99.3
100	81.5	82.3	82.1	80.4	81.2	82.3	83.6	83

$c_i^0 \in [.1, .2]$, and $\nu = 0.005, 0.03$.

The robust model with $\Gamma_{tm}^D = 20, 60$ builds more capacity than the nominal model, particularly when the capacity expansion cost is low, for example $c_i^0 \in [.1, .2]$, as shown in Table 4.5. When ambiguities in demand are realized, more demand can be satisfied with this additional capacity. Since the capacity expansion cost is low, fulfilling more demand when higher demand is realized can outweigh the penalty of wasting capacity when less demand is realized. Thus the robust solution has better expected profit than the nominal solution in this case. We also observe in Table 4.5 that the robust model with $\Gamma_{tm}^D = 20$ builds more capacity than the nominal model, and this is more noticeable when $\nu = 0.005, 0.03$. This additional capacity can be beneficial when the transportation cost is relatively lower than the capacity expansion cost. If ν has a small value, then transportation cost is also low. Also, if ν has a large value, then more facilities are built and facilities are located closer to customers, which can lead to lower transportation cost.

Table 4.5. Number of open facilities, total capacity, proportion of covered demand, and proportion of unused capacity, averaged over 30 problem instances and over a 30 sample scenario trees when varying Γ^D_{tm} and c_i^0 and ν

	$\Gamma^D(\%)$	c_i^0 [.1,.2]	[.5,.6]	[1,1.1]	[1.5,1.6]	ν 0.005	0.01	0.02	0.03
Average Number of Open Facilities	0	3.63	3.63	3.47	3.4	2.4	3.63	4.9	5.37
	20	3.46	3.27	3.13	3	2.1	3.27	4.47	5.2
	60	3.2	3.13	3.03	3	2	3.13	4.3	4.93
	100	3.03	3.03	2.9	2.9	2	3.03	4.2	4.6
Average Total Capacity	0	412	411.5	355.4	333.7	412.4	411.5	403.3	373.6
	20	490.7	415.8	361.7	305.3	431.6	415.8	408.8	389.1
	60	463.3	345.2	281.1	253.5	349.7	345.2	332.1	298.1
	100	288.5	287.8	244	232.3	288.6	287.8	278.7	252.9
Average Covered Demand(%)	0	82.7	82.7	80.9	80.0	82.7	82.7	82.4	80
	20	83.3	82.3	81	77.7	82.9	82.3	82.1	80.1
	60	83.2	80.4	76.1	72.7	80.6	80.4	79.4	75.7
	100	72.4	72.4	69.3	66.1	72.6	72.4	71.9	68.1
Average Unused Capacity(%)	0	31	31.1	28.9	25.3	31.3	31.1	30.8	30
	20	39.1	35.5	30.8	27.8	35.5	35.5	34.6	33.9
	60	37	29.5	24.7	21.9	29.8	29.5	28.4	26.3
	100	22.5	22.6	21.1	20.2	22.8	22.6	22.2	21.7

Additionally, we present the values in percentiles for the average expected profit in the Appendix A.2.1. We also compare the average percentiles of profits for nominal versus robust models as the capacity expansion cost(c_i^0) varies in Appendix A.2.2. The observations in these analysis found in Appendix align with the result of the expected profit analysis shown in Table 4.4.

Overall, at least in these examples, we observe that it is beneficial to model ambiguities in event probabilities with a proper budget of uncertainty to find risk averse solution at little expense to overall profit. Also, modeling ambiguities in demand with a proper budget of uncertainty is beneficial for finding solutions that provide better expected profit and better overall profits, particularly when the investment costs or transportation costs are low.

4.4.2 Computational Analysis of Heuristics

In this section, we solve the model in Section 4.2 using both commercial software and the heuristics presented in Section 4.3. We compare the time to solve the model and the gap to optimality using these approaches. Our goal is to determine when we can solve models to optimality using commercial software, when we need to use heuristics, and how effective those heuristics are when we do use them.

We solve our model using CPLEX version 12.6, as described above, with a 3,600 second time limit. We vary the planning horizon(T), the number of facility ($|I|$), and the number of demand locations ($|J|$) because the size of the problem depends on T, $|I|$ and $|J|$. Recall that the total number of scenarios is $|P| = 2^{T-1}$. The number of variables and constraints increases as T, $|I|$ and $|J|$ increase. In particular, we vary T, $|I|$ and $|J|$ as follows: $T = \{1, 5, 10\}$, $(|I|, |J|) = \{(5, 10), (10, 20), (20, 40), (50, 100)\}$. We also attempt to measure the impact of varying the budget of uncertainty, where $\Gamma(\%) = \{10, 20, 50\}$ and $\Gamma^D_{tm}(\%) = \{10, 50, 100\}$. we keep Γ^D_{tm} constant for all nodes tm, so we drop the subscript tm in the subsequent explanation.

For each combination of $(T, |I|, |J|, \Gamma, \Gamma^D)$, we solve (or attempt to solve) 5 problem instances using commercial software. We randomly generate the problem instances using base setting of parameters described in the Section 4.4.1.

Table 4.6 presents the average computation time over 5 problem instances for various problem sizes. For each problem instance, we solve the model to optimality or to the 3,600 second time limit, whichever comes first.

If a feasible solution is found but an optimal solution is not, we report the optimality gap. If a feasible integer solution is not found within the time limit or an out-of-memory error is returned, dashes are reported in Table 4.6. We omit reporting the results for very small instances such as $(T, |I|, |J|) = (1, 5, 10), (1, 10, 20), (1, 20, 40), (5, 5, 10)$ which obviously can be solved within a few seconds using CPLEX.

As expected, we observe that as the planning horizon, the number of facility locations, and the number of customer locations increase, the model becomes significantly harder to solve. In our tests, when $T = 5$, the model becomes challenging to solve for the instances with $(|I|, |J|) = (50, 100)$ and $\Gamma^D \geq 10$. When $T = 10$, the model can be solved only for the some instances with $(|I|, |J|) = (5, 10)$ and $(|I|, |J|) = (10, 20)$. For instances with $(|I|, |J|) = (5, 10)$, optimal solutions can be found only when the budget of uncertainty as follows: $\Gamma^D = 0, 100$ or $\Gamma = 0$ or $(\Gamma, \Gamma^D) = (10, 10), (10, 20)$. For instances with $(|I|, |J|) = (10, 20)$, optimal solutions can be found only when the budget of uncertainty as follows: $\Gamma^D = 0$ or $(\Gamma, \Gamma^D) = (0, 10)$

We observe that incorporating ambiguous probabilities into our nominal model increases computation time significantly only when the problem size is large (e.g., $(T, |I|, |J|) = (5, 50, 100), (10, 10, 20)$), and it does not increase computation time significantly for small problem instances. However, when we incorporate ambiguous demand into our nominal model, the computation time increases significantly. Recall that to formulate **FLFV-R**, we borrow the concept of ARO, where the decision variable z^{tm}_{ij} is defined as $w^{tm}_{ij} + \zeta^{tm}_{ij}\tilde{d}^{tm}_j$. As Ben-Tal et al. (2004) points out, ARO is computationally more expensive than RO, and this also explains our case. In particular, in our model, the number of variables is increased by $1+2|S|+2|N| = 3|N|+2$ (since $|N| = 2|S|-1$) when incorporating ambiguous probabilities, whereas it is increased by $(|I| + |J| + |I||J|)|N|$ when incorporating ambiguous demands. For example, our robust model in the instance with $(T, |I|, |J|) = (5, 20, 40)$ has additional number of variables, 191 from modeling ambiguous probabilities and 54,180 from modeling ambiguous probabilities, compared to the nominal model. Therefore, this significant increased number of variables from modeling ambiguous demand causes a notable increase in computation time.

We also explore the effectiveness of our heuristic algorithms, Blocked Nested Decomposition (BND) and Blocked Stochastic Dual Dynamic Programming (BSDDP). We consider several problem instances of different sizes, as follows: $(|I|, |J|, T) = (5, 10, 15)$, $(10, 20, 10)$, $(10, 20, 15)$, $(20, 40, 10)$, $(20, 40, 15)$, $(50, 100, 5)$, $(50, 100, 10)$. We also vary the budget of uncertainty. These instances are chosen based on the result of the first computational ex-

Table 4.6. Average computation time (in seconds) using CPLEX when varying problem sizes and the budget of uncertainties

T	$\|I\|$	$\|J\|$	$\Gamma(\%)$	$\Gamma^D(\%)$			
				0	10	50	100
1	50	100	0	4.37	40.8	49.6	8.95
			10	4.19	40.4	50.3	9.37
			20	4.28	39.7	50.3	9.44
			50	5.31	48.9	20.5	11.4
5	10	20	0	2.31	14.6	28.3	10.0
			10	2.97	24.0	42.5	15.2
			20	3.99	36.3	42.1	20.9
			50	4.48	49	49.8	26.0
	20	40	0	11.03	672.1	444.9	144.3
			10	21.6	655.8	656.3	318.7
			20	27.3	912.3	809.3	301.6
			50	56.9	837.5	867	414.4
	50	100	0	207.7	(20.6%)	(29.6%)	(10.4%)
			10	369.8	-	-	(12%)
			20	714.6	-	-	(16%)
			50	1089.8	(61.4%)	(70.2%)	(61%)
10	5	10	0	31.8	319.6	628.1	332.2
			10	69.6	1660.4	-	729.2
			20	65.2	1518.6	-	882.7
			50	83.9	-	-	1463.8
	10	20	0	154.1	730.7	-	-
			10	467.1	-	-	-
			20	426	-	-	-
			50	728.1	-	-	-
	20	40	0	-	-	-	-
			10	-	-	-	-
			20	-	-	-	-
			50	-	-	-	-
	50	100	0	-	-	-	-
			10	-	-	-	-
			20	-	-	-	-
			50	-	-	-	-

periment, where we observed that CPLEX is incapable of solving them due memory or time constraints.

For BSDDP, we select one sample path randomly in the forward step until the upper bound becomes stable. This approach is computationally beneficial as shown in Zou et al. (2018) and Shapiro et al. (2013). We then choose 50 samples randomly for a Monte Carlo simulation to construct a 95% confidence probabilistic lower bound.

In Table 4.7, the column Heuristic indicates the heuristic algorithm used to solve each problem instance. Time presents the computation time in seconds. Gap(%) indicates the gap between best upper bound in the backward step and probabilistic lower bound in the simulation for each problem instance of $(T, |I|, |J|, \Gamma, \Gamma^D)$. It is worth mentioning that the gaps when using BSDDP may not decrease monotonically, because the probabilistic lower bound depends on the sample paths chosen.

In our tests, we use BND to solve problem instances of $(10, 20, 10)$ and $(20, 40, 10)$, since BND requires fewer iterations than BSDDP to converge. Although each iteration takes slightly longer in BND than in BSDDP, the computational time to find converged solution can be shorten more significantly by having fewer iterations for these medium size instances. For large problem instances which have long planning horizon(in our example, $T = 15$) or large numbers of locations (in our example, $(|I|, |J|) = (50, 100)$), we find BSDDP is more efficient, and thus we report the result of this algorithm. As shown table 4.7, we find solutions with less than a 3% gap within a reasonable time in most cases using BND and BSDDP except when $(T, |I|, |J|) = (10, 50, 100)$.

When $(T, |I|, |J|) = (10, 50, 100)$, particularly with $\Gamma > 0$ or $\Gamma^D > 0$, we cannot solve this instance using CPLEX as we run out of memory. Using BSDDP, we are able to find a feasible solution for this problem instance within $3,600$ seconds, however, gaps of $26 - 27\%$ are observed.

Overall, commercial software can effectively solve moderate-sized problems with a planning horizon of fewer than 10 periods if the number of locations are not large, for example, $(|I|, |J|) = (5, 10)(10, 20)(20, 40)$ or the planning horizon is at most 5 if the number of locations are large $(|I|, |J|) = (50, 100)$. For large problem instances which are challenging to solve using commercial software, the heuristic algorithms BND and BSDDP are effective alternatives. In particular, BND can be effective for smaller problems, whereas BSDDP can be effective for larger problems which have long planning horizon or large number of locations. However, for problems with long planning horizon and the large number of locations, novel heuristics are still needed.

4.5 Conclusion

In this chapter we study the capacitated facility location model where demand is dependent on the outcome of a series of Bernoulli trials with uncertain probabilities of success, and where demand is not completely resolved after the event outcomes are known. We present a robust stochastic integer programming formulation. Our numerical studies suggest that it is beneficial to model ambiguities in event probabilities with a correctly chosen budget of uncertainty to find risk averse solution at a little expense to overall profits. Also, when the investment cost or transportation cost are low, it is beneficial to model ambiguities in

Table 4.7. Average computation time (in seconds) using heuristic algorithms when varying problem sizes and the budget of uncertainties

$\|I\|$	$\|J\|$	T	Heuristic	$\Gamma^D(\%)$	$\Gamma(\%)$	Time	Gap(%)
5	10	15	BSDDP	0	0	251.7	1.1
					10	361.7	2.03
				10	0	253.4	1.13
					10	360.6	2.03
					50	257	3
10	20	10	BND	0	0	35.99	1.99
					10	406.8	0.35
				10	0	230.6	0.67
					10	409.8	0.36
					50	674.3	0.99
		15	BSDDP	0	0	1,215.4	1.6
					10	1,487.8	1.6
				10	0	1,726.9	1.15
					10	1,506.4	1.65
					50	2,262.9	0.25
20	40	10	BND	0	0	389.3	0.8
					10	916.2	0.5
				10	0	704.1	0.67
					10	700.9	0.67
					50	1,466.8	0.4
		15	BSDDP	0	0	3,141.8	0.7
					10	3,627	0.5
				10	0	3,589.8	0.9
					10	3,553.9	0.5
					50	4,060	19
50	100	5	BSDDP	0	0	475.1	2.5
					10	313.1	2.9
				10	0	626.7	0.92
					10	619.5	0.92
					50	1,035.6	2.03
		10	BSDDP	0	0	3,590	6.9
					10	3,820	27
				10	0	3,800	27
					10	3,680	26.1
					50	3,671	26.1

demand after the event realization with a correctly chosen budget of uncertainty in terms of the expected profit and overall profits.

We also develop and test two heuristics: Blocked Nested Decomposition(BND) and Blocked Stochastic Dual Dynamic Programming(BSDDP). We demonstrate that both heuristics can effectively solve large problem instances that we cannot solve using commercial software. In particular, BND is effective for smaller problems, whereas BSDDP is effective for larger problems, although more efficient approaches needs to be developed for much larger problems with both a long planning horizon and a large number of locations.

Chapter 5

Summary

In this thesis, we studied two strategic decision planning models – a capacity planning model and a capacitated facility location planning model, where demand is dependent on the outcome of a series of Bernoulli trials with unknown probabilities of success.

In the first problem, we developed approaches to formulate and solve models that capture a variety of different objectives such as minimizing expected cost, minimizing α-reliable regret, and minimizing α-reliable mean excess regret. We formulated these models as robust stochastic integer programs and showed that incorporating ambiguity in success probabilities does not increase the complexity of the problems. Our computational study suggests that when the shortage cost is relatively higher than the capacity investment cost, explicitly modeling ambiguous probabilities focusing on minimizing expected cost leads to solutions that actually perform quite well for all of the objective functions we considered in our models. Two effective heuristic approaches – a simple rolling horizon(SRH) approach, and a delayed event spike approach with rolling horizon (DERH) are developed to find good solutions to large problems that we are unable to solve effectively with commercial optimization software. We demonstrated that DERH is computationally more expensive, but it outperforms SRH in terms of solution quality.

In this problem, we have thus far limited our consideration to a single-product, to binary event outcomes, and to deterministic demand given event outcomes. In many real world settings, however, multiple products are under consideration. For example, in the pharmaceutical industry, multiple products are often concurrently under clinical trials, and some of these products could potentially share manufacturing capacity. Often, these products are in different stage of clinical trials. It can be one direction for future research to develop approaches that enable us to address these computationally more challenging settings.

Next, we developed approaches to formulate and solve the capacitated facility location model with an objective of maximizing expected profit. In addition to ambiguous success probabilities, we considered ambiguous demand, where demand is not completely resolved after the event outcomes are known. These ambiguities were incorporated into our nominal model using robust optimization techniques.

Our numerical studies suggest that it is beneficial to model ambiguities in event probabilities with a correctly chosen budget of uncertainty to find risk averse solution at little expense to overall profits. Also, when investment or transportation cost are low, explicitly modeling ambiguities in demand after the event realization with a correctly chosen budget of uncertainty is beneficial for finding better solutions than those found by the nominal model in terms of the expected profit and the overall profits.

Two effective heuristic approaches – Blocked Nested Decomposition(BND) and Blocked Stochastic Dual Dynamic Programming(BSDDP) were developed to solve large problem instances that we cannot solve using commercial software. We demonstrated that BND is effective for smaller problems, whereas BSDDP is effective for larger problems, although more efficient approach needs to be developed for much larger problems with both a long planning horizon and a large number of locations.

Bibliography

Ahmed, S., King, A. J., & Parija, G. (2003). A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. *Journal of Global Optimization*, *26*, 3–24.

Ahmed, S., & Sahinidis, N. V. (2003). An approximation scheme for stochastic integer programs arising in capacity expansion. *Operations Research*, *51*, 461–471.

Albareda-Sambola, M., Alonso-Ayuso, A., Escudero, L. F., Fernández, E., & Pizarro, C. (2013). Fix-and-relax-coordination for a multi-period location–allocation problem under uncertainty. *Computers & Operations Research*, *40*, 2878–2892.

Antunes, A., & Peeters, D. (2001). On solving complex multi-period location models using simulated annealing. *European Journal of Operational Research*, *130*, 190–201.

Ardestani-Jaafari, A., & Delage, E. (2018). The value of flexibility in robust location–transportation problems. *Transportation Science*, *52*(1), 189–209.

Artzner, P., Delbaen, F., Eber, J.-M., & Heath, D. (1999). Coherent measures of risk. *Mathematical Finance*, *9*, 203–228.

Atamtürk, A., & Zhang, M. (2007). Two-stage robust network flow and design under demand uncertainty. *Operations Research*, *55*, 662–673.

Ballou, R. H. (1968). Dynamic warehouse location analysis. *Journal of Marketing Research*, *5*, 271.

Baron, O., Milner, J., & Naseraldin, H. (2010). Facility location: A robust optimization approach. *Production and Operations Management*, *20*, 772–785.

Bean, J. C., Higle, J. L., & Smith, R. L. (1992). Capacity expansion under stochastic demands. *Operations Research*, *40*, S210–S216.

Beltran-Royo, C., Escudero, L., Monge, J., & Rodriguez-Ravines, R. (2014). An effective heuristic for multistage linear programming with a stochastic right-hand side. *Computers & Operations Research*, *51*, 237–250.

Ben-Tal, A., El-Ghaoui, L., & Nemirovsky, A. (2009). *Robust optimization*. Princeton University Press, Princeton, NJ.

Ben-Tal, A., Goryashko, A., Guslitzer, E., & Nemirovski, A. (2004). Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, *99*, 351–376.

Bertsimas, D., & de Ruiter, F. J. C. T. (2016). Duality in two-stage adaptive linear optimization: Faster computation and stronger bounds. *INFORMS Journal on Computing*, *28*, 500–511.

Bertsimas, D., & Sim, M. (2003). Robust discrete optimization and network flows. *Mathematical Programming*, *98*, 49–71.

Bertsimas, D., & Thiele, A. (2006). A robust optimization approach to inventory theory. *Operations Research*, *54*, 150–168.

Birge, J. R. (1985). Aggregation bounds in stochastic linear programming. *Mathematical Programming*, *31*, 25–41.

Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming.* Springer New York.

Canel, C., Khumawala, B. M., Law, J., & Loh, A. (2001). An algorithm for the capacitated, multi-commodity multi-period facility location problem. *Computers & Operations Research*, *28*(5), 411–427.

Chen, G., Daskin, M. S., Shen, Z.-J. M., & Uryasev, S. (2006). The α-reliable mean-excess regret model for stochastic facility location modeling. *Naval Research Logistics*, *53*, 617–626.

Chen, Z.-L., Li, S., & Tirupati, D. (2002). A scenario-based stochastic programming approach for technology and capacity planning. *Computers & Operations Research*, *29*, 781–806.

Chen, Z.-L., & Powell, W. B. (1999). Convergent cutting-plane and partial sampling algorithm for multistage stochastic linear programs with recourse. *Journal of Optimization Theory and Applications*, *102*, 497–524.

Chien, C.-F., & Zheng, J.-N. (2011). Mini–max regret strategy for robust capacity expansion decisions in semiconductor manufacturing. *Journal of Intelligent Manufacturing*, *23*, 2151–2159.

Claro, J., & de Sousa, J. P. (2008). A multiobjective metaheuristic for a mean-risk multistage capacity investment problem. *Journal of Heuristics*, *16*, 85–115.

Correia, I., & da Gama, F. S. (2015). Facility location under uncertainty, In *Location science.* Springer International Publishing.

Daskin, M. S., Hesse, S. M., & Revelle, C. S. (1997). α-reliable p-minimax regret: A new model for strategic facility location modeling. *Location Science*, *5*, 227–246.

Davis, M. H. A., Dempster, M. A. H., Sethi, S. P., & Vermes, D. (1987). Optimal capacity expansion under uncertainty. *Advances in Applied Probability*, *19*, 156–176.

Eppen, G. D., Martin, R. K., & Schrage, L. (1989). OR practice—a scenario approach to capacity planning. *Operations Research*, *37*, 517–527.

Fine, C. H., & Freund, R. M. (1990). Optimal investment in product-flexible manufacturing capacity. *Management Science*, *36*, 449–466.

Freidenfelds, J. (1980). Capacity expansion when demand is a birth-death random process. *Operations Research*, *28*, 712–721.

Gabrel, V., Lacroix, M., Murat, C., & Remli, N. (2014). Robust location transportation problems under uncertain demands. *Discrete Applied Mathematics*, *164*, 100–111.

Hernández, P., Alonso-Ayuso, A., Bravo, F., Escudero, L. F., Guignard, M., Marianov, V., & Weintraub, A. (2012). A branch-and-cluster coordination scheme for selecting prison facility sites under uncertainty. *Computers & Operations Research*, *39*, 2232–2241.

Huang, K., & Ahmed, S. (2009). The value of multistage stochastic programming in capacity planning under uncertainty. *Operations Research*, *57*, 893–904.

Jena, S. D., Cordeau, J.-F., & Gendron, B. (2016). Solving a dynamic facility location problem with partial closing and reopening. *Computers & Operations Research*, *67*, 143–154.

Julka, N., Baines, T., Tjahjono, B., Lendermann, P., & Vitanov, V. (2007). A review of multi-factor capacity expansion models for manufacturing plants: Searching for a holistic decision aid. *International Journal of Production Economics*, *106*, 607–621.

Loomes, G., & Sugden, R. (1982). Regret theory: An alternative theory of rational choice under uncertainty. *The Economic Journal*, *92*, 805.

Luss, H. (1982). Operations research and capacity expansion problems: A survey. *Operations Research*, *30*, 907–947.

Manne, A. S. (1961). Capacity expansion and probabilistic growth. *Econometrica*, *29*, 632.

Melo, M., Nickel, S., & da Gama, F. S. (2006). Dynamic multi-commodity capacitated facility location: A mathematical modeling framework for strategic supply chain planning. *Computers & Operations Research*, *33*, 181–208.

Nickel, S., Saldanha-da-Gama, F., & Ziegler, H.-P. (2012). A multi-stage stochastic supply network design problem with financial decisions and risk management. *Omega*, *40*, 511–524.

Noyan, N. (2012). Risk-averse two-stage stochastic programming with an application to disaster management. *Computers & Operations Research*, *39*, 541–559.

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

Pereira, M. V., & Pinto, L. M. (1991). Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, *52*, 359–375.

Rajagopalan, S. (1994). Capacity expansion with alternative technology choices. *European Journal of Operational Research*, *77*, 392–403.

Riis, M., & Andersen, K. A. (2002). Capacitated network design with uncertain demand. *INFORMS Journal on Computing*, *14*, 247–260.

Riis, M., & Lodahl, J. (2002). A bicriteria stochastic programming model for capacity expansion in telecommunications. *Mathematical Methods of Operations Research*, *56*, 83–100.

Rockafellar, R., & Uryasev, S. (2000). Optimization of conditional value-at-risk. *Journal of Risk*, *2*, 21–41.

Rockafellar, R., & Uryasev, S. (2002). Conditional value-at-risk for general loss distributions. *Journal of Banking & Finance*, *26*, 1443–1471.

Schultz, R., & Tiedemann, S. (2005). Conditional value-at-risk in stochastic programs with mixed-integer recourse. *Mathematical Programming*, *105*, 365–386.

Shapiro, A., Tekaya, W., da Costa, J. P., & Soares, M. P. (2013). Risk neutral and risk averse stochastic dual dynamic programming method. *European Journal of Operational Research*, *224*, 375–391.

Singh, K. J., Philpott, A. B., & Wood, R. K. (2009). Dantzig-wolfe decomposition for solving multistage stochastic capacity-planning problems. *Operations Research*, *57*, 1271–1286.

Snow, D. C., Wheelwright, S. C., & Wagonfeld, A. B. (2006). Genentech–capacity planning. *Harvard Business School Case 606-052*.

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

Swaminathan, J. M. (2000). Tool capacity planning for semiconductor fabrication facilities under demand uncertainty. *European Journal of Operational Research*, *120*, 545–558.

Thiele, A. (2007). Robust stochastic programming with uncertain probabilities. *IMA Journal of Management Mathematics*, *19*, 289–321.

Torres Soto, J. E. (2009). *Dynamic and robust capacitated facility location in time varying demand environments* (Doctoral dissertation). Texas A&M University. College Station, TX, USA.

Van Mieghem, J. A. (2003). Capacity management, investment, and hedging: Review and recent developments. *Manufacturing & Service Operations Management*, *5*, 269–302.

Wesolowsky, G. O. (1973). Dynamic facility location. *Management Science*, *19*, 1241–1248.

Wu, S. D., Erkoc, M., & Karabuk, S. (2005). Managing capacity in the high-tech industry: A review of literature. *The Engineering Economist*, *50*, 125–158.

Yu, X., Ahmed, S., & Shen, S. (2018). On the value of multistage stochastic facility location with (or without) risk aversion.

Zeng, B., & Zhao, L. (2013). Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters*, *41*, 457–461.

Zou, J., Ahmed, S., & Sun, X. (2018). Stochastic dual dynamic integer programming. *Mathematical Programming*, *175*, 461–502.

Appendix A

Supplement Materials

A.1 Nested Decomposition for model FLFV-RC

In Section 4.2, we presented model **FLFV-RC** using following variable.

$$z_{ij}^{tm} = w_{ij}^{tm} + \zeta_{ij}^{tm}\tilde{d}_j^{tm}$$

Let the objective function at tm as follows:

$$\tilde{g}_t(s_{A(tm)}^l, s_{tm}^l, z_{tm}^l) = -\sum_{i\in I} c_i^t(s_i^{tm} - s_i^{A(tm)}) + \sum_{i\in I}\sum_{j\in J}(r^t - v_{ij}^t)(w_{ij}^{tm} + \zeta_{ij}^{tm}\bar{d}_j^{tm}) + \sum_{j\in J}\lambda_{jtm} + \Gamma_{tm}^D\mu^{tm}$$

$$\tilde{g}_0(\hat{x}_{01}^l, s_{01}^l, z_{01}^l) = -\sum_{i\in I}(f_i\hat{x}_i + c_i^t s_i^{01}) + \sum_{i\in I}\sum_{j\in J}(r^0 - v_{ij}^0)(w_{ij}^{01} + \zeta_{ij}^{01}\bar{d}_j^{01}) + \sum_{j\in J}\lambda_{j01} + \Gamma_{01}^D\mu^{01}$$

Recall that in **FLFV-RC**, the following constraints are introduced due to ambiguous parameter d_t and substitution of z_{ij}^{tm}.

$$\sum_{j\in R}(w_{ij}^{tm} + \zeta_{ij}^{tm}\bar{d}_j^{tm}) + \sum_{j\in J}\theta_{jtm}^i + \Gamma_{tm}^D\nu_{tm}^i \le U_i \sum_{n\in\mathcal{P}(tm)} y_i^n \quad i \in I \tag{A.1}$$

$$\sum_{i\in I} w_{ij}^{tm} - (1 - \zeta_{ij}^{tm})(\bar{d}_j^{tm} - \hat{d}_j^{tm} * \min(1, \Gamma_{tm}^D)) \le 0 \quad j \in J \tag{A.2}$$

$$\lambda_{jtm} + \mu_{tm} \ge \sum_{i\in I}\hat{d}_j^{tm}\zeta_{ij}^{tm}(r - v_{ij}^t) \quad \forall j \in J \tag{A.3}$$

$$\theta_{jtm}^i + \nu_{tm}^i \ge \zeta_{ij}^{tm}\hat{d}_j^{tm} \quad i \in I \tag{A.4}$$

Then, for the robust model **FLFV-RC**, the recursive equation NDR_{tm}^l at node tm which we solve in each iteration of ND-based algorithm can be formulated as a linear program as shown below.

$$(NDR^l_{tm})$$

$$\underline{Q}^l_{tm}(w^{A(tm)l}, \tilde{\psi}^l_{C(tm)}) :=$$

$$\begin{aligned} max \quad & \tilde{g}_t(s^l_{A(tm)}, s^l_{tm}, z^l_{tm}) + \psi_{c(tm)_f}(w_{tm})^l + \pi^+_{c(tm)_s}\gamma^+_{tm} - \pi^-_{c(tm)_s}\gamma^-_{tm} \\ s.t. \quad & (4.48) - (4.51), (4.52) \\ & (A.1) - (A.4) \\ & \gamma^+_{tm} - \gamma^-_{tm} \geq \psi_{c(tm)_s} - \psi_{c(tm)_f} \end{aligned}$$

$$\underline{Q}^l_{01}(w_0, \tilde{\psi}^l_{C(01)}) :=$$

$$\begin{aligned} max \quad & \tilde{g}_0(x^l_{01}, s^l_{01}, z^l_{01}) + \psi_{12}(w_{01})^l + \pi^+_{11}\gamma^+_{01} - \pi^-_{11}\gamma^-_{01} \\ s.t. \quad & (4.42) - (4.43), (4.45) \\ & (A.1) - (A.4) \\ & \gamma^+_{01} - \gamma^-_{01} \geq \psi_{11} - \psi_{12} \end{aligned}$$

where $\hat{\psi}^l(w^l_{A(tm)})$, the approximation of value function $Q_{tm}(w_{A(tm)})$, is defined as:

$$\hat{\psi}^l_{tm}(w^l_{A(tm)}) := max\{\phi_{tm} : \quad \phi_{tm} \leq B_{tm}$$
$$\phi_{tm} \leq (\beta^q_{tm} + \alpha^{Aq}_{tm} s_{A(tm)} + \alpha^{Bq}_{tm} \hat{x}_{A(tm)}), \forall q = 1..l-1\}$$

A.2 Computational Analysis in the setting with Ambiguous Demand

A.2.1 Analysis of Average Expected Profit

In Table A.1, we compare the average percentage of 10th, 20th, 30th, 50th, 70th and 90th percentile expected profit of robust models to those expected profit of nominal model ($\Gamma = 0$). We observe that the percentage values of each percentile are all very close to the percentage values of the average expected profit in Table 4.4. Thus, modeling ambiguities in demand with a proper uncertainty budget can be beneficial in terms of overall expected profit in particular when the capacity investment cost is low ($c_i \in [.1, .2]$) or the transportation cost is low ($\nu = 0.005, 0.03$). Recall that when ν is small, transportation cost is also obviously low. When ν is large, v_{ijt} is high. However, facilities are forced to be located closer to customers in the solution , and it leads to lower the transportation cost.

A.2.2 Analysis of Average Percentile of Profits

In Figure A.1, we compare the average percentiles of profits for nominal versus robust models as the capacity expansion cost(c^0_i) varies. Figure A.1 is very different from Figure 4.5 (and other figures above when we test our models in the setting with ambiguous probabilities).

Table A.1. Average expected profit

Γ (%)	Percentile (%)	c_i^0 [.1,.2]	[.5,.6]	[1,1.1]	[1.5,1.6]	ν 0.005	0.01	0.02	0.03
20	10	100.3	99.6	99.4	97.8	100.4	99.6	99.6	100.2
	20	100.4	99.7	99.4	97.8	100.4	99.7	99.6	100.3
	30	100.5	99.8	99.5	97.9	100.5	99.8	99.7	100.3
	50	100.6	99.9	99.6	98.1	100.5	99.9	99.8	100.5
	70	100.7	99.9	99.7	98.1	100.7	99.9	99.8	100.5
	90	100.8	100	99.8	98.2	100.7	100	99.9	100.6
60	10	100.3	99.9	93.8	92	98.9	99.9	96.6	99.3
	20	100.3	99.9	93.7	91.9	98.8	99.9	96.6	99.3
	30	100.4	100	93.7	91.9	98.9	100	96.6	99.3
	50	100.5	100	93.7	91.9	99	100	96.7	99.4
	70	100.5	100.1	93.6	91.8	99	100.1	96.6	99.4
	90	100.6	100.2	93.7	91.9	99.1	100.2	96.6	99.5
100	10	81.6	82.5	82.5	80.7	81.3	82.5	83.9	83.2
	20	81.6	82.5	82.4	80.6	81.3	82.5	83.8	83.1
	30	81.6	82.4	82.3	80.5	81.2	82.4	83.8	83.1
	50	81.5	82.4	82.2	80.3	81.2	82.4	83.7	83
	70	81.4	82.2	82.1	80.2	81.1	82.2	83.5	82.9
	90	81.4	82.2	82	80.2	81.1	82.2	83.5	82.8

When the capacity expansion cost is high, for example, $c_i^0 \in (1.5, 1.6)$, the 10th percentile profit is significantly higher in the robust solution with $\Gamma_{tm}^D = 20\%$ than in the nominal solution, but other percentile profits are slightly low in the same comparison. This leads to slightly lower average expected profit in the robust solution than in the nominal solution as shown in Table 4.5.

When the capacity expansion cost is low, for example, $c_i^0 \in (0.1, 0.2)$, all percentile profits are slightly higher in the robust solutions with $\Gamma_{tm}^D = 20, 60\%$ than in the nominal solutions, which aligns with the result of the expected profit analysis shown in Table 4.4.

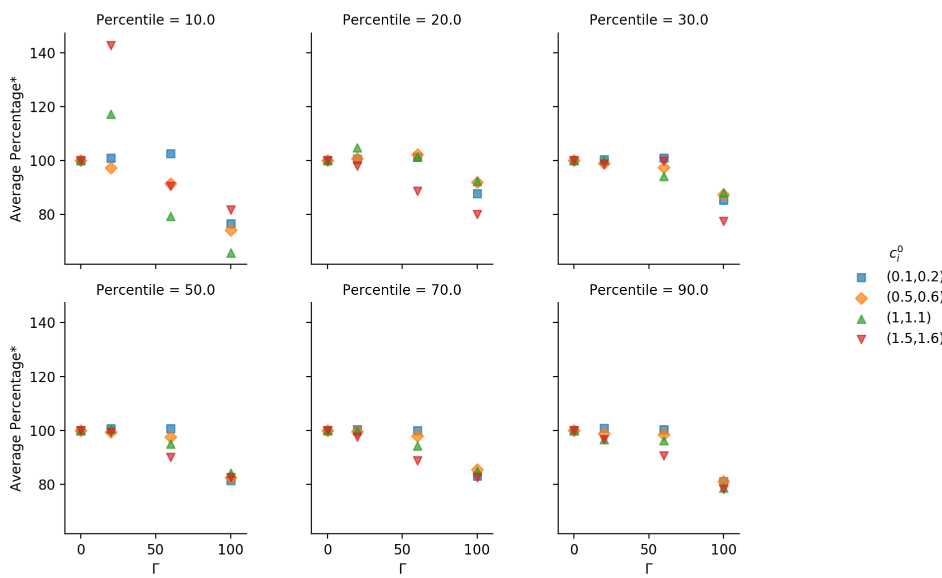


Figure A.1. Average percentage of robust model to nominal model in $10, 20, 30, 50, 70$ and 90th percentile profit when varying Γ and transportation cost(c_{it})