5-2021

Models, Theoretical Properties, and Solution Approaches for Stochastic Programming with Endogenous Uncertainty

Tanveer Hossain Bhuiyan
University of Tennessee, Knoxville, tbhuiyan@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Tanveer Hossain Bhuiyan entitled "Models, Theoretical Properties, and Solution Approaches for Stochastic Programming with Endogenous Uncertainty." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Industrial Engineering.

Hugh Medal, Major Professor

We have read this dissertation and recommend its acceptance:

James Ostrowski, Hoon Hwangbo, Guannan Zhang

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Models, Theoretical Properties, and Solution Approaches for Stochastic Programming with Endogenous Uncertainty

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Tanveer Hossain Bhuiyan

May 2021

To my wife, Sarah, and my parents, Shakawat and Noorjahan

Acknowledgments

First, I would like to show my gratitude to the almighty God for blessing me with this beautiful life. Next, I want to thank my wife, Sarah. She has supported me without hesitation and stood beside me in all of my problems in life. I would also like to thank my parents, Shakawat and Noorjahan, who have unconditionally supported and encouraged me since childhood. My gratitude goes to my siblings, Lina and Tamim, for encouraging me in my pursuit of the Ph.D. degree.

I would like to thank my advisor, Dr. Hugh Medal, for his continuous support in my research. He is a great mentor who always guided me in my research, encouraged me to take the challenge in learning new things, and provided help when I needed. I would also like to thank Dr. James Ostrowski for serving as a member in my dissertation committee. I learned many new concepts from his Integer Programming course. I would like to thank Dr. Hoon Hwangbo for serving as a member in my dissertation committee. His Machine Learning course enhanced my knowledge of several machine-learning algorithms. I also want to thank Dr. Guannan Zhang for serving as a member in my dissertation committee.

Abstract

In a typical optimization problem, uncertainty does not depend on the decisions being made in the optimization routine. But, in many application areas, decisions affect underlying uncertainty (endogenous uncertainty), either altering the probability distributions or the timing at which the uncertainty is resolved. Stochastic programming is a widely used method in optimization under uncertainty. Though plenty of research exists on stochastic programming where decisions affect the timing at which uncertainty is resolved, much less work has been done on stochastic programming where decisions alter probability distributions of uncertain parameters. Therefore, we propose methodologies for the latter category of optimization under endogenous uncertainty and demonstrate their benefits in some application areas.

First, we develop a data-driven stochastic program (integrates a supervised machine learning algorithm to estimate probability distributions of uncertain parameters) for a wildfire risk reduction problem, where resource allocation decisions probabilistically affect uncertain human behavior. The nonconvex model is linearized using a reformulation approach. To solve a realistic-sized problem, we introduce a simulation program to efficiently compute the recourse objective value for a large number of scenarios. We present managerial insights derived from the results obtained based on Santa Fe National Forest data.

Second, we develop a data-driven stochastic program with both endogenous and exogenous uncertainties with an application to combined infrastructure protection and network design problem. In the proposed model, some first-stage decision variables affect probability distributions, whereas others do not. We propose an exact reformulation for linearizing the nonconvex model and provide a theoretical justification of it. We designed an accelerated L-shaped decomposition algorithm to solve the linearized model. Results

obtained using transportation networks created based on the southeastern U.S. provide several key insights for practitioners in using this proposed methodology.

Finally, we study submodular optimization under endogenous uncertainty with an application to complex system reliability. Specifically, we prove that our stochastic program's reliability maximization objective function is submodular under some probability distributions commonly used in reliability literature. Utilizing the submodularity, we implement a continuous approximation algorithm capable of solving large-scale problems. We conduct a case study demonstrating the computational efficiency of the algorithm and providing insights.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In real-life decision-making problems, decision-makers often face uncertainty in problem parameters. In a typical optimization problem under uncertainty, decision makers' decisions do not affect the underlying uncertainty in the stochastic process, such as weather conditions—rainfall, wind, and solar. However, decisions can influence the underlying uncertainty in some application areas. For instance, consider a decision-maker seeking to strengthen a transportation network subject to random disruption or interdiction. In this problem, the post-disruption/interdiction state (survival or failure) of a network link is uncertain as the decision-maker does not have complete information about the disruption/interdiction. But, the decision-maker can influence this uncertainty by changing protection resource investment to the network links. A link is more likely to survive if the decision-maker invests more protection resources to that link. Therefore, investment decisions alter the survival probability of the links. Thus, in this optimization problem under uncertainty, decisions that are being made while solving the problem influences the underlying uncertainty in the problem parameters.

Based on the dependence of uncertainty on decisions, Goel and Grossmann [45] classifies uncertainty in the two following classes: (1) exogenous—uncertainty does not depend on the decisions, and (2) endogenous—uncertainty depends on the decisions. Decisions can either influence the probability distributions governing uncertainty or the timing at which uncertainty is resolved.

Depending on the effect of decisions on uncertainty, Goel and Grossmann [45] classifies the endogenous (decision–dependent) uncertainty into two types: (1) Type I—decisions alter the probability distributions of uncertain parameters, and (2) Type II—decisions affect the timing at which uncertainty is resolved. An example of Type I endogenous uncertainty is the above-mentioned transportation network strengthening problem under uncertainty in the links' survivability, where the investment decisions affect the survival probabilities of the links. An example of Type II endogenous uncertainty is a capacity expansion problem of a process network under uncertainty in productivity (yield) studied in Goel and Grossmann [46]. In this problem, a decision-maker seeks to install new processing units based on new technology to expand an existing network. As the new processing units are based on new technology, their yields are uncertain. The yield uncertainty of a new unit is resolved only when the decision-maker installs that unit and start production under the operating environment. Therefore, the decisions can control the timing of when the uncertainty is resolved. This Type II endogenous uncertainty is also known as *exogenous uncertainty with endogenous observations*, as the decisions do not affect the realization of uncertain parameters, instead affect the timing of revealing accurate information.

There are different modeling frameworks for problems with endogenous uncertainty in the literature, such as stochastic programming with decision-dependent uncertainty [45, 46, 94], robust optimization with decision-dependent uncertainty sets [75, 76], and distributionally robust optimization with decision-dependent ambiguity sets [86]. Different modeling frameworks are suitable for different application problems and the decision maker's preferences. Due to two types of endogenous uncertainties, the stochastic programming frameworks for decision-dependent uncertainty can be fundamentally classified into two groups. We refer to the stochastic programming framework with Type I endogenous uncertainty as *stochastic programming with endogenous uncertainty* (SPEU), whereas the stochastic programming framework with Type II endogenous uncertainty as *stochastic programming with endogenous anticipativity* (SPEA). The SPEA framework is suitable for modeling multi-period decision-making problems having decision-dependent uncertainty in the timing of when the actual value of uncertain parameters is revealed. These problems are formulated as multi-stage stochastic programs where non-anticipativity constraints

depend on the investment decision variables at each period. Among the different modeling frameworks mentioned-above, only SPEA has a fair amount of studies; other areas have been gaining attention very recently.

In the SPEU framework, problems can be modeled as two-stage or multi-stage stochastic programs, depending on the problem's nature. As the decision variables affect the probability distributions in an SPEU framework, it is different from the traditional two-stage stochastic program with exogenous uncertainty. In a scenario-based two-stage stochastic program, scenarios represent the realizations of uncertain parameters that follow specified probability distributions. Therefore, each realized value of an uncertain parameter in a scenario has a corresponding probability. First-stage decisions are made before realizing the uncertainty, whereas some recourse decisions are made after the uncertain parameters are realized in each scenario. In a stochastic program with exogenous uncertainty, scenario probabilities do not depend on the decision variables. They are fixed values, as the probability of a realized value of an uncertain parameter is independent of the first-stage decision variables.

On the other hand, in an SPEU model, the probability of occurring a particular realization of an uncertain parameter is a function of first-stage decision variables, making the scenario probabilities function of decision variables. For example, in the above-mentioned network strengthening problem, a link's survival probability is higher if more resource is invested on that link. If the decision-maker invests more resources on a subset of links, then the probability of a scenario where those links fail will be small. This dependency of scenario probabilities on decision variables makes the resulting model nonconvex.

This dissertation focuses on contributing to the SPEU methodology as well as in the potential application areas. The objectives of this dissertation are to (1) develop new data-driven SPEU models for problems having endogenous (decision-dependent) uncertainty, (2) introduce new mathematical reformulations to linearize the resulting nonconvex models, (3) analyze theoretical properties of the SPEU framework for a particular class of problems that can be utilized to solve large-scale problems, (4) design/implement both exact decomposition methods and approximation algorithms, and demonstrate the benefit of these models and solution methodologies to different real-life application areas.

1.1 Motivation

In many application areas, it is beneficial to model the probability distributions of the uncertain parameters as a function of the decision variables. For example, consider a problem of strengthening a set of critical infrastructures subject to random disruption, where the post-disruption capacity of infrastructures is uncertain. It is more practical to model the post-disruption capacity as a probabilistic function of the protection resource investment. Because it is likely that an infrastructure has a larger post-disruption capacity if more protection resources is invested in that infrastructure. In this case, the probability distribution of the post-disruption capacity is a function of the investment decisions. The SPEU framework is suitable for modeling this problem as it captures the natural decision-dependent probability (uncertainty) structure of the problem and can yield better quality solutions and more insights into the problem. On the contrary, solving problems with a natural decision-dependent uncertainty structure using a traditional stochastic program with exogenous uncertainty could be computationally very difficult. Because the model would need a larger number of scenarios than the SPEU framework to capture the decision-dependent uncertainty property.

Besides critical infrastructure protection, this SPEU framework has other application areas such as disaster preparedness and mitigation, supply chain risk mitigation, reliability of complex systems, and power generation expansion planning. But, this modeling framework has not been utilized in many application areas to date. Moreover, most of the studies on SPEU approached the problem with approximate and problem-specific heuristic methods and commercial solvers. Therefore, there is a need to advance the SPEU modeling and solution approaches, particularly by introducing new mathematical formulations, reformulation techniques, exact and approximate solution approaches utilizing some structural properties in different applications. Until this need is met, decision-makers in various application areas cannot obtain good quality solutions and valuable insights into the problems, resulting in systems that underperform and have excessive cost.

1.2 Literature Review

This section summarizes the literature related to the SPEU modeling frameworks, solution approaches, and application areas. A more detailed review of the literature pertaining to each methodological and application areas of contribution are presented in the corresponding chapters of this dissertation.

Among the early works on SPEU, Ahmed [4] is the first to introduce the notion of endogenous uncertainty in network design, server selection, and facility location problems. Later, Held and Woodruff [53] modeled a stochastic network interdiction problem as a multi-stage SPEU to maximize the probability of interdicting the flow of information or goods in a network with uncertain characteristics. The authors proposed a heuristics solution method to solve the model, demonstrating the effectiveness of the algorithm in solving multi-stage SPEU problems.

Some research studied the natural endogenous uncertainty in infrastructure protection literature. Peeta et al. [110] modeled a highway network strengthening problem as a two-stage stochastic program with the first-stage investment decisions affecting the survival probabilities of the links after a disaster. The authors used an approximation of the objective function based on the Taylor series expansion to resolve the non-convexity that arose because of the decision-dependent probability framework. Because of this approximation, the proposed approach cannot guarantee optimality, and their approximation does not provide a performance guarantee. Du and Peeta [34] extended the model of Peeta et al. [110] to allow for partial investment and solved the model using a heuristic algorithm coupled with a Monte Carlo simulation. Medal et al. [94] extended the work of Peeta et al. [110] by introducing a probability-chain reformulation of the nonconvex objective function in their two-stage stochastic program that allows for the implementation of an exact algorithm, L-shaped decomposition, in solving the model. The authors also proposed a greedy approximation algorithm to solve larger problem instances that can provide a worst-case performance guarantee of 0.63. These studies on infrastructure protection considered the problem where decision variables only affect the probability distributions. But, there are other aspects of the problem, such as a combined network design and infrastructure protection, where some

decisions affect the probability distributions, whereas others do not. Addressing this aspect of the problem would result in a new model and require new solution approaches to solve it.

Some studies utilized simulation approaches to solve SPEU problems. Ekin et al. [37] proposed a simulation-based approach for solving a two-stage SPEU model that recasts the optimization problem as a simulation problem treating the decision variables as random. Later, Ekin [36] proposed a two-stage stochastic programming model for production planning that includes exogenous and endogenous uncertainties. The authors used a simulation-based optimization method for solving the proposed model.

Recently, few studies proposed some new formulations to the SPEU literature. Hellemo et al. [55] proposed new two-stage SPEU models where the first-stage decisions can change the initial probability distribution to an entirely different one via affine transformation or a convex combination of the probability distributions. The authors solved their SPEU models using BARON solver. Krasko and Rebennack [71] proposed a two-stage multi-period SPEU model for a disaster mitigation problem, which the authors solved using BARON solver. Zhan et al. [146] presented a new multi-stage stochastic programming model with endogenous uncertainty for power generation expansion planning, where the probability distributions of electricity prices depend on the investment decisions. The authors reformulated the nonlinear multi-stage stochastic program into a mixed-integer linear one using a quasi-exact approach. Escudero et al. [38] introduced a risk-neutral and a risk-averse three-stage stochastic program for disaster preparedness and management problem where both exogenous and endogenous uncertainties exist. The authors proposed a cluster dual descent algorithm to obtain a feasible solution for the problem.

Most of the studies mentioned earlier used simulation approaches and commercial solvers to solve their models. But the simulation approaches often fail to provide optimal solution and commercial solvers may not be available to practitioners. Therefore, there is need for accelerated decomposition algorithms to solve SPEU problems to optimality. Also, the benefit of the SPEU framework has not been studied in other application areas, such as complex system reliability allocation. Though few studies introduced the SPEU framework to disaster mitigation, there are other aspects of this area where the SPEU framework can be beneficial to the practitioners.

Few studies analyzed the theoretical properties—submodularity and convexity/concavity—of the SPEU framework. Karaesmen and Van Ryzin [65] shows that the expected value function is submodular and componentwise concave utilizing the semigroup property of the binomial random variables having the same probability of success. Later, Medal et al. [94] proved the submodularity of the expected value function for binomial random variables having different probabilities of success. The authors used this submodularity property to provide a worst-case performance guarantee of a greedy approximation algorithm in solving large-sized SPEU problems. But, there are several other probability distributions widely used in many application areas, such as exponential and Weibull distributions. These distributions do not possess the particular semigroup property. Submodularity needs to be studied for these probability distributions to facilitate solving large-scale optimization problems under endogenous uncertainty in different application areas.

1.3 Contributions

In the current literature, the SPEU framework has not been utilized in many application areas such as combined network design and facility protection, disaster mitigation, and reliability allocation of complex systems. Moreover, as mentioned earlier, new reformulation techniques and decomposition algorithms need to be introduced to solve problems in those areas. Though some theoretical properties are analyzed for a few probability distributions, there is still a need to study those properties for other commonly used distributions in many real-life application areas.

Therefore, to fill the gaps in the literature, this dissertation makes the following contributions. Chapter 2 proposes a new data-driven SPEU model to model a risk-based incentive structure design problem that mitigates the risks of catastrophic natural disasters (wildfires) [17]. This model integrates a supervised learning algorithm to estimate probability distributions of the uncertain parameters by analyzing survey data. This model also incorporates a simulation program to efficiently compute the recourse problem's objective value for a very large number of scenarios.

Chapter 3 presents a new data-driven stochastic optimization model with exogenous and endogenous uncertainties for a combined network design and facility protection problem [16]. In this model, some first-stage decisions alter the probability distributions of uncertain parameters, whereas others do not. The developed model integrates several supervised machine learning algorithms to estimate probability distributions of uncertain parameters. This work introduces a mathematical reformulation to linearize the non-convex model and an accelerated decomposition algorithm to solve the model.

The third work of this dissertation (Chapter 4) studies the submodularity of the SPEU framework with an application to complex systems reliability, where component lifetime is uncertain and depends on the investment amount. We prove the submodularity of the SPEU framework's objective function for probability distributions commonly used in complex system reliability allocation problems. Utilizing the submodularity property, we implement a continuous approximation algorithm for solving large-scale problems.

Finally, chapter 5 summarizes the conclusions of this dissertation and presents possible future research directions to extend this research.

Chapter 2

A Stochastic Programming Model with Endogenous Uncertainty for Incentivizing Fuel Reduction Treatment Under Uncertain Landowner Behavior

This chapter and Appendix A are based on the paper published by Tanveer Hossain Bhuiyan, Maxwell C. Moseley, Hugh R. Medal, Eghbal Rashidi, and Robert K. Grala:

> Bhuiyan, T. H., Moseley, M. C., Medal, H. R., Rashidi, E., and Grala, R. K. (2019). A stochastic programming model with endogenous uncertainty for incentivizing fuel reduction treatment under uncertain landowner behavior. *European Journal of Operational Research*, 277(2): 699–718.

Authors Medal and Grala posed the research problem. Authors Bhuiyan and Medal developed the mathematical models and solution approaches. Author Bhuiyan implemented the models and the solution algorithms. Authors Rashidi and Grala collected the data. Authors Bhuiyan and Moseley conducted the computational experiments, wrote the manuscript and created the figures and tables. Authors Medal, Rashidi and Grala edited the manuscript. During the three revisions in the peer-review process, author Bhuiyan

addressed the reviewers' comments, conducted the computational experiments, and revised the manuscript. Author Medal reviewed and edited the revised manuscript and the responses to reviewers' comments.

2.1 Introduction

This chapter presents a new data-driven stochastic programming model with endogenous uncertainty for a wildfire risk reduction problem. This study addresses the problem of designing a cost-share program that incentivizes nonindustrial private forest (NIPF) landowners to implement a fuel reduction treatment on their lands, resulting in a comprehensive reduction in the damage and severity of wildfires. Specifically, the objective is to develop a decision-making model that land management agencies can use to efficiently allocate limited cost-share resources to NIPF landowners to mitigate the potential damage from future wildfires. This study considers two primary factors in developing these strategic cost-share assistance levels: 1) the wildfire risk NIPF lands pose to their surrounding landscape and 2) the cost-share budget restrictions of agencies. The goal of this study is to i) develop an optimization method that generates strategic financial cost-share assistance levels for agencies to present to NIPF landowners, ii) understand how risk-based allocation reduces wildfire damage compared to other allocation strategies, and iii) provide experimental results and managerial insights to help agencies evaluate alternative solutions.

2.1.1 Motivation

Wildfires continue to devastate extensive areas throughout the United States each year, mostly in the western United States, with annual suppression costs climbing into the billions of dollars for state and federal agencies [23]. In response to wildfires and other natural occurrences, some agencies, mostly in the southeastern United States have begun offering cost-share programs in which the agency agrees to cover a portion of the costs associated with implementing required management practices for a landowner in exchange for the landowner's involvement in the program. Examples include the Mississippi Forestry Commission's program designed to enhance forest improvement strategies [100], and the

United States Forest Service Collaborative Forest Landscape Restoration Program, which encourages implementation of fuel reduction treatments [137]. Haines et al. [49] reported that almost 50% of wildfire hazard mitigation programs in the United States involved cost-share incentives.

The inherent challenge with these cost-share programs, however, is motivating landowners to participate. In regions with mixed land ownership, such as the southern United States, the process of implementing and coordinating landscape-level forest fuel reduction treatments might be challenging due to the diversity of involved stakeholders, their different forest management objectives, and their different perceptions of wildfire risk prevention costs and expected wildfire mitigation benefits. Thus, we seek to optimize the effectiveness of these programs by developing cost- and resource-effective strategies that land managers can adopt in incentivizing private landowners to implement fuel reduction treatments and thereby reduce hazardous fuel loads across the landscape.

Fuel reduction treatments help to reduce the severity and risk of wildfires in a large landscape. Fuels represent live and dead biomass that can burn in wildfire [134]. Typically, fuels include litter, branches, and shrubs as well as small and large trees that can facilitate the spread of wildfire both horizontally and vertically [2]. Fuel reduction treatments rely on the premise that breaking continuity of accumulated fuels by removing unwanted dead and live vegetation will reduce accumulation of hazardous fuels and thus change wildfire behavior by reducing its spread and intensity [139]. Fuel reduction methods vary depending on fire management objectives and public acceptance [59]. However, they commonly include mechanical vegetation control, chemical vegetation control, and purposeful use of prescribed fire [134, 2]. In the southern United States, prescribed burning is conducted as a fuel reduction treatment; however, due to some reasons such as social resistance, prescribed burning is not conducted in many other states, where mechanical treatments are usually performed. A strategic placement of fuel reduction treatments can influence spread of wildfire [43], especially in non-severe fire seasons, as well as improve fire suppression efforts by providing better access to the main fire, facilitating more effective communication within the firefighting crew, and enabling more effective suppression of spot fires [101], thereby limiting associated damages.

The effectiveness of the fuel reduction treatments in mitigating wildfire damages and the challenges in cost-share programs encourage us to study the problem from the perspective of the agencies seeking to efficiently allocate resources among NIPF landowners to mitigate wildfire risk through fuel reduction treatments. In this research, we aim to use risk to inform the allocation of these agencies' cost-share resources among NIPF landowners. By assessing the risk, economic impact, and wildfire susceptibility of land parcels in a landscape, our optimization model can estimate the efficacy of different cost-share budget allocations and determine the most effective configuration that minimizes the wildfire damage to the landscape. By tailoring resource allocation levels to landowner preferences, agencies can increase the likelihood that NIPF landowners will participate in the cost-share program to implement fuel reduction treatments, reduce the damage from wildfire, and optimize the use of their resources.

2.1.2 Related Literature

Several research teams have applied operations research tools to provide decision-making support for minimizing damage from wildfires. Minas et al. [98], and Martell [91] provide a detailed review of the application of operations research techniques to wildland fire management. Bettinger [14] provides a review of the literature on the use of operations research tools to develop efficient forest planning by incorporating the impact of wildfires.

To reduce the severity and damage from wildfires, research studies have proposed operations research methods for planning fuel reduction treatment activities. Fuel reduction treatments can reduce the accumulation of flammable forest fuels and can decrease the likelihood of wildfire occurrence in a large landscape as well as limit the extent of wildfire damage. Finney [43] documented that optimal selection of the locations of fuel reduction treatments can be more efficient in reducing the fire spread than performing fuel reduction treatments at random locations. Researchers have introduced various methodologies, tools, and techniques to develop efficient fuel reduction treatment plans. Ager et al. [3] employed a simulation methodology to provide a comparison of different alternative fuel reduction treatment strategies utilizing the distance to residential structures and stand density to prioritize fuel reduction treatment areas. They simulated the fuel reduction treatment on a

wildland-urban interface landscape in Oregon, U.S. Finney [43] proposed a methodology to determine major paths of fire travel and the fuel reduction treatment patterns to reduce the spread of fire in a heterogeneous landscape. Higgins et al. [56] modeled a seasonal resource allocation problem using an integer programming model to develop fuel reduction strategy in public lands. Rashidi et al. [120] implemented fuel reduction treatment as a mitigation approach to reduce the impact of worst-case wildfire scenario in a pyro-terror attack.

Thompson and Calkin [136] suggested that optimization approaches are suitable for evaluating a large number of alternative fuel reduction treatments, whereas simulation methodologies are suitable for modeling fire behaviors. Therefore, many researchers have adopted optimization approaches or integrated simulation-optimization approaches to optimally allocate resources in fuel reduction treatments to reduce the damage from wildfires. Hof et al. [58] proposed a timing-oriented programming model to perform spatial allocation of fuel reduction treatments that reduces the speed of a particular fire spread to protect specified land areas. In an extension of this study Hof and Omi [57] developed another timing-oriented linear programming model for scheduling fuel management that determines fuel reduction treatments to mitigate the effects of a particular fire with known origin and spread behavior. Wei et al. [144] proposed a mixed-integer programming model that optimally allocates fuel reduction treatments across a landscape to break paths of fire probability accumulation. Minas et al. [97] presented an integer programming model incorporating both fuel management and suppression preparedness decisions to provide an integrated planning framework for wildfire damage reduction.

Several sources of uncertainty are associated with fuel treatment management. Decisions made based on specific parameter values may perform poorly under different values. Therefore, some studies addressed the uncertainty in the underlying parameters of the fuel reduction treatment problem. Acuna et al. [1] considered uncertainty in wildfire occurrences and spread in their integrated fire and forest management planning model that provides scheduling of timber harvests to protect valuable forest from future fires. Minas et al. [99] presented a mixed-integer programming model that incorporates heterogeneous landscape, ecological and operational aspects to provide decision support for multi-period scheduling of fuel reduction treatments. The authors proposed that fuel treatment planning be conducted

each year using a rolling horizon approach to account for uncertainty in the effectiveness of fuel reduction treatments. In this approach, the landscape status is updated in each period. Rachmawati et al. [114] extended the work of Minas et al. [99] to account for multiple vegetation types in determining an optimal schedule and location for fuel reduction treatments.

Instead of a rolling horizon approach, some studies have applied stochastic programming approach to deal with parameter uncertainty, where scenarios are constructed to explicitly represent different values of uncertain parameters and decisions are made considering all the scenarios simultaneously. While stochastic programming approach provides solutions that are robust in the presence of uncertainty, constructing the scenarios is a non-trivial task and solving these problems presents computational challenges. Wei [142] presented a two-stage model that addresses uncertainty in ignition location and duration of wildfires in determining the optimal fuel reduction treatment layout in a landscape. Wei and Long [143] modified Wei [142] model to account for the worst-case wildfire scenario that generates fuel breaks in high fire hazard fuel patches. Rytwinski and Crowe [122] integrated simulation with a stochastic optimization framework that addresses the uncertainty in the location of fire ignition and spread direction. This framework minimizes the risk of fire using fuel-breaks under limitation on the area to be used for fuel-breaks. The authors used a heuristic approach to solve their problem, which can only guarantee a near-optimal solution. Konoshima et al. [70] proposed a stochastic dynamic programming model accounting for uncertainty in fire ignition locations and weather conditions to determine the optimal spatial patterns of fuel treatments and timber harvesting on a small stylized landscape. Their work was further extended by Konoshima et al. [69] where the authors also used a stochastic dynamic approach to explore insights regarding the decisions of fuel reduction treatment and harvest for different economic, topological, and weather conditions. Another application of stochastic dynamic programming is found in Ferreira et al. [40] in developing the optimal scheduling of fuel reduction treatment and harvesting under the risk of wildfire.

2.1.3 Contributions

In summary, no research has studied the cost-share resource allocation problem in implementing fuel reduction treatments on privately owned lands. The existing studies on fuel reduction management focused mainly on optimizing the spatial patterns of fuel reduction treatments. The uncertainties addressed in some of those studies include fire ignition locations, spread direction, weather condition, and effectiveness of fuel treatments undertaken. To the best of our knowledge, no research study has addressed uncertainty in landowner behavior in modeling a cost-share program for fuel reduction treatments.

The major contribution of this research is that it studies the cost-share budget allocation problem under uncertainty in landowner behavior, which is a practical and challenging problem often faced by land managers in implementing fuel reduction treatments on privately owned lands. While other research has studied fuel reduction treatment optimization from a spatial point-of-view, our research accounts for the uncertainty in human behavior by considering uncertainty in the response of private landowners to a cost-share offer. The uncertainty is *decision–dependent* (endogenous) as the landowner's response to a cost-share offer depends on the offer amount. We address the uncertainty in fire ignition locations by considering random ignition locations in the landscape.

We propose a simulation-optimization integrated approach for optimizing cost-share resource allocation under uncertainty to reduce the expected damage from wildfires. Our proposed approach provides the basis for a decision support system for land managers or government agencies offering cost-share assistance to private landowners to encourage them to implement a fuel reduction treatment through a cost-share program. The stochastic programming model considers uncertainty in the effect of cost-share assistance offer amount to the decision of the landowners to accept or reject the cost-share program. In this decision–dependent uncertainty, the likelihood of a landowner accepting a cost-share offer increases as the amount of the cost-share offer increases.

This research extends the existing wildfire management literature by introducing new mathematical models that consider reducing the expected damage from a wildfire by optimally allocating cost-share assistance to private landowners. Specifically, in this research

we have made the following contributions: (1) proposed a risk-based cost-share resource allocation strategy to efficiently coordinate implementation of fuel reduction treatments on private lands that minimizes the overall wildfire damage, (2) developed stochastic programming models with endogenous uncertainty to optimally allocate cost-share assistance to private landowners to encourage implementing fuel reduction treatments, (3) introduced a predictive modeling technique to estimate the likelihood that landowners accept or reject the cost-share offer, (4) provided experimental results to demonstrate the effects of model parameters and different cost-share allocation strategies on the optimal solution and runtime, and the sensitivity of the risk-based model to the number of cost-share allocation levels used in the model.

2.2 Problem Description

We study a problem from the perspective of an agency seeking to reduce the potential damage resulting from wildfire by providing monetary incentives to private landowners to implement fuel reduction treatment on their land through a cost-share program. We assume that landowners accepting a cost-share offer implement a fuel reduction treatment on all of their land parcels. However, this is a simplifying assumption of a more practical fuel reduction treatment scheme where landowners often implement treatment on a subset of their land parcels depending on the vegetation types, ages, and wildfire risk posed by the land parcels. We assume that the rate of fire spread and intensity in a land parcel subject to fuel reduction treatment is reduced and the reduced intensity fire can be stopped using available fire suppression resources (see [143] for example). The agency has a limited budget to provide cost-share assistance to the private landowners. The agency follows a risk-based budget allocation strategy, where rather than offering landowners the same cost-share amount per acre, the agency chooses the cost-share resource allocation that results in the maximum risk reduction in the overall landscape of interest. This is in contrast to a typical cost-share program in which agencies usually offer equal amount of cost-share assistance to all landowners (uniform allocation). A variation of this typical strategy is one in which the agency chooses which landowners to offer cost-share assistance and then offers

a predetermined amount to all chosen landowners (hybrid allocation). Our risk-based model allows us to measure the effects of using a risk-based allocation. This research compares these three strategies to measure their effectiveness in reducing wildfire risk.

A key feature of the problem studied in this research is the uncertainty in landowner behavior with respect to their decision of whether or not to accept a cost-share offer. Specifically, the agency does not know whether or not a landowner will accept an offer for a given amount of cost-share assistance. However, the agency can estimate the probability that a landowner will accept a given cost-share amount, for example, using historical or survey data. The objective of the agency is to minimize the expected total area burned due to wildfire. We refer to the problem as Stochastic Risk-based Allocation Problem for Fuel Reduction Treatments (SRAPFRT). To study the problem, we consider a landscape divided into a number of grid cells, each cell owned by a landowner. Each cell has various biophysical characteristics, and a probability of fire ignition, which, in this study is considered equal for all cells, because of lack of historical data.

We model the three cost-share budget allocation problems as two-stage stochastic programs where the first-stage consists of the agency allocating its budget among the landowners. The scenarios represent which landowners accept or reject the cost-share program.

The second-stage models the spread of wildfire in the landscape and computes the damage from wildfires in each scenario. We model the spread of fire using a simulation program that includes the use of FlamMap and the Monte Carlo simulation. The simulation program takes the physical characteristics of each cell of the landscape, information about the locations where fuel breaks are implemented, and location of fire ignition and then outputs which cells are burned as well as the extent of damage. Details of this simulation program are described in section 2.3.4.

In modeling the spread of wildfire, we have made the following assumptions: (1) locations of fire ignition points are randomly distributed across the landscape, (2) wildfires can ignite at multiple locations of the landscape at the same time, and prevail for the same duration under the same fire weather conditions, (3) the shape of the fire spread is elliptical, and (4) once the wildfire reaches the center of a cell, the cell is burned.

2.3 Methods

This section details the methodology for solving the SRAPFRT problem. We formulate the problem as a stochastic integer nonlinear programming model and implement a reformulation technique to linearize the original nonlinear model to obtain a mixed-integer linear stochastic programming model. We also implement a predictive modeling technique to estimate the probabilities of the landowners' decision states for different cost-share allocation levels and compute each scenario's second-stage value. To evaluate our methodology on a test landscape for different number of landowners, we reassign the ownership of land parcels in the test landscape among the landowners. These methodologies are described in detail in the following sub-sections.

2.3.1 Mathematical Formulation

We formulate the problem of optimally allocating cost-share assistance to the private landowners as a stochastic programming model with decision-dependent uncertainty. In addition to the mathematical formulation of the stochastic risk-based cost-share assistance allocation problem, we present the mathematical formulation of the uniform and the hybrid cost-share allocation problems.

Before presenting the mathematical formulations, the necessary sets, parameters, and variables are listed in Table 2.1.

Risk-based Allocation Model

In this sub-section, we present the two-stage stochastic programming model with decision-dependent uncertainty for the stochastic risk-based cost-share assistance allocation problem to minimize the expected area burned. The first-stage model decides on the optimal amount of cost-share assistance to be offered to each landowner so that the expected damage will be minimized. We refer to the first-stage model as RA-NLP. The second-stage model computes the damage due to wildfire in each scenario for a particular accept/reject decision states of the landowners.

Table 2.1: Notation.

(a) Sets

Parameters	Description
$\mathcal{J}$	Set of landowners indexed by j
$\mathcal{K}$	Set of cost-share assistance levels indexed by k
$\mathcal{L}$	Set of decision states of the landowners
Ω	Set of scenarios indexed by ω

(b) Parameters

Parameters	Description
c_k	Amount of cost-share assistance per acre at level k
a_j	Total area (acre) belonging to landowner j
B	Total cost-share budget of the agency
g^ω	The second-stage objective value for scenario ω

(c) Variables

Variables	Description
y_{jk}	1 if k level of cost-share assistance is offered to landowner j, 0 otherwise
w^ω_{jk}	The probability-weighted damage in a scenario
x_k	1 if k level ($k \geq 1$) of cost-share assistance is offered to the landowners chosen to offer cost-share assistance, 0 otherwise

In our stochastic programming model, the decision of a landowner to accept or reject a cost-share offer is uncertain. We construct discrete scenarios to represent the landowners' random decision states, which are to accept or reject a cost-share offer. Each scenario represents a particular combination of the landowners' decision states. The second-stage objective value is computed for each of these scenarios by simulating the spread of fire in the landscape, which is discussed in detail in Section 2.3.4.

The decision of a landowner j to accept or reject a cost-share program in a scenario ω depends probabilistically on the amount of cost-share assistance offered to that landowner. Let's consider the landowners' decision states (accept or reject) denoted by the random vector $\mathbf{d} = (d_j)_{j \in \mathcal{J}}$, whose element d_j, corresponding to landowner j, depends on the amount of cost-share assistance offered to landowner j. Therefore, the conditional probability function that landowner j is in decision state d_j for a given cost-share assistance amount y_j is $f_j(d_j^\omega \mid y_j)$. We assume that the landowners decide about the cost-share offer independently. As possible combinations of the landowners' decision states are represented by a finite set of scenarios, Ω, where $\omega \in \Omega$ defines a particular combination of all the landowners' decision states, the probability of scenario ω can be defined as:

$$\mathbb{P}^\omega(\mathbf{y}) = \prod_{j \in \mathcal{J}} f_j(d_j^\omega \mid y_j) \tag{2.1}$$

We can discretize the cost-share assistance amounts into $|\mathcal{K}|$ different levels and define the landowners' decision states as follows:

- $\mathcal{K} := \{0, 1, 2, ..., k\}$ is the set of cost-share assistance levels indexed by k.
- $\mathcal{L} := \{0, 1\}$ is the set of decision states of the landowners indexed by ℓ, where $\ell = 0$ stands for rejecting and $\ell = 1$ stands for accepting a given cost-share offer.

Using the discretized sets of cost-share assistance levels and landowners' decision states, the probability of scenario ω can be expressed as follows:

$\mathbb{P}^\omega(\mathbf{y}) = \prod_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} \mathbb{P}_{j\ell(j,\omega)k} y_{jk}$, where $\mathbb{P}_{j\ell(j,\omega)k}$ is the probability that the landowner's decision state is ℓ in scenario ω, given that landowner j is offered k level of cost-share assistance. In this research, we consider a heterogeneous landscape, where the

lands belonging to different landowners have different topography and fuel conditions characteristics that cause different wildfire behavior and risk. Based on the impact of land parcels on wildfire risk, the agency offers different amounts of cost-share assistance to the landowners. In this setting, the decisions of the landowners could be independent of each other.

The decision-maker can estimate the probability $\mathbb{P}_{j\ell(j,\omega)k}$ that landowner j will accept the cost-share offer for k level of cost-share assistance. In this research, we use a predictive modeling technique to estimate the probabilities of a landowner accepting a cost-share offer for a given amount of cost-share assistance. The predictive modeling technique for estimating probabilities is described in Section 2.3.3.

The first-stage of our two-stage risk-based stochastic programming model (2.2) is presented below (RA-NLP):

$$\min \quad \sum_{\omega\in\Omega} \mathbb{P}^{\omega}(\mathbf{y})\, g^{\omega} \tag{2.2a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}} a_j c_k y_{jk} \leq B \tag{2.2b}$$

$$\sum_{k\in\mathcal{K}} y_{jk} = 1 \qquad \forall j \in \mathcal{J} \tag{2.2c}$$

$$y_{jk} \in \{0,1\} \qquad \forall j \in \mathcal{J}, k \in \mathcal{K} \tag{2.2d}$$

The objective function (2.2a) seeks to minimize the expected damage over all scenarios. The total cost of allocating different cost-share assistance levels to all the landowners cannot exceed the agency's cost-share budget (2.2b). Each constraint (2.2c) ensures that only one level of cost-share assistance can be allocated to each landowner. The binary nature of the cost-share allocation decision is represented by the constraints (2.2d).

Unlike a traditional stochastic programming model, in our stochastic programming model with decision-dependent uncertainty, computation of the second-stage value does not depend on the first-stage decision variables. The inputs to our second-stage model are the "accept/reject" decision states of each landowner, which define the scenarios. Rather than determining the "accept/reject" decisions, the first-stage model decides how much cost-share

assistance to offer each landowner. In this way, second-stage is only indirectly tied to the first-stage in that the probability of the "accept/reject" decision state of a landowner depends on the cost-share amount offered to that landowner. As we can compute the spread of wildfire for a given scenario without knowing the probability of that scenario, we can compute the second-stage values *a priori* without solving first-stage model and then consider these values as known model parameters.

We see that the probability of a scenario (Eq.2.1) contains the product of the cost-share allocation decision variables for the landowners. Therefore, the objective function (2.2a) of the first-stage model (2.2) contains the product of the decision variables, which makes the model nonlinear and non-convex. As nonlinear and non-convex models cannot be solved using off-the-shelf mixed-integer programming solvers such as Gurobi [48], this model is reformulated as a mixed-integer linear (MILP) programming model using the procedure described in Section 2.3.2.

Uniform Allocation Model

The risk-based model allocates cost-share assistance based on the impact of land parcels in the landscape on wildfire hazard. Some parcels are more densely vegetated than others and therefore more susceptible to a wildfire, or have higher economic value than other parcels, whether due to standing timber and their proximity to populations or the forest and agricultural activities occurring there. By emphasizing these differences in impact and risk, the risk-based model can prioritize these parcels over other parcels with less impact on the landscape, and therefore offer higher cost-share assistance to the owners of these high-priority parcels. This allocation strategy could be controversial, however, as in a typical cost-share program all landowners receive an equal amount of cost-share assistance per unit of enrolled land area. Therefore, we model the uniform cost-share allocation problem where all the landowners receive the same level of cost-share assistance to provide a comparison with our risk-based allocation model in their effectiveness to mitigate the wildfire risk. The stochastic uniform cost-share assistance allocation model (2.3) is presented below (UA-NLP):

$$\min \quad \sum_{\omega \in \Omega} \mathbb{P}^{\omega}(\mathbf{y})\, g^{\omega} \tag{2.3a}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} a_j c_k y_k \leq B \tag{2.3b}$$

$$\sum_{k \in \mathcal{K}} y_k = 1 \tag{2.3c}$$

$$y_k \in \{0, 1\} \qquad \forall k \in \mathcal{K} \tag{2.3d}$$

The objective function (2.3a) minimizes the expected wildfire damage over all scenarios by providing the same level of cost-share assistance to the landowners. Constraint (2.3b) ensures that the total cost of allocating k level of cost-share assistance to all the landowners does not exceed the agency's budget. Constraint (2.3c) ensures that only one level of cost-share assistance is offered to all landowners. Constraint (2.3d) represents the binary nature of the cost-share allocation decision variables.

Hybrid Allocation Model

In a variation of the typical cost-share allocation strategy, a subset of landowners are chosen to be offered a predetermined level of cost-share assistance while remaining landowners are not offered any cost-share assistance. This hybrid (semi-uniform) cost-share allocation strategy is more pragmatic than the uniform allocation. Because, rather than just providing assistance to all landowners, it might be more practical for the agencies to select some potential landowners to offer assistance and thereby more efficiently use their resources to mitigate wildfire risk. Therefore, we model this hybrid allocation problem to provide a comparison with our risk-based allocation model. The stochastic hybrid cost-share assistance allocation model (2.4) is presented below (HA-NLP):

$$\min \quad \sum_{\omega \in \Omega} \mathbb{P}^{\omega}(\mathbf{y})\, g^{\omega} \tag{2.4a}$$

$$\text{s.t.} \quad y_{jk} \leq x_k \qquad \forall k \in \mathcal{K} \setminus \{0\} \tag{2.4b}$$

$$\sum_{k \in \mathcal{K} \setminus \{0\}} x_k \leq 1 \tag{2.4c}$$

$$x_k \in \{0,1\} \qquad \forall k \in \mathcal{K} \setminus \{0\} \tag{2.4d}$$

$$(2.2b) - (2.2d)$$

The objective function (2.4a) seeks to minimize the expected damage over all scenarios by optimally allocating the cost-share assistance to the landowners. Constraints (2.4b) and (2.4c) together ensure that all landowners in the subset of landowners chosen to offer cost-share assistance are offered the same k level of cost-share assistance.

2.3.2 Reformulation

The first-stage objective function of our risk-based stochastic programming model (2.2) can be written as:

$$\min \sum_{\omega \in \Omega} g^{\omega} \left(\prod_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} \mathbb{P}_{j\ell(j,\omega)k} y_{jk} \right) \tag{2.5}$$

Equation (2.5) minimizes the total probability-weighted wildfire damage over all scenarios. To linearize the nonlinear objective function, we adopt a probability-chain reformulation technique proposed in Medal et al. [94]. In this reformulation technique, the probability-weighted damage is quantified for each scenario using a set of recursive equations. These recursive equations replace the requirement of the product term over the landowners to compute the probability of a scenario.

As we assume that the probabilities of the decision states of the landowners are independent, we can substitute the product term $\prod_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} \mathbb{P}_{j\ell(j,\omega)} y_{jk}$ with a recursive expression using bookkeeping variables to store the product of the probabilities and the

second-stage value. Assume that w^{ω}_{rk} is a variable that stores the product of the second-stage value for scenario ω and the probabilities that the landowners $1, 2, ..., r$ are in their corresponding decision states in scenario ω. The following recursive equations are used to compute the value of w^{ω}_{rk} for each scenario ω.

$$w^{\omega}_{1k} = g^{\omega}\mathbb{P}_{1,\ell(r,\omega),k}y_{1k} \quad \forall\omega \in \Omega, k \in \mathcal{K} \tag{2.6}$$

$$\sum_{k\in\mathcal{K}} w^{\omega}_{r-1,k} = \sum_{k\in\mathcal{K}} \frac{1}{\max\left\{\mathbb{P}_{r,\ell(r,\omega),k}, \epsilon\right\}} w^{\omega}_{rk} \quad \forall r = 2, \ldots, |\mathcal{J}|; \omega \in \Omega \tag{2.7}$$

$$w^{\omega}_{rk} \leq g^{\omega}y_{rk} \quad \forall r \in \mathcal{J}, k \in \mathcal{K}, \omega \in \Omega \tag{2.8}$$

Equation (2.6) computes the value of the w^{ω}_{rk} variable for the first landowner in scenario ω. The variable w^{ω}_{1k} holds the product of the second-stage value for scenario ω and the probability that the first landowner is in their corresponding decision state in scenario ω. The value of w^{ω}_{rk} for each subsequent landowner is computed by equations (2.7). For the second landowner in scenario ω, the w^{ω}_{2k} variable holds the product of the value of the w^{ω}_{1k} variable corresponding to the first landowner and the probability that the second landowner is in their corresponding decision state. In this way, the value of w^{ω}_{rk} is computed recursively for each landowner until the last landowner. When this process reaches the last landowner, the variable w^{ω}_{rk} becomes equal to $g^{\omega}\left(\prod_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\mathbb{P}_{j\ell(j,\omega)k}y_{jk}\right)$. In the denominator of the fraction on the right side of equation (2.7), we take the maximum between the probability $\mathbb{P}_{r,\ell(r,\omega),k}$ and a very small positive number ϵ, to avoid division by zero, as in some instances the probability for a landowner to be in a decision state might be zero.

Equation (2.8) ensures that the value of w^{ω}_{rk} depends on the allocation decision variables y_{jk}. The value of w^{ω}_{rk} for landowner j for a given allocation level k can be positive only when cost-share assistance level k is offered to landowner j.

The following numerical example demonstrates the computation of the probability-weighted damage in a scenario ω using the recursive equations (2.6)-(2.8). Consider a rasterized landscape with two landowners. The decision state of each landowner is either to accept or reject a given cost-share assistance offer. For simplicity, we consider only two

possible cost-share allocation levels, 0 and 1, with 0 meaning no cost-share assistance and 1 unit of cost-share assistance, respectively. Table 2.2 demonstrates the estimated probabilities that a landowner accepts or rejects a given cost-share assistance offer.

We assume that 1 unit of cost-share assistance is offered to the first landowner and no cost-share assistance is offered to the second, i.e., $y_{11} = 1$, $y_{20} = 1$. Also, in a given scenario ω, we assume that the area burned from wildfire is 500 acres when both landowners reject the cost-share offer. Now, according to equation (2.8), we have:

first landowner:

$$w_{11}^{\omega} \leq 500 y_{11}$$

second landowner:

$$w_{20}^{\omega} \leq 500 y_{20}$$

From the recursive equation (2.6), the probability-weighted damage corresponding to the first landowner in scenario ω is computed as follows:

$$w_{11}^{\omega} = \mathbb{P}^{\omega}_{1,\ell(1,\omega),1} y_{11} \times 500 \quad = 0.15 \times 500 = 75$$

Using the value of w_{11}^{ω} in the recursive equation (2.7), the probability-weighted damage corresponding to the second landowner in scenario ω is computed as follows:

$$w_{11}^{\omega} = \frac{1}{\mathbb{P}^{\omega}_{2,\ell(2,\omega),0}} w_{20}^{\omega} \quad \Rightarrow w_{20}^{\omega} = 0.90 \times 75 = 67.5$$

Therefore, the probability-weighted wildfire damage in scenario ω is 67.5. This recursive process computes the probability-weighted wildfire damage for each landowner starting with

Table 2.2: Estimated probability for a given allocation level

Allocation Level	Decision State	Estimated Probability
0	Rejects	0.90
0	Accepts	0.10
1	Rejects	0.15
1	Accepts	0.85

the first landowner and continues until it reaches the last landowner, where we obtain the total probability-weighted wildfire damage for that scenario.

The reformulated risk-based mixed-integer stochastic programming model (2.9) is stated as follows (RA-MIP):

$$\min \quad \sum_{\omega\in\Omega}\sum_{k\in\mathcal{K}} w^{\omega}_{|\mathcal{J}|k} \tag{2.9a}$$

$$\text{s.t.} \quad w^{\omega}_{1k} = g^{\omega}\mathbb{P}_{1,\ell(r,\omega),k}y_{1k} \qquad \forall \omega\in\Omega, k\in\mathcal{K} \tag{2.9b}$$

$$\sum_{k\in\mathcal{K}} w^{\omega}_{r-1,k} = \sum_{k\in\mathcal{K}} \frac{1}{\max\left\{\mathbb{P}_{r,\ell(r,\omega),k},\epsilon\right\}} w^{\omega}_{rk} \qquad \forall r=2,\ldots,|\mathcal{J}|;\omega\in\Omega \tag{2.9c}$$

$$w^{\omega}_{rk} \le g^{\omega}y_{rk} \qquad \forall r\in\mathcal{J}, k\in\mathcal{K}, \omega\in\Omega \tag{2.9d}$$

$$w^{\omega}_{rk} \ge 0 \qquad \forall \omega\in\Omega, r\in\mathcal{J}, k\in\mathcal{K} \tag{2.9e}$$

$$(2.2b)-(2.2d)$$

Similarly, we reformulate the uniform and hybrid cost-share allocation models into mixed-integer stochastic uniform (UA-MIP) (2.10) and hybrid (HA-MIP) (2.11) allocation models, respectively, which are presented below:

$$\min \quad \sum_{\omega\in\Omega}\sum_{k\in\mathcal{K}} w^{\omega}_{|\mathcal{J}|k} \tag{2.10a}$$

$$\text{s.t.} \quad w^{\omega}_{1k} = g^{\omega}\mathbb{P}_{1,\ell(r,\omega),k}y_{k} \qquad \forall \omega\in\Omega, k\in\mathcal{K} \tag{2.10b}$$

$$\sum_{k\in\mathcal{K}} w^{\omega}_{r-1,k} = \sum_{k\in\mathcal{K}} \frac{1}{\max\left\{\mathbb{P}_{r,\ell(r,\omega),k},\epsilon\right\}} w^{\omega}_{rk} \qquad \forall r=2,\ldots,|\mathcal{J}|;\omega\in\Omega \tag{2.10c}$$

$$w^{\omega}_{jk} \le g^{\omega}y_{k} \qquad \forall j\in\mathcal{J}, k\in\mathcal{K}, \omega\in\Omega \tag{2.10d}$$

$$w^{\omega}_{rk} \ge 0 \qquad \forall \omega\in\Omega, r\in\mathcal{J}, k\in\mathcal{K} \tag{2.10e}$$

$$(2.3b)-(2.3d)$$

$$\min \sum_{\omega\in\Omega}\sum_{k\in\mathcal{K}} w^{\omega}_{|\mathcal{J}|k} \tag{2.11a}$$

$$\text{s.t.} \quad (2.9b) - (2.9e)$$

$$(2.2b) - (2.2d)$$

$$(2.4b) - (2.4d)$$

After this linearization procedure, these mixed-integer stochastic programming models can be solved using off-the-shelf integer programming solvers such as the Gurobi optimizer [48].

2.3.3 Estimating the Probability of Decision States

We need to estimate the probability $\mathbb{P}_{j\ell(j,\omega)k}$ of landowner j to be in decision state ℓ in scenario ω for a given cost-share assistance level k. Each landowner's decision states are either to accept or reject the cost-share assistance offered by the agency. Therefore, we can use a binary logistic regression model to estimate the probability $\mathbb{P}_{j\ell(j,\omega)k}$ for a given level of assistance. In the logistic regression model, the response variable is the landowner's decision of either accepting (ℓ =1) or rejecting (ℓ = 0) the cost-share offer, for a given predictor variable (y_j), which is the amount (U.S. dollars) of cost-share assistance offered to landowner j. The estimated probability that a landowner j accepts ($\ell = 1$) the cost-share offer for a given cost-share assistance level k or given cost-share assistance amount y_j can be computed from the following equation:

$$\mathbb{P}_k = \frac{1}{1+exp[-(\beta_0+\beta_1 y_j)]} \tag{2.12}$$

where $\mathbb{P}_k$ is the probability of accepting the cost-share offer for k cost-share assistance level or y_j dollar amounts, and β_0 and β_1 are the parameter estimates of the logistic regression model.

We generate synthetic data based on the prescribed burning (fuel reduction treatment method in the southern United States) cost estimates obtained from a mail survey sent to 2000 randomly selected NIPF landowners. The survey was implemented by making four mail contacts [33]. Using the estimates of the prescribed burning costs per acre, we select a range of cost-share assistance amounts per acre based on which we generate the synthetic data. We build the logistic regression model and train it using the synthetic data that contain landowners' corresponding decision states (accept/reject) for different amounts (U.S. dollar) of cost-share assistance offered per acre (y_j). Thus, the prediction model builds a relationship between the dollar amounts offered per acre and the decision of a landowner. We discretize the entire range of the dollar amounts per acre offered by the agency to the landowners into $\mid \mathcal{K} \mid$ different levels and obtain the amounts c_k of cost-share assistance corresponding to the discrete cost-share assistance levels k.

We use our fitted logistic regression model to predict the probabilities of the decision states for given c_k values. The sample results of the logistic regression model showing the estimated probabilities corresponding to the cost-share assistance levels and dollar amounts are presented in Appendix A.5.

These probabilities $\mathbb{P}_k$ are used as $\mathbb{P}_{j\ell(j,\omega)k}$ in the stochastic programming model. That is, for the stochastic programming model, additional subscripts are needed for $\mathbb{P}$. Specifically, $\mathbb{P}_{j0(j,\omega)k}$ is the probability that landowner j rejects the cost-share offer in scenario ω given that c_k amount (U.S. dollars) or k level of cost-share assistance is offered to landowner j and $\mathbb{P}_{j1(j,\omega)k}$ is the probability that landowner j accepts the offer.

2.3.4 Computation of Second-Stage Values

We compute the wildfire damage for each scenario representing the decision states of the landowners of our stochastic programming model. To compute the damage, we simulate the spread of wildfire in a landscape that is divided into raster cells. The wildfire simulation procedure of this research is similar to the one used in Rashidi [117] and Rashidi et al. [119].

The center of each cell is considered as a node, and each node is connected to its neighboring nodes through directed arcs, which together model the landscape as a directed network. For details of the rasterized landscape and the corresponding directed network representation, see Rashidi et al. [118]. We consider multiple random nodes in the landscape where fire can ignite simultaneously. Based on the random choice of the ignition point location, numerous wildfire scenarios can occur, which are referred to as sub-scenarios in this research. Thus, in each scenario of our stochastic programming model, multiple wildfire sub-scenarios exist, and the average damage of these sub-scenarios is the damage, or second-stage value for that scenario.

We generate the wildfire sub-scenarios by randomly selecting the ignition points among the cells of the landscape. We see that the computation of average damage of all the sub-scenarios with three or more ignition points becomes computationally challenging. For example, with three random ignition points in the Santa Fe National Forest data, we have C_3^{625}, or more than 40 million wildfire sub-scenarios. Computation of damage for this large number of sub-scenarios in each scenario ω is computationally very expensive. Moreover, we have $2^{10} = 1024$ scenarios in our stochastic programming model for 10 landowners in the Santa Fe data. Therefore, the whole problem becomes computationally difficult if we consider all possible wildfires sub-scenarios. Therefore, we implement a Monte Carlo simulation (MCS) to generate and evaluate a random sample of 5,000 possible wildfire sub-scenarios with 5 ignition points, each with equal probability in each scenario ω and compute the average damage.

In this research, we use FlamMap [42] to compute the rate of fire spread (ROS) and the major fire spread direction (direction of fire with fastest speed) in each cell of the landscape. FlamMap takes different landscape characteristics such as elevation, slope, wind speed, wind direction, and moisture content as inputs and computes the rate of fire spread and the major fire direction. Besides the major fire spread direction, fire also spreads in other directions at lower speeds. We use formulas (2.13) and (2.14) to calculate the rate of fire spread along other minor directions.

$$ROS = \frac{\alpha^2 - \beta^2}{\alpha - \beta cos(\phi)} \quad 0 \leq \phi \leq \frac{\pi}{2} \tag{2.13}$$

$$ROS = \frac{\alpha^2 - \beta^2}{\alpha + \beta cos(\pi - \phi)} \quad \frac{\pi}{2} \leq \phi \leq \pi \tag{2.14}$$

Here, ϕ is the angle between the major fire spread direction in a cell and the fire spread direction from this cell to the center of the adjacent cells. α and β are the parameters, called foci, describing the ellipse of the fire spread and are computed by FlamMap. For details, see Rashidi et al. [118] and Green et al. [47].

In each scenario ω, we run FlamMap before simulating the fire spread in the landscape with the random ignition points to compute the major fire spread direction and the α and β parameters. We compute the rate of fire speed r_{mn} from cell m to one of its adjacent cells n using formulas (2.13) and (2.14). If we denote the distance between the two cells as d_{mn}, the time for the fire to spread from m to n is $T_{mn} = \frac{d_{mn}}{r_{mn}}$. We assume that if a landowner accepts a cost-share offer, the cells belonging to that landowner are subjected to fuel reduction treatment. We also assume earlier that after implementation of a fuel reduction treatment in a cell, the reduced intensity fire can be stopped using available fire suppression resources. Therefore, if cell m belongs to a landowner j who accepts the cost-share offer in a scenario ω of our stochastic programming model, the time T_{mn} for the fire to spread from cell m to n is modified to have a value of M, larger than the fire duration time D, so that the fire cannot reach from m to n. This way the treated cells prevent fire from spreading into the landscape and thus reduce the damage from wildfire. However, by manipulating the value of M, we can modify the effect of fuel reduction treatment on the rate of fire spread in our fire spread modeling. For example, the fire travel time or the value of M after implementation of a fuel reduction treatment can be made to be equal to the double of the fire travel time without fuel reduction treatment. This way our model can represent a more realistic fuel treatment effect.

We use the minimum travel time algorithm (MTT) proposed by Finney [41] to compute the minimum travel time path from each possible ignition point (cell) m to another cell q, and denote this minimum travel time as F_{mq}. If this minimum travel time F_{mq} is smaller

than the fire duration time D, we consider that the fire ignited at cell m can reach cell q and burn it.

Let $X_{mq} = 1$ if fire from cell m reaches cell q, and 0 otherwise. Therefore, if $F_{mq} \leq D$, $X_{mq} = 1$. Now, if we denote Q_q as the value of cell q that is lost when cell q is burned, then the total value lost in a wildfire sub-scenario can be computed as:

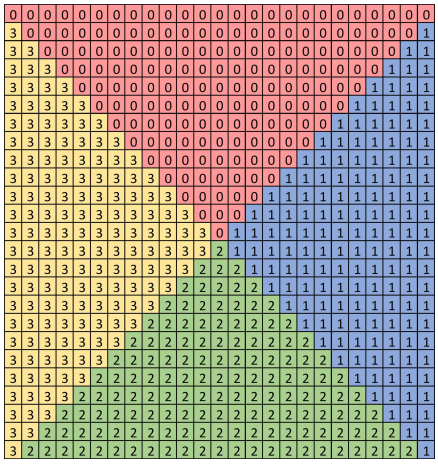
$$\sum_{m \in S'_n} \sum_{q \in S} Q_q X_{mq}$$

where S and S'_n denote the set of all cells in the landscape and the set of ignition points in the wildfire sub-scenario n, respectively. If we denote the total number of sub-scenarios as N, then the second-stage objective value (the total value lost due to wildfire) in each scenario ω of our stochastic programming model is computed as:

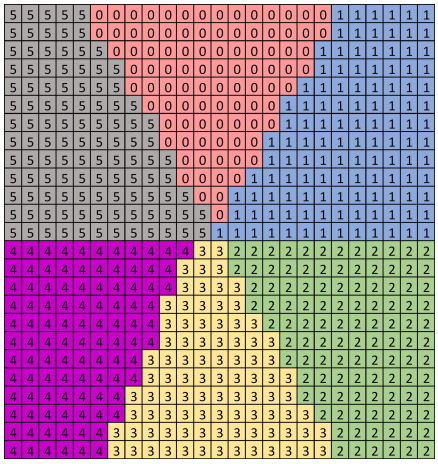
$$\frac{1}{N} \left(\sum_{n=1}^{N} \sum_{m \in S'_n} \sum_{q \in S} Q_q X_{mq} \right)$$

2.3.5 Test Landscape

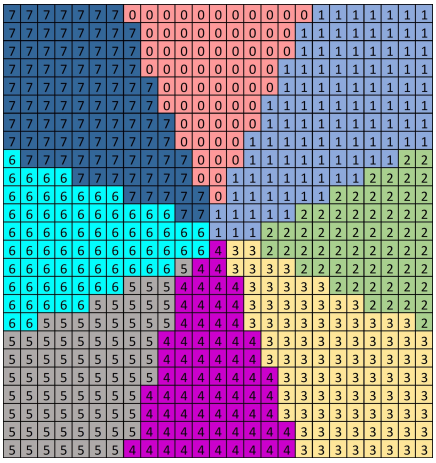
To evaluate the proposed model and solution approach, we ran numerical experiments using a test landscape based on the Santa Fe National Forest in New Mexico, in which there are 625 land parcels distributed among $|\mathcal{J}|$ landowners. Each land parcel has biophysical characteristics such as elevation, slope, and moisture content that are used to model the spread of fire through the landscape. For the purposes of these experiments, we reassigned the ownership of the 625 parcels among four, six, eight, and ten landowners, as shown in Figure 2.1. As a result, the number of acres owned by each landowner varies according to the number of landowners present in the landscape. Table A.1 in Appendix A.1 lists the acreages for each landowner in each version of the modified Santa Fe landscape.



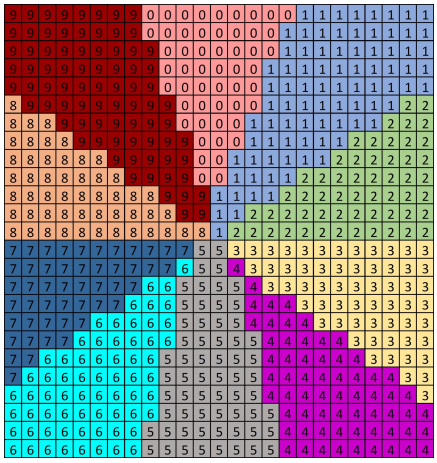
(a) Landscape with $|\mathcal{J}| = 4$ landowners



(b) Landscape with $|\mathcal{J}| = 6$ landowners



(c) Landscape with $|\mathcal{J}| = 8$ landowners



(d) Landscape with $|\mathcal{J}| = 10$ landowners

Figure 2.1: Land parcel ownership reassigned according to the number of landowners, $|\mathcal{J}|$, present in the landscape.

2.4 Computational Results and Discussion

In this section, we use our optimization models to provide insight into several questions.

1. How does the size of the agency's cost-share budget affect the expected damage of a wildfire?

2. How does the number of cost-share allocation levels offered by an agency impact the expected damage?

3. How sensitive is the mixed-integer stochastic risk-based allocation model to changes in the number of allocation levels modeled?

4. How does the risk-based cost-share assistance allocation strategy perform compared to the uniform and hybrid cost-share assistance allocation strategies?

2.4.1 Experimental Setup

We implemented the RA-MIP, UA-MIP, and HA-MIP models in Python 2.7 with Gurobi solver [48]. We also implemented the simulation program in Python 2.7 to model the spread of wildfire and compute the damage in each scenario. The logistic regression model estimating the probabilities of landowners' decision states is implemented using the scikit-learn package in Python 2.7.

We conducted numerical experiments for the Santa Fe landscape using the ownership scenarios as shown in Figure 2.1, for every combination of the parameter values in Table 2.3 to observe the effects of each parameter on the expected wildfire damage and runtime. Trials were run five times each and then averaged to account for sampling noise associated with computing the second-stage objective value. The results of all experiments are detailed in the following sub-sections.

The estimates that we obtain from the NIPF landowner survey for the minimum, maximum, and mean costs of prescribed burning per acre are \$1, \$35, and \$17.5, respectively. Based on these cost estimates, we select the range of per acre cost-share assistance amounts from \$0 to \$40 to generate our synthetic dataset. In our dataset, the minimum and maximum

Table 2.3: Parameters and their values used in the experiments

Parameter	Values Used
Number of landowners, $\lvert\mathcal{J}\rvert$	4, 6, 8, 10
Number of allocation levels, $\lvert\mathcal{K}\rvert$	2, 4, 6, 8
Budget, B	\$20,000, \$40,000, \$60,000, \$80,000, \$100,000

cost-share assistance offered to a landowner are \$0 and \$39.83 per acre, respectively. We discretize the continuous offer amounts into a number of discretized levels by dividing the range from \$0 to \$39.83 per acre by the number of cost-share allocation levels specified by the agency and then use the upper bounds of the resulting ranges as the amount (U.S. dollars) of cost-share assistance corresponding to the allocation levels. The cost-share assistance amounts associated with each level of cost-share allocation are shown in Table A.2 in Appendix A.1.

2.4.2 Runtime of the Solution Approach

To observe the solution runtime, we ran experiments on a computer running MacOS with Intel Core i9 @ 2.9 GHz with 16 GB of installed RAM. The runtimes of the risk-based allocation experiments are shown in Table A.3 in Appendix A.2 and are visualized in Figure 2.2, while Tables A.4 and A.5 in Appendix A.2 show the runtimes of the uniform and hybrid allocation experiments, respectively.

Based on the experimental runtimes in Figure 2.2 and Table A.3 in Appendix A.2, we see that the the size of the agency's cost-share budget has no clear correlation with the model runtime for four and six landowners. However, there are some spikes in the runtimes for certain budget levels for eight and 10 landowners. For instance, with eight landowners and six allocation levels, the runtime is higher at budget levels of \$40,000 and \$60,000, and with 10 landowners and four allocation levels, the runtime is substantially higher at budget levels of \$40,000 and \$60,000. These runtime spikes are likely due exploring an increased number of possible solutions necessitated by those particular parameter combinations. Similar to the effect of cost-share budget size, the number of allocation levels does not have a clear impact on the runtime for four and six landowners. However, model runtime increases for eight landowners as the number of allocation levels increases beyond $|\mathcal{K}| = 4$. Moreover, the effect of increasing number of allocation levels on runtime is more evident for 10 landowners; the runtime increases substantially as the number of allocation levels increases from $|\mathcal{K}| = 2$. Our solution methodology cannot solve the risk-based allocation problems for 10 landowners and six allocation levels within 6 hours, thereby yielding sub-optimal cost-share allocation

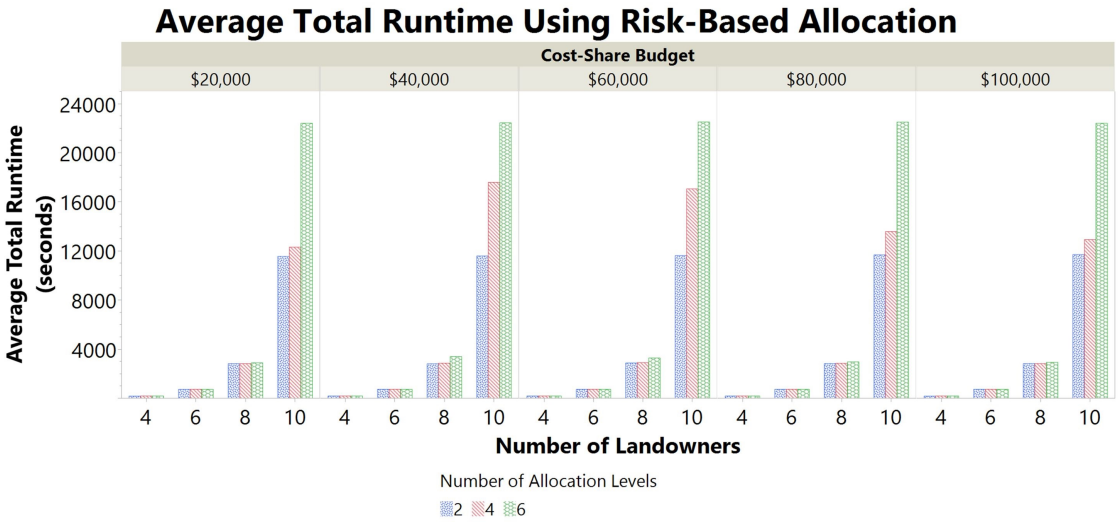


Figure 2.2: Impact of number of landowners on runtime using risk-based cost-share allocation.

decisions at the time of termination. We mark the timed-out parameter combinations with an asterisk (*) in Table A.3.

The number of landowners has a noticeable effect on the runtime of the model. As illustrated in Table A.3 and Figure 2.2, the model's runtime increases exponentially as the number of landowners increases. While increasing the cost-share budget might impact the allocation configuration and increasing the number of allocation levels adds additional variables to the model, increasing number of landowners adds both a complete set of new variables and scenarios to the model. Thus, the number of landowners has a far more considerable influence on the model's complexity and therefore runtime.

Figure 2.3 also indicates this trend for uniform and hybrid allocation strategies. However, both models' runtimes for each number of landowners are consistent across budgets and number of allocation levels, perhaps a result of the decreased complexity of those models compared to the risk-based model.

2.4.3 Effect of Budget on Expected Damage

Our expectation was that as the amount of cost-share assistance offered to NIPF landowners increases, the expected damage to the landscape will decrease, as the increased budget will allow the agency to offer higher amounts to each landowner, which will increase their probability of acceptance and thus incentivize more NIPF landowners to participate in the cost-share program.

After running these risk-based allocation experiments five times each and calculating the average results for each experiment, we plot the expected damage by a wildfire against the budget, in increments of $20,000 from $20,000 to $100,000. Figure 2.4 demonstrates the average expected damage at each budget amount across all allocation levels for each number of landowners.

From Figure 2.4 we can conclude that the model performs as expected, and that the expected damage decreases as the budget increases. Based on the convergence of all lines

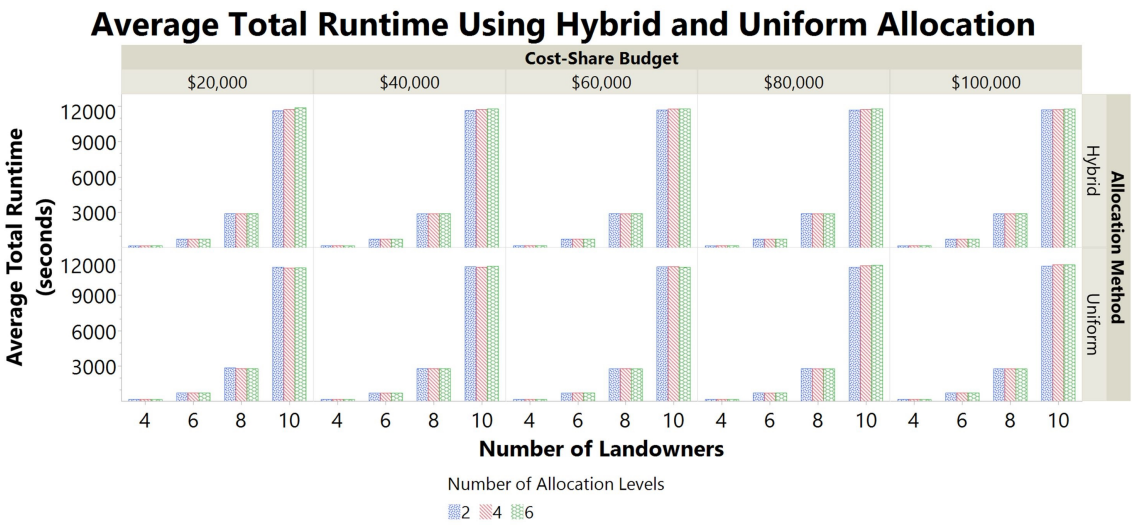


Figure 2.3: Impact of number of landowners on runtime using hybrid and uniform cost-share allocation strategies.

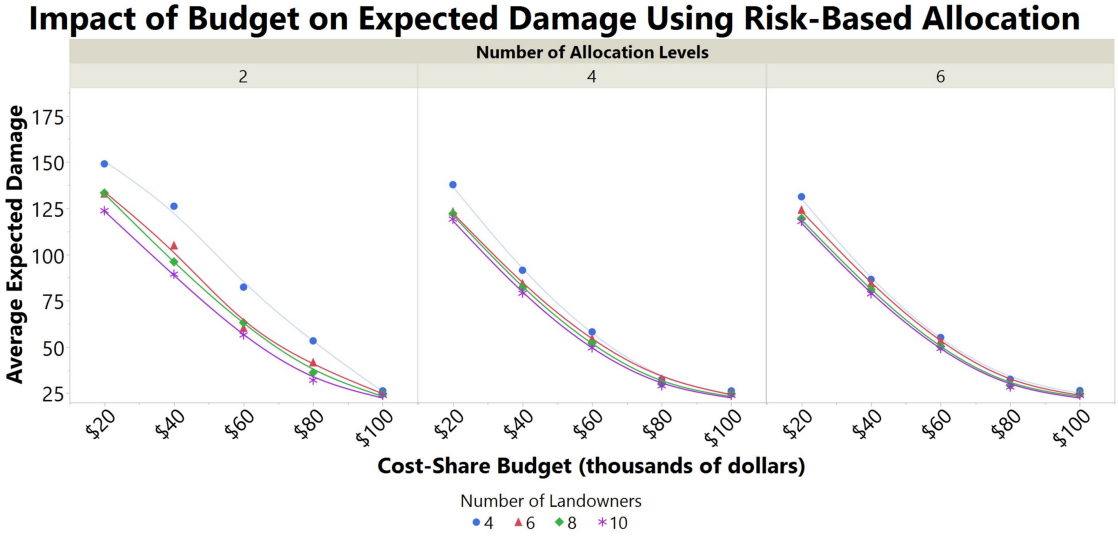


Figure 2.4: Impact of cost-share budget on expected damage using risk-based cost-share allocation.

once the budget reaches $100,000, Figure 2.4 also shows that the expected damage decreases as budget increases until the point at which the budget is large enough that all landowners receive the highest per-acre allocation level. For this Santa Fe landscape, the landowners own a total of around 2,161 acres. Using the model's maximum offer amount of $39.83 per acre, the budget necessary to offer all landowners the maximum per-acre offer amount is $86,120. Therefore, we can conclude that a budget that can afford the maximum per-acre offer amount for all acres in the landscape, such as $100,000, will produce the best solution. Furthermore, Table A.6 in Appendix A.3 shows that increasing the budget in increments of $20,000 from $20,000 to $100,000 provides, on average, a 33.18% improvement in the expected wildfire damage reduction to the landscape when using risk-based cost-share assistance allocation. Figure A.1 and Tables A.7 and A.8 in Appendix A.3 show the effect of budget on expected damage when using uniform and hybrid allocation strategies.

2.4.4 Effect of the Number of Allocation Levels on Expected Damage

Using our model, an agency can increase the number of allocation levels used by the model to explore the effect of increased model fidelity on the solution. To illustrate this capability, we analyze the sensitivity of our stochastic risk-based allocation model to changes in the number of cost-share allocation levels used. To measure how sensitive our model is to different values of the number of allocation levels $|\mathcal{K}|$, we consider a reference number of allocation levels, $|\mathcal{K}^*|$. We introduce a metric called relative change to measure the percent amount by which the expected damage changes if we use a different number of allocation levels $|\mathcal{K}|$ than the reference number of allocation levels $|\mathcal{K}^*|$. The relative change can be expressed as:

$$RC = \frac{Q(|\mathcal{K}^*|) - Q(|\mathcal{K}|)}{Q(|\mathcal{K}^*|)} \times 100\% \tag{2.15}$$

Here, $Q(|\mathcal{K}^*|)$ and $Q(|\mathcal{K}|)$ are the expected damages corresponding to using $|\mathcal{K}^*|$ and $|\mathcal{K}|$ number of cost-share allocation levels in the model. A positive value of the metric, relative change (RC) represents the percentage reduction in the expected damage due to

changing the number of allocation levels used in the model from $|\mathcal{K}^*|$ to $|\mathcal{K}|$. Tables 2.4a and 2.4b demonstrate the relative change in expected damage for different budget levels due to changing the number of allocation levels from $|\mathcal{K}^*| = 2$ and $|\mathcal{K}|^* = 3$, respectively for eight landowners.

Our expectation was that as the number of allocation levels increases, the offer amounts become more refined, leading to cost-share assistance offers that are more tailored to each NIPF landowner. As a result, we anticipate that the expected damage will decrease as the number of allocation levels increases. We see that for different budget levels, the expected damage decreases as the number of allocation levels increases from $|\mathcal{K}^*| = 2$ (Table 2.4a) and $|\mathcal{K}^*| = 3$ (Table 2.4b). This confirms our expectation that more number of allocation levels would allow the model to offer more accurate levels of cost-share assistance that would increase the probability of the landowners accepting the cost-share offer. However, the rate of improvement in the percent reduction of expected damage diminishes as the number of allocation levels increases. Table 2.4a reveals that the rate of improvement diminishes considerably after $|\mathcal{K}| = 4$ levels and falls below 1% (16.74-16.11) between $|\mathcal{K}| = 6$ and $|\mathcal{K}| = 8$ levels. While increasing the number of allocation levels allows the model to offer a more refined solution, beyond a certain point the number of allocation levels has a relatively small impact on the objective value, but may increase the model complexity and runtime substantively. Therefore, the decision-maker can consider a trade-off between model runtime and solution quality.

Figure 2.5 demonstrates the relative improvement in expected wildfire damage reduction against the relative increment in runtime as the number of allocation levels increases from $|\mathcal{K}^*| = 2$ for different budget levels. We see from Figure 2.5 that the runtime increases substantially beyond $|\mathcal{K}| = 6$ with a very small improvement in solution quality in all budget levels; especially the relative increment in runtime is very high at \$40,000 and \$60,000 budget levels (Figures 2.5b and 2.5c). Therefore, agencies can use a moderately large number of allocation levels, such as $|\mathcal{K}| = 6$, and the model will still be able to generate a sufficiently accurate risk-based allocation strategy that can effectively mitigate the impact of a wildfire on a landscape.

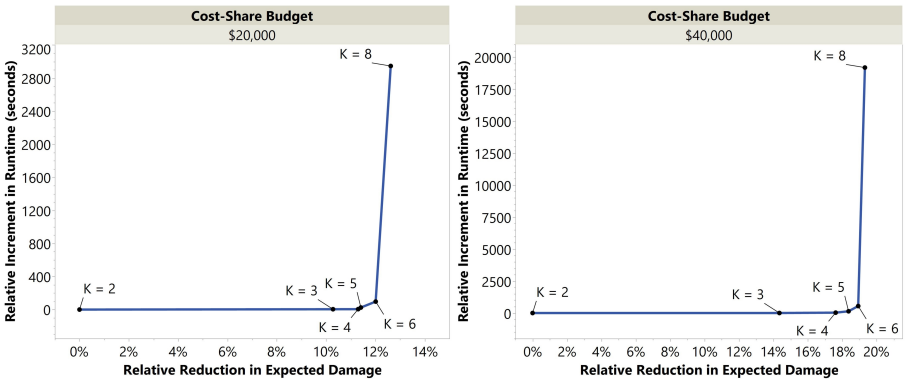
Table 2.4: Relative change for different number of allocation levels and budget.

(a) Relative change from number of allocation levels modeled $|\mathcal{K}^*| = 2$.

$\lvert\mathcal{K}\rvert$	Budget					Average
	\$20,000	\$40,000	\$60,000	\$80,000	\$100,000	
2	0%	0%	0%	0%	0%	0%
3	10.28%	14.35%	18.35%	16.52%	0%	11.90%
4	11.30%	17.62%	21.12%	20.96%	0%	14.20%
5	11.41%	18.37%	24.68%	21.31%	0%	15.15%
6	12.01%	18.93%	25.59%	24.03%	0%	16.11%
8	12.62%	19.32%	26.58%	25.19%	0%	16.74%

(b) Relative change from number of allocation levels modeled $|\mathcal{K}^*| = 3$.

$\lvert\mathcal{K}\rvert$	Budget					Average
	\$20,000	\$40,000	\$60,000	\$80,000	\$100,000	
2	-9.32%	-12.55%	-15.50%	-14.18%	0%	-10.31%
3	0%	0%	0%	0%	0%	0%
4	1.05%	2.68%	2.34%	3.81%	0%	1.98%
5	1.03%	3.53%	5.36%	4.11%	0%	2.81%
6	1.19%	4.08%	6.12%	6.44%	0%	3.57%
8	1.20%	4.35%	6.96%	7.45%	0%	3.99%



(a) Trade-off for \$20,000 budget **(b)** Trade-off for \$40,000 budget

(c) Trade-off for \$60,000 budget **(d)** Trade-off for \$80,000 budget

Figure 2.5: Trade-off between relative reduction in expected damage and relative increment in runtime.

We also see from Tables 2.4a and 2.4b that the effect of increasing the number of allocation levels on expected damage varies as the budget changes. We see from Table 2.4a that the percentage reduction in the expected damage increases as the budget increases from $20,000 to $80,000. As the budget increases, higher levels of cost-share assistance are offered to the landowners that increase the probability of the landowners accepting the cost-share program and eventually reduce the expected damage. However, at a lower budget such as $20,000, the model using $|\mathcal{K}| = 2$ may not be able to allocate any cost-share assistance to the landowners and thus incur a larger expected damage, whereas with the same budget, a model with a larger number of allocation levels can provide relatively higher levels of cost-share assistance to the landowners, eventually resulting in a reduced expected damage. On the other hand, as the budget increases to a sufficiently large amount so that it is possible to allocate the maximum cost-share assistance to all landowners even with $|\mathcal{K}| = 2$, there is no improvement in damage reduction using larger numbers of allocation levels. That is why the percent improvement at the highest budget level, $100,000, is consistently 0% as the number of allocation levels increases from $|\mathcal{K}| = 2$.

2.4.5 Cost of Uniformity

We conducted experiments with the uniform allocation model described in Section 2.3.1 to compare its performance with the risk-based allocation strategy for the same parameters values—budget size, number of allocation levels, and number of landowners. Table 2.5 shows the expected damage for each parameter combination in each allocation strategy as well as the percent improvement in wildfire damage reduction from risk-based allocation over uniform allocation strategy. Cells marked with asterisk (*) represent the combinations that cannot reach optimality within 6 hours for which we take the best sub-optimal solution and the corresponding upper bound as expected damage found at termination. The cost-share assistance levels allocated to each landowner for each parameter combination in each allocation strategy are presented in Table A.9 in Appendix A.4.

Table 2.5 shows that the risk-based allocation strategy provides up to 37.3% improvement in damage reduction over the uniform allocation strategy for budgets from $20,000 to $80,000. The improved flexibility of risk-based allocation enables an agency to use its budget more

Table 2.5: Comparison of expected damages from uniform, hybrid, and risk-based allocation strategies, where B is cost-share budget, $|\mathcal{K}|$ is the number of allocation levels, and $|\mathcal{J}|$ is the number of landowners.

B	$\|\mathcal{K}\|$	$\|\mathcal{J}\|$	Expected Damage			Improvement in Damage Reduction		
			Uniform	Hybrid	Risk-based	Risk-based over Uniform	Risk-based over Hybrid	Hybrid over Uniform
\$20,000	6	4	142.6	131.8	131.2	8.7%	0.4%	8.2%
		6	140.4	125.2	124.2	13.1%	0.9%	12.1%
		8	138.4	121.4	119.5	15.8%	1.6%	14.0%
		10	136.3	120.0	117.9*	15.6%	1.8%	13.6%
\$40,000	6	4	103.1	88.4	86.5	19.1%	2.1%	16.6%
		6	100.8	90.7	84.4	19.4%	7.4%	11.1%
		8	98.7	83.3	81.0	21.8%	2.8%	18.5%
		10	96.5	80.7	78.7*	22.6%	2.5%	19.6%
\$60,000	6	4	68.4	67.0	55.1	24.1%	21.5%	2.1%
		6	66.3	56.2	53.6	23.7%	4.9%	17.9%
		8	64.6	54.6	50.2	28.6%	8.8%	18.3%
		10	62.7	54.6	49.2*	27.5%	10.9%	14.8%
\$80,000	6	4	42.8	42.8	32.5	31.7%	31.9%	0%
		6	41.3	41.3	30.8	34.1%	33.8%	0%
		8	40.1	36.4	29.2	37.3%	24.7%	10.2%
		10	38.8	32.0	28.4*	36.7%	12.7%	21.3%

efficiently by allocating higher level of cost-share assistance to the owners of high priority land parcels (e.g. hazardous land parcels, land parcels with high economic value) while allocating lower level of cost-share assistance to the owners of low priority land parcels. This in turn increases the high priority land parcels owner's probability of accepting the cost-share offer leading to a larger reduction in the overall wildfire damage. On the other hand, the rigidity of uniform allocation limits the agency's ability to efficiently allocate its cost-share budget. Therefore, while the risk-based allocation strategy might spur controversy among the targeted landowners, the risk-based allocation strategy allows the agency to develop more customized and tailored solutions for the given landscape and landowners, which could help the agency better accomplish its wildfire mitigation objectives.

While the uniform allocation strategy sacrifices solution quality, it does outperform the risk-based allocation in terms of runtime, especially for larger number of landowners as seen in Figure 2.6, and Tables A.3, and A.4 in Appendix A.2. Decision-makers can consider this trade-off between model runtime and solution quality when selecting their allocation strategy. However, since the cost-share budget most likely would be allocated before a wildfire season, runtimes are not critically important. Therefore, the longer runtimes (additional 1-2 hours) of the risk-based allocation strategy are negligible compared to the improvement in wildfire damage reduction.

2.4.6 Cost of Semi-Uniformity

We performed computational experiments for the hybrid allocation strategy described in Section 2.3.1 using the same parameter combinations as the risk-based and uniform allocation experiments and compared with the results from the risk-based and uniform allocation experiments in Table 2.5. We see from Table 2.5 that the risk-based allocation strategy provides up to 33.8% improvement in damage reduction over the hybrid allocation strategy, and hybrid allocation strategy provides up to 21.3% improvement in damage reduction over the uniform allocation strategy for budgets ranging from \$20,000 to \$80,000.

Though the hybrid allocation strategy might mitigate some of the potential controversy of the risk-based allocation, the requirement to offer equal levels of cost-share assistance to all selected landowners limits the agency's ability to efficiently allocate its cost-share

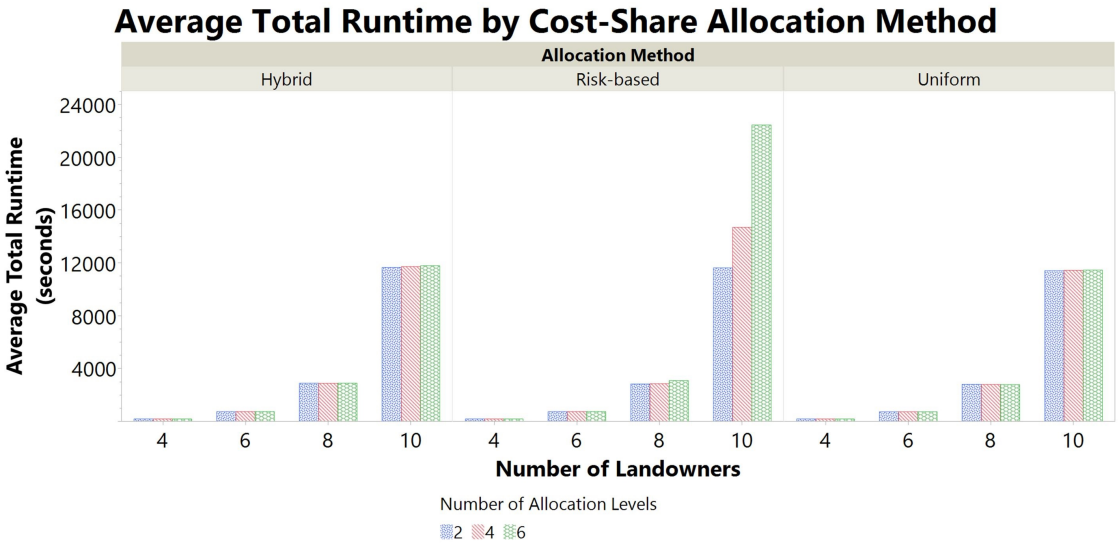


Figure 2.6: Impact of cost-share allocation strategy on runtime.

budget. This lack of versatility limits both the hybrid and uniform allocation strategies and hence the risk-based allocation strategy provides better solutions. That being said, the runtimes of hybrid allocation are consistently lower than those of risk-based allocation for larger number of landowners, as demonstrated in Figure 2.6 and further in Tables A.3 and A.5 in Appendix A.2. However, as noted earlier runtimes are not critically important for this cost-share allocation problem, hence the improvement in wildfire damage reduction provided by the risk-based allocation strategy is beneficial for the land managers, despite the longer (additional 1-2 hours) runtime.

The additional flexibility in the hybrid allocation strategy provides better quality solutions than uniform allocation with runtimes that are only few minutes longer than the uniform allocation runtimes, especially for $|\mathcal{J}| = 8$, and $|\mathcal{J}| = 10$, as seen from Figure 2.6, and Tables A.4, and A.5 in Appendix A.2.

2.5 Conclusions

We studied the problem of reducing the damage from wildfire by offering cost-share assistance to NIPF landowners through a cost-share program to encourage them to implement treatments reducing the accumulation of hazardous fuels on their lands. The decision-maker offers their limited cost-share budget without knowing whether the landowners will accept or reject their individual offers. We developed a stochastic simulation-optimization approach with decision-dependent uncertainty to model this problem. Our model prescribes the optimal allocation of cost-share assistance to the landowners to minimize the expected damage from wildfire. Our stochastic programming framework provides the cost-share allocation decision and is informed by a simulation approach that models wildfire spread through the landscape and computes associated damage. We also incorporated a predictive modeling technique that uses landowner data to estimate the probability that a landowner will accept a given cost-share assistance offer.

Computational experiments demonstrate the performance of our methodology. We see that the runtime increases exponentially as the number of landowners in the problem increases, because the increment in the number of landowners increases the number of integer

variables and constraints as well as scenarios in the stochastic mixed-integer programming model. We see that the runtime increases with the cost-share budget levels and number of allocation levels for larger number of landowners, $|\mathcal{J}| = 8$ and $|\mathcal{J}| = 10$. The runtime suggests that our stochastic risk-based cost-share assistance allocation model can be used to solve real-life problems over large landscapes in a reasonable amount of time. For instance, the computational experiments demonstrate that the average runtime of the stochastic risk-based allocation model with 10 landowners and four cost-share allocation levels is approximately 4 hours, and the average runtime with eight landowners and six cost-share allocation levels is less than 1 hour. Therefore, our model can be used as the basis for a decision-making tool for allocating and coordinating fuel reduction treatments over large landscapes.

We see that the runtime varies with different cost-share allocation strategies. In the uniform and hybrid cost-share allocation models, the runtimes remain more consistent across different cost-share budgets and allocation levels for a given number of landowners than the runtimes of the risk-based allocation model. This is because the computational complexity of the risk-based model is higher than the uniform and hybrid models. The decision-making in risk-based allocation requires the exploration of more possible combinations of cost-share allocation than the uniform and hybrid models, as those models are more rigid in making allocation decisions. Intuitively, the numerical results also show that the expected damage decreases as the budget increases.

An important finding from our experimental results is that the expected damage varies as we move from risk-based cost-share resource allocation to uniform and hybrid cost-share allocation strategies. We see that the expected damage is lower in the risk-based resource allocation than the expected damage resulting from uniform and hybrid allocation with the same cost-share budget size and the same number of cost-share allocation levels. The risk-based cost-share allocation strategy provides up to 37.3% and 33.8% more reduction in damage than the uniform allocation and hybrid allocation strategies, respectively. This is because, unlike the uniform and hybrid allocation strategies, in risk-based allocation the decision-maker takes into account fire hazard level and economic value associated with each property parcel in the landscape and allocates cost-share resources to minimize the overall risk of wildfire damage. In contrast, the uniform cost-share allocation does not ensure that

assistance will be allocated to most hazardous land parcels nor that owners of these parcels will be willing to implement fuel reduction treatments at offered assistance levels. However, risk-based allocation allows for customizing monetary offer levels based on wildfire risk, offering higher assistance to the owners of more hazardous or higher economic value land parcels, and thus increasing the probability that owners of these parcels will accept the cost-share offer and implement fuel reduction treatments.

A practical implication of the risk-based allocation strategy is that land managers can identify areas that pose greatest wildfire hazard and those are most crucial in terms of wildfire spread, and allocate cost-share assistance accordingly. Land managers can thus utilize available cost-share dollars more efficiently to coordinate implementation of fuel reduction treatments by incentivizing their placement in strategic locations in the landscape that minimizes the overall wildfire damage.

Results demonstrate that the stochastic risk-based resource allocation model is sensitive to the number of cost-share resource allocation levels used in the model. A measure of the relative change demonstrates that the expected damage decreases as the number of allocation levels used in the model increases. With more cost-share allocation levels, the cost-share offer amounts become more refined, allowing for a more efficient use of resources, resulting in a decrease in the expected damage.

Increasing the number of allocation levels increases the solution quality but can also increase the runtime. Moreover, the rate of damage reduction diminishes as the number of allocation levels becomes increasingly large. Thus, a moderately large number of allocation levels (i.e., $|\mathcal{K}| = 6$ for the test landscapes in this study) can provide sufficiently high solution quality and a reasonable runtime.

2.5.1 Future Work

In this study, we modeled the spread of wildfire considering random points of origin for the wildfire in the landscape. However, if we want to prescribe a cost-share budget allocation decision to minimize damage against the worst-case wildfire scenario, a more conservative approach might be to select the fire ignition points that maximize the wildfire damage (e.g., [118]). In this regard, we can formulate the problem as a bi-level min-max model where

the inner problem is the damage maximization problem by optimally selecting the ignition points. The outer level minimizes the expected maximum damage by optimally offering cost-share assistance to the landowners.

Considering the economies of scale inherent in fuel reduction treatment might also be beneficial. For example, if we can incentivize the implementation of fuel reduction treatment on adjacent cells, this may lead to cost savings from allowing landowners to use the same equipment, crews etc. This aspect is important because many small landowners face difficulties with implementing prescribed fire on their land due to costs, lack of equipment, and need to hire qualified labor. Pooling resources together might not only help landowners to overcome these challenges but also increase wildfire mitigation benefits.

Another possible extension of this research could be modeling the cost-share program for fuel reduction treatments from the game-theoretic perspective to address the strategic interactions between private landowners and government agencies (e.g., [25]).

In this research, we assume that fire travel time is increased in land parcels subject to fuel reduction treatment and that this can prevent fire from spreading to adjacent parcels. However, in practice, fire propagation will not be deterministic as it will depend on both the fuel complex and uncertain fire weather conditions (i.e., temperature, humidity, wind). Accordingly, a relaxation to the assumption of deterministic fire spread can be made by considering a probability of fire spread through land parcels, where the probability will be smaller for treated cells.

Another limiting assumption of this research is that we model the fuel reduction treatment as a single period problem. But, in reality, fuel can accumulate in the treated cells over time, diminishing the effectiveness of fuel treatments. This requires to account the fuel accumulation over time to reduce damage from wildfires over a planning horizon. The multi-period setting of our problem can be modeled as a multi-stage stochastic programming/stochastic dynamic programming with endogenous uncertainty framework. This multi-stage model would keep track of the fuel re-growth in the landscape and allows modification of the cost-share allocation decisions at each investment period (e.g., a year) over the planning horizon. This way the multi-stage stochastic programming model provides decision support to efficiently allocate limited cost-share resources in incentivizing

fuel reduction treatments over a planning horizon and maintain a fire-resistant landscape. However, solution of this type of multi-stage model would warrant the development of a new solution technique as this model has not been solved in the existing literature to date. Unlike the existing multi-stage stochastic programming with endogenous uncertainty (e.g., [45]) where decisions uncover more accurate information to resolve uncertainty, in our multi-stage endogenous uncertainty framework, the cost-share allocation decisions affect the probability distributions governing uncertainty and thus alter the probability of the scenario tree. Additionally, computational challenge would be posed because of the existence of large number of binary decision variables for each node of the decision tree in each period.

We assume that landowners accepting a cost-share offer implement a fuel reduction treatment on all of their land parcels. However, landowners can own heterogeneous land parcels with different vegetation types. An extension to the existing model can be made by assuming that the agencies consider a number of clusters of land parcels in the landscape delineated based on the vegetation types, ages etc., where each landowner can own multiple clusters. In this setting, in addition to deciding which landowners to offer cost-share assistance to, the agency must also specify which land clusters the assistance applies to. This requires the introduction of a new set of decision variables for incorporating clusters into the existing stochastic programming model. In addition to the landowners' accept/reject decisions, the scenarios represent the combination of which clusters are treated or not. In this framework, both the cost-share allocation and cluster selection decisions affect the scenario probabilities. For example, if a cluster is treated in a scenario but not chosen by the agency to offer cost-share assistance, the probability of that scenario becomes zero, meaning that scenario does not exist. This framework is able to represent a more practical fuel treatment scheme. However, this relaxed model will pose additional computational challenges due to the increased number of binary decision variables as well as scenarios representing which clusters are treated or not. Therefore, implementation of a sample average approximation technique would provide computational benefit. We can also solve the model using stochastic optimization algorithms, such as L-shaped decomposition to make the solution procedure computationally faster.

An additional possible extension of this research is to relax the assumption that landowners decide independently about the cost-share offer. Incorporating possible correlation among the landowners' decisions in the modeling framework will provide decision support for a more general fuel reduction treatment incentivizing problem. However, this will result in a non-convex mixed-integer nonlinear stochastic programming (MINLP) model that is computationally very challenging. One way of solving the non-convex stochastic MINLP is to develop an approximate mixed-integer linear programming model by factorizing the original model. For further details of the possible solution methods, we refer readers to Burer and Letchford [24].

Chapter 3

A Stochastic Programming Model with Endogenous and Exogenous Uncertainty for Reliable Network Design Under Random Disruption

This chapter and Appendices B, C, D, and E are based on the paper published by Tanveer Hossain Bhuiyan, Hugh R. Medal, and Sarah Harun:

> Bhuiyan, T. H., Medal, H. R., and Harun, S. (2020). A stochastic programming model with endogenous and exogenous uncertainty for reliable network design under random disruption. *European Journal of Operational Research*, 285(2): 670–694.

Authors Bhuiyan and Medal posed the research problem. Author Bhuiyan developed the mathematical models and the algorithmic framework. Authors Bhuiyan and Medal developed the reformulation and the proof of Proposition 1. Authors Bhuiyan and Harun implemented the algorithms and conducted the computational experiments. Author Bhuiyan wrote the manuscript and created the figures and tables. Authors Medal edited the manuscript. During the three revisions in the peer-review process, author Bhuiyan addressed the reviewers'

comments, conducted the computational experiments, and revised the manuscript. Author Medal reviewed and edited the revised manuscript and the responses to reviewers' comments.

3.1 Introduction

This chapter studies an integrated network design and infrastructure protection (NDIP) problem where the facilities are exposed to random disruptions. The goal is to optimally allocate protection resources among the facilities and build links in a network that minimize the expected post-disruption transportation cost in satisfying the demands of customer locations. This is the first study to model a NDIP problem under the following assumptions: (1) protection is imperfect, meaning that despite allocating protection resources, a facility's capacity can degrade when exposed to disruption; (2) the protection is multi-level in the sense that a facility is more fortified as more protection resources are allocated to it; (3) a facility has multiple post-disruption capacity states (PDCSs), which are probabilistic function of the protection investment amount and the random disruption exposure level. To incorporate these assumptions, we model the NDIP problem as a stochastic program with both endogenous and exogenous uncertainty. The uncertainty that is affected by the decision maker's decision is known as endogenous (decision-dependent) uncertainty, whereas the exogenous uncertainty is not influenced by the decisions. The main goal of this research is to explore the benefits of incorporating the imperfect, multi-level protection and multi-state capacity of the facilities in a combined NDIP problem. Also, our interest is to develop solution methodology and gain insights in solving an optimization problem with both exogenous and endogenous uncertainty.

3.1.1 Motivation

In a supply chain or distribution network, facilities contribute to its smooth operation by producing the required amount of commodities to satisfy customer demands. An efficient distribution network transports the commodities to customer locations at a minimum transportation cost. With limited resources, decision-makers always seek to design a transportation network that maximizes efficiency. However, smooth operation of the network

is often disrupted when facilities are exposed to unavoidable disruptions, such as natural disasters, or equipment failures. These disruptions can cause inefficient operation or even unexpected shutdowns of the system and can eventually lead to significant financial loss. Examples of devastating disruptions include the supply chain disruption of Ford F-150 trucks [61], Hurricane Harvey in August 2017 in the U.S. [113], Hurricane Katrina in August 2005 in the U.S. [12], and the Tōhoku tsunami in March 2011 in Japan [30].

Therefore, company decision-makers seek to find the best allocation of their limited protection resources to the infrastructures to minimize the post-disruption transportation cost (PDTC) as well as provide maximum post-disruption utility to the customers. Decision-makers can identify the critical infrastructures and implement measures such as structural retrofication, improved monitoring systems, and fire prevention systems to fortify facilities. Although the existing research studies inform decision-makers regarding efficient NDIP, most of the studies provide solutions based on assumptions that fail to address real-life problems. Such assumptions include (1) a facility is either completely protected if fortified (perfect protection) or left completely unprotected and fails completely once exposed to a disruption (perfect disruption), (2) after a disruption, facilities can be either operable with their full capacity or non-operable with zero post-disruption capacity (binary-state capacity). But, in reality, protection does not provide perfect defense against disruptions, and the facilities can be partially operable after disruptions depending on the amount of protection and the intensity of the disruptions. Thus, existing models based on these assumptions are likely to provide inferior protection investment decisions. In fact, Medal et al. [94] demonstrated that assuming perfect and binary protection yields solutions that provide much lower utility to the customers compared to the solutions that accounts for imperfect and multi-level protection. However, it is unclear whether this finding holds for a model that includes both NDIP decisions. Therefore, there is a need to relax these assumptions in a combined NDIP problem to provide robust solutions for real-life problems.

3.1.2 Related Literature

We provide a review of the literature related to our study from the perspective of infrastructure (facility) protection, combined NDIP, and stochastic programming with endogenous uncertainty in the following sub-sections.

Facility Protection Models

In the facility protection literature, researchers have studied the problem of fortifying a set of facilities by allocating protection resources or building backup facilities. Snyder et al. [133] presented a wide range of models for facility location-protection subject to disruptions, either caused by a natural disaster or intentional attacks (interdiction). Church and Scaparra [29] studied a facility fortification problem, known as the $r-$interdiction median problem with fortification (RIMF), that seeks to optimally allocate limited protection resources to a subset of critical facilities to minimize the post-interdiction demand-weighted distance. Later, other studies extended the RIMF literature in different directions such as minimizing worst-case losses [127, 128], using a budget constraint on the protection resources and possible capacity expansion of the facilities [7], developing a tri-level model with capacitated facilities [126, 81], and addressing uncertainty in the number of disrupted facilities [80, 79].

Some research has studied the problem of locating and protecting facilities simultaneously. O'Hanley et al. [104, 105] studied the problem of locating and protecting nature reserves against random and intentional disruptions caused by human intrusions or natural disasters. Lim et al. [82] modeled a problem of locating fortified backup facilities to improve the reliability of a facility network. This work was extended by Li et al. [78] where a subset of selected facilities are retrofitted within a limited budget that makes the facilities perfectly reliable. Aksen et al. [6] modeled a facility location-fortification problem as a bi-level integer programming model where the facilities are exposed to interdiction. Medal et al. [93] extended the work of Aksen et al. [6] by introducing a tri-level formulation of the location-fortification problem. Qin et al. [112] proposed a two-stage stochastic programming model to minimize the cost of fortifying an existing network of capacitated facilities subject

to disruptions and the cost of post-disruption emergency reassignment of the customers to the facilities.

Besides the generic facility protection models, some researchers have studied the problem of fortifying facilities in specific network applications, including fortifying hub facilities in a hub network against natural disruptions [11] as well as interdictions [44, 116], protecting terminals in a rail intermodal terminal network [125].

The above mentioned studies assumed perfect impact of protection and perfect impact of a disruption on a facility. However, some studies have relaxed the assumption of perfect protection and perfect disruptions. Losada et al. [84] extended Church and Scaparra [29] and the RIMF literature by incorporating the imperfect effect of disruption, where the effect of disruption on a facility depends on disruption intensity. In another study, Losada et al. [85] modeled a RIMF problem with recovery time incorporating imperfect protection, where protection reduces the recovery time required to restore a facility to its operational status. However, the authors assumed that a disruption always makes a facility completely inoperable. Eiselt and Marianov [35] studied a cell phone tower fortification problem against natural disaster with an assumption that a fortified facility may fail with a given probability. Zhu et al. [147] incorporated imperfect protection of the critical facilities against multiple interdictors. Aksen et al. [5] modeled imperfect interdiction assuming that facilities remain partially operable after interdiction depending on the intensity of the interdiction. Jabbarzadeh et al. [62] proposed a hybrid robust-stochastic programming facility fortification model against natural disruptions considering a reliable and an unreliable set of facilities, where the reliable facilities never fail but the unreliable facilities can fail either completely or partially depending on the fortification level. The authors modeled the disruption intensity to an unreliable facility as a function of protection resources. In their model, the authors considered fixed post-disruption capacities corresponding to each fortification levels. Unlike Jabbarzadeh et al. [62], Medal et al. [94] proposed a facility fortification model against natural disruptions, where no facility is perfectly reliable and the post-disruption capacity of a facility is a function of the protection resources allocated and the disruption intensity.

Integrated Network Design and Facility Protection Models

The facility protection problem is often modeled as a problem separate from the network design problem. But it is more cost efficient to model facility fortification problem simultaneously with network design, as the combined model could be able to make better solutions by utilizing the limited budget. Some research studies combined facility protection with network design and thus provided a complete decision support model to the system managers in making their transportation network both reliable and efficient when subject to disruptions. We classify the literature to be in the class of integrated network design and facility protection that simultaneously considers the decisions of facility protection and physical network design by constructing links, or distribution centers, or transshipment nodes. Snyder et al. [133] introduced the notion of reliable network design along with facility location and protection against disruptions by opening transshipment nodes through which the products are shipped from the facilities to the customer locations. Peng et al. [111] proposed a robust optimization approach with stochastic $p-$ robustness criteria to design a reliable logistic network, where backup transshipment facilities are opened to supply customer locations when the corresponding primary facility fails under disruption. Another robust optimization model with $p-$robustness criteria for forward-reverse logistic network design problem was developed by Hatefi and Jolai [52] accounting for demand uncertainty and facility disruptions. Shishebori et al. [132] formulated a mixed-integer programming model for a reliable facility location and network design problem that seeks to minimize the cost of reassignments of demands to backup facilities when primary facility fails under disruption. However, this reassignments requires more link construction and increases the transportation cost. Later, a robust optimization model was presented by Shishebori and Babadi [130] that seeks to optimally design capacitated medical services network along with opening capacitated medical service centers subject to disruptions. The authors constructed scenarios to represent disruptions with a probability, where each scenario involves the set of unavailable facilities and links. Shishebori et al. [131] presented a two-stage stochastic programming model for a logistics network design problem with capacitated facilities and links subject to disruptions. Fang and Zio [39] proposed a robust optimization framework for

increasing the resiliency of a power transmission network against natural hazards by opening backup generation unit on a transmission node and allocating protection resources to the transmission lines.

All of the above-mentioned models assumed perfect impact of protection and disruption to the facilities or the network links, which is not always possible in reality. Mohammadi et al. [102] studied a hub location network design problem considering partial disruption of the hubs and links, where the demands are re-allocated to surviving hubs after disruption. Azad et al. [10] relaxed the assumption of perfect protection for unreliable facilities to some extent in a supply chain network design problem. The authors assumed the following : (1) two types of facilities exist, reliable and unreliable, where the unreliable facilities are only subject to disruptions; (2) the lost capacity of an unreliable facility due to a disruption depends on the amount of initial investment to that facility; (3) the lost capacity of an unreliable facility is supplied from a reliable facility. Despite assuming multiple levels of protection to the unreliable facilities, the authors modeled perfect protection for the reliable facilities. Also, this research fixed the amount of capacity lost for a given initial investment which cannot be known *a priori* in real-life. Later, Azad et al. [9] relaxed the assumption of a fixed amount of capacity loss for an investment level by taking the capacity loss from a normal distribution with a known mean and variance. The authors used a conditional-value-at-risk (CVaR) measure to formulate the amount of capacity loss of the disrupted unreliable facilities. However, similar to Azad et al. [10], Azad et al. [9] also assumed that the reliable facilities cannot be disrupted when exposed to a disruption. Also, they did not consider the effect of disruption intensity on the post-disruption capacity of the unreliable facilities. With these restricted assumptions, these models may not provide robust solutions to the realistic NDIP problems and thus are likely to incur a substantial amount of financial penalty.

Stochastic Programming with Endogenous Uncertainty

In the stochastic programming literature, decisions made by the decision-makers typically do not influence the underlying uncertainty in the stochastic process. However, in some application areas, decisions can affect the uncertainty in the stochastic process. Goel and Grossmann [45] classified the latter uncertainty as endogenous, whereas the former

as exogenous. Goel and Grossmann [45] further classified the stochastic programming with endogenous uncertainty into Type I or Type II, depending on the effect of decisions on uncertainty. In Type I endogenous uncertainty, decisions alter the probability distributions of the uncertain parameters. For example, the survival probability of a component increases as the decision-maker invests more protection resources on that. In Type II endogenous uncertainty (a.k.a., *exogenous uncertainty with endogenous observations*), decisions affect the information that the decision-maker has about the uncertainty (e.g., the timing of when the uncertainty is resolved). For example, the time when the decision-maker observes the true production rate of an oil field depends on when the investment is made to begin production.

In this research, we study the endogenous uncertainty of Type I with exogenous uncertainty. While there exist a good number of research studies on Type II endogenous uncertainty [8, 88], only a few studies modeled stochastic programming with Type I uncertainty in the application areas including infrastructure protection [110, 34, 94], natural gas infrastructure [54], disaster preparedness [38], wildfire hazard mitigation [71, 17], maintenance and production planning [36], and network design, server selection, and facility location [4]. To the best of our knowledge, Peeta et al. [110] is the first study to introduce the endogenous uncertainty in infrastructure protection, where the first-stage investment decisions reduce the likelihood of link failure in a highway network due to disruption. Du and Peeta [34] further extended Peeta et al. [110] to allow for partial investment. Medal et al. [94] modeled a facility fortification problem as a two-stage stochastic programming with endogenous uncertainty and solved the model using an exact algorithm. However, these studies on infrastructure protection modeled their problem using endogenous uncertainty only, thus, the first-stage decisions directly affecting the recourse decisions are absent. However, from the view of stochastic programming literature, a problem with both endogenous and exogenous uncertainty is an open research problem, where the introduction of exogenous uncertainty would pose additional challenges as well as provide insights to the decision-makers.

3.1.3 Contributions

In summary of the existing literature, no research has modeled an integrated NDIP problem under random disruptions with imperfect effect of protection and disruption, multiple levels of protection and multiple post-disruption capacity of the facilities. To fill the gap in the literature, we studied an integrated NDIP problem by relaxing the assumptions of perfect protection and disruption, and binary state capacity of the facilities when subject to random disruptions. We modeled the problem as a two-stage stochastic program, where both exogenous and endogenous uncertainty coexist to analyze whether previous findings of facility fortification models with only endogenous uncertainty (e.g., [94]) hold true.

Though we model the imperfect protection in a similar way to Medal et al. [94], our research makes significant contribution to the literature compared to Medal et al. [94] from several key aspects as follows: (1) we study a new problem that seeks to decide how much protection to invest to the facilities in conjunction with building links in a transportation network; (2) due to the presence of network design decision in our problem, it is difficult to take advantage of the special structure as in Medal et al. [94] (in which the second-stage cost in each scenario can be computed *a priori*), which requires a different modeling approach for our integrated NDIP problem; (3) despite a similar approach, L-shaped decomposition algorithm, the different problem structure necessitates modification of the algorithm; (4) we implement a data-driven approach for estimating the effect of protection resources and disruption intensities to the facilities; (5) we present new insights into the effects of imperfect, multi-level protection, and multiple post-disruption capacities in an integrated NDIP problem.

Our research extends the literature on network design and facility location-protection under random disruptions by introducing a new model that presents new insights addressing real-life issues. Specifically, in this research we have made the following contributions: (1) developed a new stochastic programming model with both endogenous and exogenous uncertainty that seeks to minimize a network's expected PDTC by optimally constructing network links and allocating protection resources to facilities subject to random disruptions, (2) implemented an accelerated L-shaped decomposition algorithm to solve the model, (3)

introduced predictive modeling techniques to estimate the likelihood of a facility's PDCS for a given protection investment and a disruption intensity level the facility is exposed to, (4) provided experimental results to demonstrate the effects of model parameters on the runtime of the solution methodology and on expected cost, the significance of modeling uncertainty in the PDCSs of the facilities, the sensitivity of the model to changes in the number of protection investment levels and post-disruption capacity levels used in the model, and the sensitivity of the stochastic programming model to the estimation error in the likelihood of a facility's PDCS from different prediction models.

3.2 Problem Description

In our problem, a decision-maker seeks to optimally build links in a network of spatially located facilities and demand locations and allocate protection resources to the facilities within a limited budget to minimize the PDTC. We assume that no links exist *a priori* among the facilities and demand locations in the network, where the decision-maker has to open links incurring a cost to transport commodities from one location to another. Real-life applications of this type of network includes oil/gas transportation from the oil field to different cities through a pipeline network and power transmission networks where the transmission lines are built to transfer the electrical energy from power plants to substations. In these applications, and as assumed in this research, the facilities can be placed only in particular geographic locations.

We also assume that both the facilities and the network links have a limited capacity. This assumption complies with the real-life examples mentioned earlier, where there is a maximum limit on the cubic-feet per hour of oil/gas that can be transported through a pipeline. The facility capacity is degraded through exposure to disruptions such as natural disasters. We assume that the links are completely reliable and are unaffected by disruptions. The decision-maker can allocate limited protection resources such as robust building materials, fire sensors, and pollution control systems to the facilities to protect against disruptions. The decision-maker has a limited budget that can be spent on link construction and protection investment. In this research, we assume imperfect and multi-level protection. The effect

of disruption on a facility is imperfect, and the post-disruption capacity decreases as the severity of the disruption increases. The post-disruption capacity of a facility is uncertain and depends probabilistically on the protection amount invested to that facility and the disruption intensity level the facility is exposed to. The probability that a facility has a specific post-disruption capacity for a given protection and a disruption intensity level is unknown to the decision-maker, but the decision-maker can estimate the probability from historical data as well as the probability of disruptions that could occur at a geographic location.

We assume multiple post-disruption capacities of the facilities where the probability that a facility has a higher post-disruption capacity increases as the protection investment increases and disruption intensity decreases. To represent uncertainty in the post-disruption capacity, we construct scenarios where each scenario contains a disruption event and a particular realization of the post-disruption capacities of the facilities. The total number of scenarios is equal to the product of the number of disruption events and the number of possible combinations of the facilities' post-disruption capacities. A scenario-based stochastic programming approach is suitable to model the failure of capacitated elements (Medal et al. 94, Peng et al. 111). After a disruption is realized, the facilities supply the demand locations with their available post-disruption capacities. We assume a penalty cost for each unit of unmet demand of the customer locations. We refer to this problem as probabilistic network design and infrastructure protection problem (PNDIPP).

As the post-disruption capacities of the facilities are a probabilistic function of the protection decisions, those decisions affect the probability of scenarios, making the uncertainty decision-dependent or endogenous. For example, if the protection to a facility is high, then the scenarios with lower post-disruption capacity of that facility will have small probability with a reduced contribution to the overall objective value. On the other hand, the link opening decisions directly affect the decisions of the amount of flow through the links in each scenario, which introduces the exogenous uncertainty to the problem. We present a numerical example to demonstrate how the facility protection decisions affect the scenario probabilities.

Consider a small network with two facilities having two possible post-disruption capacities. For simplicity, we consider only one disruption event and both facilities are exposed to zero disruption intensity level. With two possible post-disruption capacities of each facility, this example problem has four possible combinations of the facilities' post-disruption capacities, resulting in four scenarios. We assume that the decision-maker can make two possible protection investments: no protection (0 unit), and 1 unit of protection. Table 3.1 shows the estimated probabilities of a facility's post-disruption capacities for given investments and disruption intensity level.

We assume that 1 unit of protection is invested to facility 1 and no protection is invested to facility 2. The probabilities of the four scenarios based on this investment decision are presented in Table 3.2. We see from Table 3.2 that in scenario 1 both facilities have zero (0) post-disruption capacity. From Table 3.1, the probabilities that a facility has zero (0) capacity for given 1 unit of protection and no protection are 0.15 and 0.90, respectively. Therefore, the probability of scenario 1 is 0.135 $(= 0.15 \times 0.90)$, as shown in Table 3.2 along with the other scenario probabilities.

3.3 Mathematical Formulation

We formulate the PNDIPP as a two-stage stochastic programming model with both endogenous and exogenous uncertainty. Tables 3.3, 3.4, and 3.5 list the necessary sets, parameters, and variables, respectively, that support the mathematical formulations.

The first-stage model provides the optimal link opening decisions in the network and the decisions regarding the optimal amount of protection to be allocated to the facilities. Given the protection and link construction decisions from the first-stage, the second-stage model seeks to minimize the PDTC in each scenario by optimally sending flow through the network. In our stochastic programming model, the post-disruption capacity of a facility is uncertain and depends on the first-stage protection investment decision and the intensity level of the random disruption to which the facility is exposed. We consider a finite set of scenarios Ω, where each scenario $\omega \in \Omega$ represents a disruption event and a particular combination of the post-disruption capacities of the facilities. The random disruption events are represented by

Table 3.1: Estimated probability for a given protection investment level

Protection Level	**Disruption Intensity Level**	**Capacity**	**Estimated Probability**
0	0	0	0.90
0	0	1	0.10
1	0	0	0.15
1	0	1	0.85

Table 3.2: Probability of the scenarios

Scenarios	**Capacity**		**Probability**
	Facility 1	**Facility 2**	
1	0	0	$0.15 \times 0.90 = 0.135$
2	0	1	$0.15 \times 0.10 = 0.015$
3	1	0	$0.85 \times 0.90 = 0.765$
4	1	1	$0.85 \times 0.10 = 0.085$

Table 3.3: Sets

Sets	Description
$\mathcal{N}_D$	Set of demand points n
$\mathcal{N}_\mathcal{F}$	Set of facilities z
$\mathcal{N}$	Set of all nodes in the network, $\mathcal{N}_D \cup \mathcal{N}_\mathcal{F}$
Ω	Set of scenarios indexed by ω
$\mathcal{L}$	Set of links ℓ
$\mathcal{A}$	Set of arcs (i, j)
$\mathcal{A}_{p1}$	Set of arcs (i, j) whose head $j \in \mathcal{N}_\mathcal{F}$ and tail $i \in \mathcal{N}_D$
$\mathcal{A}_{p2}$	Set of arcs (i, j) whose head $j \in \mathcal{N}_D$ and tail $i \in \mathcal{N}_\mathcal{F}$
$\mathcal{A}_{p3}$	Set of arcs (i, j) whose head $j \in \mathcal{N}_D$ and tail $i \in \mathcal{N}_D$
$\mathcal{A}_{p4}$	Set of arcs (i, j) whose head $j \in \mathcal{N}_\mathcal{F}$ and tail $i \in \mathcal{N}_\mathcal{F}$
$\mathcal{RS}(n)$	Set of arcs (i, j) incoming to node n
$\mathcal{FS}(n)$	Set of arcs (i, j) outgoing from node n
$\mathcal{K}$	Set of protection levels indexed by k
$\mathcal{S}$	Set of capacity states indexed by s
$\mathcal{M}$	Set of disruption intensity levels indexed by m

Table 3.4: Parameters

Parameters	Description
d_n	Demand of demand point $n \in \mathcal{N}_D$
a_{zs}^{ω}	Post-disruption capacity of facility z corresponding to state s in scenario ω
C_ℓ	Cost of opening link $\ell \in \mathcal{L}$
C_k	Cost of allocating k level of protection resource to a facility
U_{ij}	Maximum amount of flow through arc $(i,j) \in \mathcal{A}$
t_{ij}	Transportation cost of per unit of flow through arc $(i,j) \in \mathcal{A}$
B	Total budget of the decision-maker
C_D	Capacity of the dummy facility D

Table 3.5: Variables

Variables	Description
x_ℓ	1 if link ℓ is opened, 0 otherwise
y_{zk}	1 if k level of protection resources are allocated to facility z, 0 otherwise
f_{ij}^{ω}	Amount of flow through arc (i,j) in scenario ω

a set, $\mathcal{H}$, where $h \in \mathcal{H}$ denotes a particular disruption event. Depending on the distance of a facility from the center of a disruption, each facility is exposed to a disruption intensity level. $\mathcal{M} := \{0, 1, \dots M\}$ is the set of disruption intensity levels indexed by m, where $m(z, \omega)$ represents that facility z is exposed to disruption intensity level m in scenario ω for disruption event h. The conditional probability function that a facility$z$ has a post-disruption capacity a_z^ω when exposed to disruption intensity level m in scenario ω for a given protection amount y_z is $f_z(a_z^\omega \mid y_z, m(z, \omega))$.

We assume that one facility's post-disruption capacity is independent of another facility. Defining the probability of occurrence of a disruption event in a scenario $\omega \in \Omega$ as $\mathbb{P}_h^\omega$, the probability of that scenario can be expressed as:

$$\mathbb{P}^\omega(\mathbf{y}) = \mathbb{P}_h^\omega \left(\prod_{z \in \mathcal{N}_\mathcal{F}} f_z(a_z^\omega \mid y_z, m(z, \omega)) \right) \tag{3.1}$$

We can discretize the entire protection amounts into $|\mathcal{K}|$ different levels and the post-disruption capacity into $|\mathcal{S}|$ different states as follows:

- $\mathcal{K} := \{0, 1, 2, \dots, K\}$ is the set of protection investment levels indexed by k.
- $\mathcal{S} := \{0, 1, 2, \dots, S\}$ is the set of PDCSs of the facilities indexed by s.

Using these sets of discrete protection levels and capacity states, the conditional probability function that a facility's PDCS is s when exposed to disruption intensity level of m in scenario ω, given k level of protection allocated to that facility can be expressed as $\sum_{k \in \mathcal{K}} \mathbb{P}^\omega{}_{zs(z,\omega)m(z,\omega)k} y_{zk}$. Here $\mathbb{P}^\omega{}_{zs(z,\omega)m(z,\omega)k}$ denotes the probability that facility z is in capacity state s when exposed to disruption intensity level m in scenario ω and k level of protection is invested to that facility. $s(z, \omega)$ denotes that the PDCS of facility z is s in scenario ω where the available post-disruption capacity of that facility is a_{zs}^ω. Therefore, the probability of scenario ω can be expressed as follows:

$\mathbb{P}^\omega(\mathbf{y}) = \mathbb{P}_h^\omega \left(\prod_{z \in \mathcal{N}_\mathcal{F}} \sum_{k \in \mathcal{K}} \mathbb{P}^\omega{}_{zs(z,\omega)m(z,\omega)k} y_{zk} \right)$. Decision-makers can estimate these probabilities $\mathbb{P}_{zs(z,\omega)m(z,\omega)k}$ from historical data using predictive modeling techniques (see Section 3.4.3). The number of scenarios in our stochastic programming model can be

computed as: $|\Omega| = |\mathcal{H}| \times |\mathcal{S}|^{|\mathcal{N}_\mathcal{F}|}$. The first-stage of our two-stage stochastic programming model (3.2) is presented below (MinExpCost):

$$\min \quad \sum_{\omega \in \Omega} \mathbb{P}^\omega(\mathbf{y}) \left(\sum_{(i,j) \in \mathcal{A}} t_{ij} f_{ij}^\omega \right) \tag{3.2a}$$

$$\text{s.t.} \quad \sum_{k \in \mathcal{K}} y_{zk} = 1 \qquad \forall z \in \mathcal{N}_\mathcal{F} \tag{3.2b}$$

$$\sum_{\ell \in \mathcal{L}} C_\ell x_\ell + \sum_{z \in \mathcal{N}_\mathcal{F}} \sum_{k \in \mathcal{K}} C_k y_{zk} \leq B \tag{3.2c}$$

$$y_{zk} \in \{0,1\} \qquad \forall z \in \mathcal{N}_\mathcal{F}, k \in \mathcal{K} \tag{3.2d}$$

$$x_\ell \in \{0,1\} \qquad \forall \ell \in \mathcal{L} \tag{3.2e}$$

The objective function (3.2a) seeks to minimize the expected PDTC over all the scenarios. Each constraint (3.2b) ensures that only one level of protection can be allocated to each facility. The total cost of link construction and protection investment to the facilities cannot exceed the decision maker's budget (3.2c). Constraints (3.2d) and (3.2e) represent the binary nature of the protection and link construction decisions, respectively.

We solve the second-stage model for each scenario ω using the link construction decisions obtained from the first-stage model. In our PNDIPP, the links are undirected. To model the direction of the flow of commodities, we consider two oppositely directed arcs (i,j) and (j,i) for each link ℓ. This ensures that the flow is possible in either direction between two nodes. Both arcs (i,j) and (j,i) have the same per unit transportation cost and capacity as the original link ℓ. We use the notation $\ell(i,j)$ to define that the arc (i,j) corresponds to link ℓ.

After realizing a disruption event in a scenario, the available post-disruption capacities of the facilities are used to satisfy the demand of the customer locations. We assume a dummy facility having very high capacity to satisfy any unmet demand with a very high transportation cost. The second-stage model corresponding to scenario ω is presented as follows (MinCost):

$$\min \quad \sum_{(i,j)\in\mathcal{A}} t_{ij} f_{ij}^{\omega} \tag{3.3a}$$

$$\text{s.t.} \quad f_{ij}^{\omega} \leq U_{ij}\hat{x}_{\ell(i,j)} \qquad \forall (i,j) \in \mathcal{A} \tag{3.3b}$$

$$\sum_{(i,j)\in\mathcal{RS}(n)} f_{ji}^{\omega} - \sum_{(i,j)\in\mathcal{FS}(n)} f_{ij}^{\omega} = d_n \qquad \forall n \in \mathcal{N}_{\mathcal{D}} \tag{3.3c}$$

$$\sum_{(i,j)\in\mathcal{FS}(z)} f_{ij}^{\omega} - \sum_{(i,j)\in\mathcal{RS}(z)} f_{ji}^{\omega} \leq a_{zs}^{\omega} \qquad \forall z \in \mathcal{N}_{\mathcal{F}} \tag{3.3d}$$

$$\sum_{(i,j)\in\mathcal{FS}(D)} f_{ij}^{\omega} \leq C_D \tag{3.3e}$$

$$f_{ij}^{\omega} \geq 0 \qquad \forall (i,j) \in \mathcal{A} \tag{3.3f}$$

The objective function (3.3a) minimizes the total PDTC in scenario ω. Each of the constraints (3.3b) ensures that no flow of commodity is possible through arc (i,j) if the corresponding link ℓ is not opened. Here, $\hat{x}_{\ell(i,j)}$ is a parameter of the second-stage model (3.3) that represents the value of the link construction decision variable x_ℓ computed by the first-stage model (3.2). Constraints (3.3c) and (3.3d) are the flow conservation constraints for the demand locations and the facilities, respectively. Constraint (3.3e) ensures that the amount of flow out of the dummy facility cannot exceed the capacity of that facility.

3.4 Solution Approach

This section details the solution methodology for solving our stochastic programming model. In this section, we introduce a reformulation technique that linearizes the nonlinear and non-convex stochastic programming model into a mixed-integer linear stochastic program. We implement an accelerated L-shaped decomposition algorithm to solve the resulting mixed-integer two-stage stochastic programming model. We also implement predictive modeling techniques to estimate the probabilities of the facilities' PDCSs for different protection and disruption intensity levels. These methodologies are described in the following sub-sections.

3.4.1 Reformulation

We can write the first-stage objective function of our nonlinear stochastic programming model (3.2) as:

$$\sum_{\omega\in\Omega}\left[\mathbb{P}^{\omega}_{h}\left(\prod_{z\in\mathcal{N}_{\mathcal{F}}}\sum_{k\in\mathcal{K}}\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)k}y_{zk}\right)\left(\sum_{(i,j)\in\mathcal{A}}t_{ij}f^{\omega}_{ij}\right)\right] \tag{3.4}$$

Equation (3.4) computes the expected PDTC over all scenarios. We see that Eq. (3.4) contains the product of the variables y_{zk} over all the facilities and also the product of the variables y_{zk} and f^{ω}_{ij}. This makes the model nonlinear and non-convex. To implement the existing decomposition algorithms to solve this stochastic programming model, we need to linearize this nonlinear model. Therefore, we introduce a reformulation technique to linearize this nonlinear objective function. This reformulation technique is similar to the one proposed in previous work [94, 106] and is known as a probability-chain reformulation. This reformulation technique computes the probability-weighted PDTC for each scenario ω using a set of recursive equations and avoids the requirement of the product term over the facilities in computing the probability of a scenario.

Assuming that the facilities fail independently from each other, we can substitute the product term $\prod_{z\in\mathcal{N}_{\mathcal{F}}}\sum_{k\in\mathcal{K}}\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)k}y_{zk}\left(\sum_{(i,j)\in\mathcal{A}}t_{ij}f^{\omega}_{ij}\right)$ with recursive expressions that use bookkeeping variables to store the partial probability-weighted post-disruption transportation cost (PPWPDTC) in scenario ω. Assume q^{ω}_{rk} is a variable that stores the PPWPDTC upto facility $r, i.e., \sum_{k\in\mathcal{K}}q^{\omega}_{rk} = \prod_{z=1}^{r}\sum_{k\in\mathcal{K}}\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)k}y_{zk}\left(\sum_{(i,j)\in\mathcal{A}}t_{ij}f^{\omega}_{ij}\right)$. The PPWPDTC is the product of the PDTC for scenario ω and the probabilities that the facilities $1, 2, ..., r$ are in their corresponding PDCSs in scenario ω. We introduce the following recursive equations to compute the value of q^{ω}_{rk} for each scenario ω.

$$q^{\omega}_{1k} = {\mathbb{P}^{\omega}}_{1s(1,\omega)m(1,\omega)k} y_{1k} \left(\sum_{(i,j)\in\mathcal{A}} t_{ij} f^{\omega}_{ij} \right) \quad \forall k \in \mathcal{K}, \omega \in \Omega \tag{3.5}$$

$$\sum_{k\in\mathcal{K}} q^{\omega}_{(r-1)k} = \sum_{k\in\mathcal{K}} \frac{1}{\max\left\{ {\mathbb{P}^{\omega}}_{rs(r,\omega)m(r,\omega)k}, \epsilon \right\}} q^{\omega}_{rk} \quad \forall r = 2, \ldots, |\mathcal{N}_{\mathcal{F}}|, \omega \in \Omega \tag{3.6}$$

$$q^{\omega}_{rk} \leq \left(\sum_{(i,j)\in\mathcal{A}} t_{ij} f^{\omega}_{ij} \right) y_{rk} \quad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, \omega \in \Omega \tag{3.7}$$

$$q^{\omega}_{rk} \geq 0 \quad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, \omega \in \Omega \tag{3.8}$$

In each scenario, the Eq. (3.5) computes the value of the q^{ω}_{rk} variable for the first facility. The variable q^{ω}_{1k} holds the product of the PDTC for scenario ω and the probability that the first facility is in its corresponding PDCS in scenario ω. The value of q^{ω}_{rk} variables for the facilities ranging from the second to the last facility are computed by the Eqs. (3.6). For the second facility in scenario ω, the variable q^{ω}_{2k} holds the product of the value of q^{ω}_{1k} variable for the first facility computed in Eq. (3.5) and the probability that the second facility in its corresponding PDCS. In this way, the value of q^{ω}_{rk} variable is computed recursively for each facility until it reaches the last facility where $\sum_{k\in\mathcal{K}} q^{\omega}_{|\mathcal{N}_{\mathcal{F}}|k} = \prod_{z\in\mathcal{N}_{\mathcal{F}}} \sum_{k\in\mathcal{K}} {\mathbb{P}^{\omega}}_{zs(z,\omega)m(z,\omega)k} y_{zk} \left(\sum_{(i,j)\in\mathcal{A}} t_{ij} f^{\omega}_{ij} \right)$. In some instances the probability of a facility's PDCS might be very close to zero or even zero. To avoid division by zero, we take the maximum value between the probability ${\mathbb{P}^{\omega}}_{zs(z,\omega)m(z,\omega)k}$ and a very small positive number ϵ in the denominator on the right side of Eq. (3.6). Equation (3.7) ensures that the value of q^{ω}_{rk} for a facility z for a given protection level k can be positive only if k level of protection is allocated to facility z.

The Proposition 1 provides justification that for a given protection decision $\hat{\mathbf{y}}$, the probability-chain reformulation (3.5)–(3.8) computes the probability-weighted PDTC for each scenario ω.

Proposition 1. *Assume all the probabilities* ${\mathbb{P}^{\omega}}_{zs(z,\omega)m(z,\omega)k}$ *are greater than* ϵ*, a small positive number. If the facilities fail independently of each other, then for a given solution* $\hat{\boldsymbol{y}}$*,* $\sum_{k\in\mathcal{K}} q^{\omega}_{|\mathcal{N}_{\mathcal{F}}|k} = \prod_{z\in\mathcal{N}_{\mathcal{F}}} \sum_{k\in\mathcal{K}} {\mathbb{P}^{\omega}}_{zs(z,\omega)m(z,\omega)k} \hat{y}_{zk} \left(\sum_{(i,j)\in\mathcal{A}} t_{ij} f^{\omega}_{ij} \right)$ *in scenario* ω*.*

Proof. For a given solution $\hat{\mathbf{y}}$, let the mapping $k(z|\hat{\mathbf{y}})$ for a given facility z be the value of k such that $\hat{y}_{zk} = 1$. For notational simplicity, we omit the $\hat{\mathbf{y}}$ notation and use $\hat{k}(z)$.

Note that $\forall r \in \mathcal{N}_{\mathcal{F}}$, we have, $\hat{y}_{rk} = 0, \forall k \in \mathcal{K} \setminus \hat{k}(r)$. Therefore, according to Eqs. (3.7), we have

$$q^{\omega}_{rk} = 0 \, \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K} \setminus \hat{k}(r), \tag{3.9}$$

$$q^{\omega}_{r\hat{k}(r)} \geq 0 \, \forall r \in \mathcal{N}_{\mathcal{F}}, \hat{k}(r) \in \mathcal{K} \tag{3.10}$$

By assumption, the post-disruption capacity states of the facilities are independent. Using Eqs. (3.9)–(3.10), Eq. (3.5) becomes:

$$q^{\omega}_{1\hat{k}(1)} = \mathbb{P}^{\omega}{}_{1s(1,\omega)m(1,\omega)\hat{k}(1)} \left(\textstyle\sum_{(i,j)\in\mathcal{A}} t_{ij} f^{\omega}_{ij} \right) \tag{3.11}$$

Given that $\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)k} > \epsilon$, Eqs. (3.6) have the following form,

$$q^{\omega}_{(r-1)1} + \cdots + q^{\omega}_{(r-1)|\mathcal{K}|} = \frac{1}{\mathbb{P}^{\omega}{}_{rs(r,\omega)m(r,\omega)1}} q^{\omega}_{r1} + \cdots + \frac{1}{\mathbb{P}^{\omega}{}_{rs(r,\omega)m(r,\omega)|\mathcal{K}|}} q^{\omega}_{r|\mathcal{K}|} \quad \forall r = 2, ..., |\mathcal{N}_{\mathcal{F}}| \tag{3.12}$$

Applying Eqs. (3.9)–(3.10), we have

$$q^{\omega}_{r\hat{k}(r)} \quad = \mathbb{P}^{\omega}{}_{rs(r,\omega)m(r,\omega)\hat{k}(r)} q^{\omega}_{(r-1)\hat{k}(r-1)} \forall r = 2, \ldots, |\mathcal{N}_{\mathcal{F}}| \tag{3.13}$$

For any $r \geq 2$, Eqs. (3.13) results in the following expression:

$$
\begin{aligned}
q^{\omega}_{r\hat{k}(r)} &= \left(\mathbb{P}^{\omega}{}_{rs(r,\omega)m(r,\omega)\hat{k}(r)}\right)\left(\mathbb{P}^{\omega}{}_{(r-1)s(r-1,\omega)m(r-1,\omega)\hat{k}(r-1)}\right)\cdots \\
&\quad \cdots\left(\mathbb{P}^{\omega}{}_{2s(2,\omega)m(2,\omega)\hat{k}(2)}q^{\omega}_{1\hat{k}(1)}\right) \\
&= \left(\mathbb{P}^{\omega}{}_{rs(r,\omega)m(r,\omega)\hat{k}(r)}\right)\left(\mathbb{P}^{\omega}{}_{(r-1)s(r-1,\omega)m(r-1,\omega)\hat{k}(r-1)}\right)\cdots \\
&\quad \cdots\mathbb{P}^{\omega}{}_{1s(1,\omega)m(1,\omega)\hat{k}(1)}\left(\textstyle\sum_{(i,j)\in\mathcal{A}} t_{ij}f^{\omega}_{ij}\right)
\end{aligned}
$$

Thus, for $r = |\mathcal{N}_{\mathcal{F}}|$, we have,

$$
q^{\omega}_{|\mathcal{N}_{\mathcal{F}}|\hat{k}(|\mathcal{N}_{\mathcal{F}}|)} = \prod_{z=1}^{|\mathcal{N}_{\mathcal{F}}|}\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)\hat{k}(z)}\left(\sum_{(i,j)\in\mathcal{A}} t_{ij}f^{\omega}_{ij}\right)
$$

Therefore, considering all possible k in set $\mathcal{K}$, we get,
$\sum_{k\in\mathcal{K}} q^{\omega}_{|\mathcal{N}_{\mathcal{F}}|k} = \prod_{z\in\mathcal{N}_{\mathcal{F}}}\sum_{k\in\mathcal{K}}\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)k}\hat{y}_{zk}\left(\sum_{(i,j)\in\mathcal{A}} t_{ij}f^{\omega}_{ij}\right)$.

□

We present a numerical example in Appendix C to demonstrate the computation of the probability-weighted PDTC in a scenario ω using the recursive Eqs. (3.5)–(3.8).

Linearized Formulation

We see that the recursive Eqs. (3.5) and (3.7) contain the product of the binary protection decision variables y_{zk} and the continuous flow variables f^{ω}_{ij}. Therefore, even after implementing the probability-chain reformulation, there is non-linearity in the model. However, this non-linearity can be linearized using McCormick linearization [92], where an additional continuous variable $v^{\omega}_{(i,j)rk}$ is introduced to replace the product of the variables y_{zk} and f^{ω}_{ij}. After implementing the probability-chain reformulation and the McCormick linearization, the original nonlinear two-stage stochastic programming model

(3.2) reduces to the reformulated mixed-integer linear stochastic programming model as follows (MinExpCost-R):

$$\min \quad \sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \left(\sum_{k\in\mathcal{K}} q^{\omega}_{|\mathcal{N}_{\mathcal{F}}|k} \right) \tag{3.14a}$$

$$\text{s.t.} \quad q^{\omega}_{1k} = \mathbb{P}^{\omega}_{1s(1,\omega)m(1,\omega)k} \left(\sum_{(i,j)\in\mathcal{A}} t_{ij} v^{\omega}_{(ij)1k} \right) \qquad \forall k \in \mathcal{K}, \omega \in \Omega \tag{3.14b}$$

$$\sum_{k\in\mathcal{K}} q^{\omega}_{(r-1)k} = \sum_{k\in\mathcal{K}} \frac{1}{max\left\{\mathbb{P}^{\omega}_{rs(r,\omega)m(r,\omega)k}, \epsilon\right\}} q^{\omega}_{rk} \quad \forall r = 2, .., |\mathcal{N}_{\mathcal{F}}|, \omega \in \Omega \tag{3.14c}$$

$$q^{\omega}_{rk} \leq \sum_{(i,j)\in\mathcal{A}} t_{ij} v^{\omega}_{(ij)rk} \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, \omega \in \Omega \tag{3.14d}$$

$$v^{\omega}_{(ij)rk} \leq f^{\omega}_{ij} \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A}, \omega \in \Omega \tag{3.14e}$$

$$v^{\omega}_{(ij)rk} \leq U_{ij} y_{rk} \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A}, \omega \in \Omega \tag{3.14f}$$

$$v^{\omega}_{(ij)rk} \geq f^{\omega}_{ij} - (1 - y_{rk}) U_{ij} \quad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A}, \omega \in \Omega \tag{3.14g}$$

$$v^{\omega}_{(ij)rk} \geq 0 \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A}, \omega \in \Omega \tag{3.14h}$$

(3.2b) – (3.2e)

(3.3b) – (3.3f)

The objective function (3.14a) seeks to minimize the expected PDTC over all scenarios. Constraints (3.14b)–(3.14d) are linearized version of the recursive equations (3.5)–(3.7) obtained by implementing the McCormick linearization to compute the PPWPDTC for each scenario ω. Constraints (3.14e)–(3.14g) are introduced to implement the McCormick linearization that linearizes the product of the variables y_{zk} and f^{ω}_{ij}.

3.4.2 Accelerated L-shaped Decomposition Algorithm

Our two-stage stochastic programming model has integer decision variables—link construction decisions and protection investment decisions—in the first-stage model, and continuous decision variables—the amount of flow through the links—in the second-stage model for each scenario. This structure of the problem is suitable for the implementation of a well known

decomposition algorithm, L-shaped decomposition algorithm, where the original problem is decomposed into a master problem with the complicating integer first-stage variables, and scenario sub-problems with the continuous second-stage decision variables. This structural advantage motivates us to implement a L-shaped decomposition algorithm to solve the model. L-shaped decomposition algorithm is also applied to solve two-stage stochastic programming model in a wide range of application areas including but not limited to supply chain network design [124, 66], cyber security [18], and inventory control [123].

Classical L-Shaped Decomposition

L-shaped decomposition is Benders decomposition [13] applied to the special structure of a stochastic program. In Benders decomposition, a mixed-integer linear program is decomposed into a master problem (MP) with complicating integer variables and a linear sub-problem with continuous variables. Using duality of the linear sub-problem and the associated extreme rays and points, feasibility and optimality cuts are generated for the master problem. Due to the difficulty in enumerating all the cuts, Benders [13] proposed a relaxation strategy that adds the cuts iteratively after solving the master problem and sub-problem in each iteration.

Van Slyke and Wets [140] first introduced the L-shaped method by using Benders decomposition to solve a two-stage stochastic program. A general two-stage stochastic program with first-stage binary variable (x) and second-stage continuous variables (y) is as follows:

$$\min \quad c^T x + \mathbb{E}[Q(x,\omega)] \tag{3.15a}$$

$$\text{s.t.} \quad Ax \leq B \tag{3.15b}$$

$$Tx + W^\omega y^\omega \leq h^\omega \qquad \forall \omega \in \Omega \tag{3.15c}$$

$$x \in \{0,1\} \tag{3.15d}$$

$$y^\omega \geq 0 \qquad \forall \omega \in \Omega \tag{3.15e}$$

where $Q(x,\omega) = min\,(a^\omega)^T y^\omega$ is the second-stage objective value for scenario ω. Denoting p^ω be the probability of scenario ω, we have $\mathbb{E}[Q(x,\omega)] = \sum_{\omega\in\Omega} p^\omega Q(x,\omega)$. L-shaped decomposition algorithm computes an approximation of (3.15). The original two-stage stochastic program (3.15) is decomposed into a master problem with the first-stage variables and sub-problems for each scenario with the second-stage variables. The first-stage variables (x) connect the master problem with the scenario sub-problems. The primal sub-problem for each scenario ω consists of the objective function $\min\,(a^\omega)^T y^\omega$ and constraints (3.15c) and (3.15e) for scenario ω. The dual of the primal sub-problem (DSP) for scenario ω and the master problem (MP) are presented as follows:

$$\begin{aligned}
[\text{DSP}] \quad \max \quad & (u^\omega)^T (h^\omega - T\hat{x}) && \text{(3.16a)}\\
\text{s.t.} \quad & (W^\omega)^T u^\omega \le a^\omega && \text{(3.16b)}\\
& u^\omega \le 0 && \text{(3.16c)}
\end{aligned}$$

where $(u^\omega)^T$ are the dual variables in scenario ω and $\hat{x}$ is the value of the first-stage variable computed in the master problem. Using this dual sub-problem for each scenario, the optimality and feasibility cuts are constructed for the master problem.

$$\begin{aligned}
[\text{MP}] \quad \min \quad & c^T x + \theta && && \text{(3.17a)}\\
\text{s.t.} \quad & Ax \le B && && \text{(3.17b)}\\
& \theta \ge \sum_{\omega} p^\omega \left[\left((u^\omega)^T \right)^i (h^\omega - Tx) \right] && \forall i \in \mathcal{E} && \text{(3.17c)}\\
& \theta \ge \sum_{\omega} p^\omega \left[\left((u^\omega)^T \right)^r (h^\omega - Tx) \right] && \forall r \in \mathcal{R} && \text{(3.17d)}\\
& x \in \{0,1\}, \theta \in \mathbb{R} && && \text{(3.17e)}
\end{aligned}$$

where $\mathcal{E}$ and $\mathcal{R}$ are the sets of extreme points and rays, respectively, of the polyhedron defined by (3.16b)–(3.16c) of all the scenarios. The objective value of this master problem provides a valid lower bound to the optimal objective value of the original problem (3.15)

at each iteration. The feasibility cut (3.17d) is added to the master problem if the DSP is unbounded for a given master problem solution [123]. The optimality cuts (3.17c) are linear approximations of $\mathbb{E}[Q(x,\omega)]$ and are added iteratively to the master problem.

L-shaped Decomposition for the Proposed Model

In this sub-section, we describe the implementation of the L-shaped decomposition algorithm for our two-stage stochastic programming model.

Lower Bound Calculation

To obtain a lower bound for the optimal objective value of our PNDIPP, we solve the master problem presented as follows (PNDIPP-MP):

$$lb^e = \min \quad \theta \tag{3.18a}$$

s.t.

$$\theta \geq \sum_{z\in\mathcal{N}_\mathcal{F}} \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} b^e_{zk(ij)} U_{ij} y_{zk} + \sum_{(i,j)\in\mathcal{A}} d^e_{ij} U_{ij} x_{\ell(i,j)} + g^e, \;\; e = 1, 2.. \tag{3.18b}$$

(3.2b) – (3.2e)

Here, θ represents the expected PDTC upto iteration e. $b^e_{zk(i,j)}$, $d^e_{i,j}$, and g^e are the coefficients of the optimality cut (3.18b) at iteration e. Computation of these cut coefficients are described in detail later in this Section. The objective function (3.18a) and the constraint (3.18b) together ensure that the objective of the master problem is to minimize the maximum of all the expected PDTC upto iteration e. As our scenario sub-problems are feasible for any master problem solution, we add only the optimality cuts (3.18b) to the master problem.

Upper Bound Calculation

At each iteration of the algorithm, we solve the sub-problem (MinCost-R) (3.19) for each scenario ω using the optimal link opening decisions, $\hat{\mathbf{x}}^e$, and protection decisions, $\hat{\mathbf{y}}^e$ obtained from the master problem solution.

$$Q(\hat{\mathbf{x}}^e, \hat{\mathbf{y}}^e, \omega) = \min \quad \sum_{k \in \mathcal{K}} q^{\omega}_{|\mathcal{N}_{\mathcal{F}}|k} \tag{3.19a}$$

$$\text{s.t.} \quad q^{\omega}_{1k} = \mathbb{P}^{\omega}_{1s(1,\omega)m(1,\omega)k} \left(\sum_{(i,j) \in \mathcal{A}} t_{ij} v^{\omega}_{(ij)1k} \right) \qquad \forall k \in \mathcal{K} \quad (\rho) \tag{3.19b}$$

$$\sum_{k \in \mathcal{K}} q^{\omega}_{(r-1)k} = \sum_{k \in \mathcal{K}} \frac{1}{max \left\{ \mathbb{P}^{\omega}_{rs(r,\omega)m(r,\omega)k}, \epsilon \right\}} q^{\omega}_{rk} \; \forall r = 2, .. \mid \mathcal{N}_{\mathcal{F}} \mid \tag{3.19c}$$

$$q^{\omega}_{rk} \leq \sum_{(i,j) \in \mathcal{A}} t_{ij} v^{\omega}_{(ij)rk} \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K} \quad (\phi) \tag{3.19d}$$

$$v^{\omega}_{(ij)rk} \leq f^{\omega}_{ij} \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A} \quad (\delta) \tag{3.19e}$$

$$v^{\omega}_{(ij)rk} \leq U_{ij} \hat{y}^e_{rk} \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A} \quad (\pi) \tag{3.19f}$$

$$v^{\omega}_{(ij)rk} \geq f^{\omega}_{ij} - (1 - \hat{y}^e_{rk}) U_{ij} \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A} \quad (\beta) \tag{3.19g}$$

$$f^{\omega}_{ij} \leq U_{ij} \hat{x}^e_{\ell(i,j)} \qquad \forall (i,j) \in \mathcal{A} \quad (\gamma) \tag{3.19h}$$

$$\sum_{(i,j) \in \mathcal{RS}(n)} f^{\omega}_{ji} - \sum_{(i,j) \in \mathcal{FS}(n)} f^{\omega}_{ij} = d_n \qquad \forall n \in \mathcal{N}_{\mathcal{D}} \quad (\alpha) \tag{3.19i}$$

$$\sum_{(i,j) \in \mathcal{FS}(z)} f^{\omega}_{ij} - \sum_{(i,j) \in \mathcal{RS}(z)} f^{\omega}_{ji} \leq a^{\omega}_{zs} \qquad \forall z \in \mathcal{N}_{\mathcal{F}} \quad (\beta) \tag{3.19j}$$

$$\sum_{(i,j) \in \mathcal{FS}(D)} f^{\omega}_{ij} \leq C_D \quad (\psi) \tag{3.19k}$$

$$f^{\omega}_{ij} \geq 0 \qquad \forall (i,j) \in \mathcal{A} \tag{3.19l}$$

$$v^{\omega}_{(ij)rk} \geq 0 \qquad \forall r \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K}, (i,j) \in \mathcal{A} \tag{3.19m}$$

The objective function (3.19a) seeks to minimize the probability-weighted post-disruption transportation cost (PWPDTC) in scenario ω. Constraints (3.19b)–(3.19g) are the same as constraints (3.14b)–(3.14g) for a particular scenario ω. Constraints (3.19h)–(3.19k) are the same as (3.3b)–(3.3e).

We use the sub-problem objective values from all the scenarios to compute the expected PDTC. The Greek letters in the parentheses on the right side of the constraints in the sub-problem are the corresponding dual variables. We use the optimal value of these dual variables to compute the coefficients of the optimality cut (3.18b) at iteration $e+1$ as follows:

$$b_{zk(i,j)}^{e+1} = \sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \hat{\pi}_{rk(i,j)}^{\omega} + \sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \hat{\mu}_{rk(i,j)}^{\omega}$$

$$d_{ij}^{e+1} = \sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \hat{\gamma}_{i,j}^{\omega}$$

$$g^{e+1} = -\sum_{r\in\mathcal{N}_{\mathcal{F}}} \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} \left(\sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \hat{\mu}_{rk(i,j)}^{\omega}\right) U_{\ell} + \sum_{n\in\mathcal{N}_{\mathcal{D}}} \left(\sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \hat{\alpha}_n^{\omega}\right) d_n$$
$$+ \sum_{z\in\mathcal{N}_{\mathcal{F}}} \left(\sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \hat{\beta}_z^{\omega} a_{zs}^{\omega}\right) + C_D \left(\sum_{\omega\in\Omega} \mathbb{P}_h^{\omega} \hat{\psi}^{\omega}\right)$$

The complete L-shaped decomposition algorithm is presented in Algorithm 1.

Acceleration Techniques

Based on the preliminary experiments, we observed that the basic L-shaped decomposition algorithm requires a large number of iterations to converge. Moreover, as we add optimality cut to the master problem at each iteration of the algorithm, the size of the master problem increases over the iterations. This increases the computation time of the master problem over the iterations and eventually increases the total runtime of the algorithm. Therefore, to improve the computational efficiency of the algorithm and thus speed up the convergence, we implement the following enhancements to the algorithm.

Multiple Optimality Cuts

In the L-shaped decomposition algorithm, we add one optimality cut at each iteration to the master problem (3.18) by aggregating the cut coefficients from all the scenarios. According to Birge and Louveaux [21], in another variant of the L-shaped algorithm, known as *multi-cut L-shaped decomposition algorithm,* $|\Omega|$ cuts are added to the master problem at each iteration, where each cut is derived from each scenario sub-problem. The rationale of disaggregating the cuts is that using the approximations of all the sub-problems individually provides more information to the master problem than a single aggregated cut and thereby reduce the number of iterations to converge [22].

Algorithm

Algorithm 1 L-shaped Decomposition Algorithm

function L-SHAPEDDECOMPOSITION

  initialize. Set $ub \leftarrow \infty$, $lb \leftarrow 0$, $e \leftarrow 1$, $\delta \leftarrow 0.01$, $\mathbf{x}^* \leftarrow \mathbf{0}$, $\mathbf{y}^* \leftarrow \mathbf{0}$

  while $\frac{ub-lb}{ub} > \delta$ **do**

    Solve master problem (3.18), returning lb^e, $\hat{\mathbf{x}}^e$ and $\hat{\mathbf{y}}^e$

    if $lb^e > lb$ **then** $lb \leftarrow lb^e$, $\mathbf{x}^* \leftarrow \hat{\mathbf{x}}^e$ and $\mathbf{y}^* \leftarrow \hat{\mathbf{y}}^e$

    end if

    Solve subproblem MINCOST-R($\hat{\mathbf{x}}^e$,$\hat{\mathbf{y}}^e$) $\forall \omega \in \Omega$, returning optimal flow decisions $\hat{\mathbf{f}}^\omega$ and $Q(\hat{\mathbf{x}}^e, \hat{\mathbf{y}}^e, \omega)$

    Compute $f(\hat{\mathbf{x}}^e, \hat{\mathbf{y}}^e) := \sum_{\omega \in \Omega} \mathbb{P}_h^\omega Q(\hat{\mathbf{x}}^e, \hat{\mathbf{y}}^e, \omega)$

    if $f(\hat{\mathbf{x}}^e, \hat{\mathbf{y}}^e) < ub$ **then** $ub \leftarrow f(\hat{\mathbf{x}}^e, \hat{\mathbf{y}}^e)$, $\mathbf{x}^* \leftarrow \hat{\mathbf{x}}^e$ and $\mathbf{y}^* \leftarrow \hat{\mathbf{y}}^e$

    end if

    if $\frac{ub-lb}{ub} \leq \delta$ **then break.** Otherwise, go to next step.

    end if

    Add the optimality cut (3.18b) to the master problem (3.18) using the optimal dual multipliers of the sub-problems

    $e \leftarrow e + 1$.

  end while

  return Optimal link opening decisions, $\mathbf{x}^*$ and protection investment decisions, $\mathbf{y}^*$

end function

In this multi-cut version, the master problem (3.18) is formulated as follows:

$$
\begin{aligned}
lb^e = \min \quad & \sum_{\omega \in \Omega} \mathbb{P}_h^{\omega} \theta^{\omega} \\
\text{s.t.} \quad & \\
& \theta^{\omega} \geq \sum_{z \in \mathcal{N}_{\mathcal{F}}} \sum_{k \in \mathcal{K}} \sum_{(i,j) \in \mathcal{A}} b_{zk(ij)}^{\omega e} U_{ij} y_{zk} + \sum_{(i,j) \in \mathcal{A}} d_{ij}^{\omega e} U_{ij} x_{\ell(i,j)} \\
& \qquad + g^{\omega e}, \ e = 1, 2, \ldots; \omega = 1, 2, \ldots, |\Omega| \qquad (3.20a) \\
& (3.2b) - (3.2e)
\end{aligned}
$$

The cut coefficients for iteration $e+1$ are as follows:

$$
\begin{aligned}
b^{\omega, e+1} &= \hat{\pi}_{rk(ij)}^{\omega} + \hat{\mu}_{rk(ij)}^{\omega} \\
d^{\omega, e+1} &= \hat{\gamma}_{ij}^{\omega} \\
g^{\omega, e+1} &= -\sum_{r \in \mathcal{N}_{\mathcal{F}}} \sum_{k \in \mathcal{K}} \sum_{(i,j) \in \mathcal{A}} \hat{\mu}_{rk(ij)}^{\omega} U_{ij} + \sum_{n \in \mathcal{N}_{\mathcal{D}}} \hat{\alpha}_n^{\omega} d_n \\
&\quad + \sum_{z \in \mathcal{N}_{\mathcal{F}}} \hat{\beta}_z^{\omega} a_{zs}^{\omega} + C_D \hat{\psi}^{\omega}
\end{aligned}
$$

This multi-cut version reduces the total number of iterations for the algorithm to converge while increasing the computation time of each iteration due to the larger size of the master problem.

Trust Region Cuts (TR)

At the initial iterations of the L-shaped decomposition algorithm, the master problem solutions are very divergent resulting in a slower convergence. To mitigate this drawback, Linderoth and Wright [83] proposed to add constraints to limit the distance between the master problem solutions of two consecutive iterations within a trust region. In this research, we add a trust region cut to the master problem to stabilize the master problem solutions in the initial iterations. Our trust region cut bounds the Hamming distance [50], the number of bits changed in the solution, between the master problem solutions of two consecutive iterations. Denoting $\hat{\mathbf{x}}^e$ and $\hat{\mathbf{y}}^e$ be the master problem solutions obtained at iteration e, and

$\hat{\mathbf{Y}_1}^{\mathbf{e}} := \{(z,k) : y_{zk} = 1\}$, and $\hat{\mathbf{X}}_1^{\mathbf{e}} := \{\ell : x_\ell = 1\}$. We add the following trust region cut at iteration $e+1$:

$$\sum_{(z,k)\notin \hat{\mathbf{Y}}_1^e} y_{zk} + \sum_{(z,k)\in \hat{\mathbf{Y}}_1^e} (1 - y_{zk}) \leq \Delta \times |\hat{\mathbf{y}}| \tag{3.21}$$

$$\sum_{\ell\notin \hat{\mathbf{X}}_1^e} x_\ell + \sum_{\ell\in \hat{\mathbf{X}}_1^e} (1 - x_\ell) \leq \Delta \times |\hat{\mathbf{x}}| \tag{3.22}$$

where $\hat{\mathbf{y}}$ and $\hat{\mathbf{x}}$ are the protection and link construction decision vectors. Constraints (3.21) and (3.22) force that the maximum change in the decisions from iteration e to $e+1$ is $\Delta\%$. The left hand side of the constraint (3.21) computes the Hamming distance between the decisions of iteration e and iteration $e+1$. Master problem with this trust region cut does not provide a valid lower bound; keeping this trust region cut throughout the algorithm does not guarantee convergence [124]. Therefore, we add this cut at the few initial iterations of the algorithm and then remove it when the iterates become stable. Computational experiments demonstrate that applying this cut at the 10 initial iterations of the multi-cut L-shaped algorithm to limit the maximum change in master problem solutions to 40% improves the algorithm runtime.

Knapsack Inequality (KI)

According to Santoso et al. [124], if we have a good upper bound in L-shaped decomposition algorithm, adding a knapsack inequality along with the optimality cut has a significant effect in producing a quality solution. We add the optimality cut: $\theta \geq \sum_{z\in\mathcal{N}_\mathcal{F}} \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} b^e_{zk(ij)}\ U_{ij} y_{zk} + \sum_{(i,j)\in\mathcal{A}} d^e_{ij} U_{ij} x_{\ell(i,j)} + g^e$ to the master problem at each iteration. As for minimization problem, the L-shaped decomposition algorithm ensures that $ub \geq \theta$, we add the following valid inequality to the master problem at iteration $e+1$:

$$ub^e - g^e \geq \sum_{z\in\mathcal{N}_\mathcal{F}} \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} b^e_{zk(ij)} U_{ij} y_{zk} + \sum_{(i,j)\in\mathcal{A}} d^e_{ij} U_{ij} x_{\ell(i,j)} \tag{3.23}$$

Here, ub^e is the best upper bound available at the end of iteration e. Santoso et al. [124] also mentioned that state-of-the-art solvers can derive various valid inequalities from the knapsack inequality (Eq. 3.23) and thereby speed up the convergence.

Pareto-Optimality Cuts

Sub-problems having network sub-structure usually have multiple dual optimal solutions, which makes it possible to obtain alternatives for the optimality cuts [124]. According to Magnanti and Wong [87], adding cuts that are not dominated by other optimality cuts could improve the convergence of the L-shaped decomposition algorithm. Assuming (b^1, d^1, g^1) and (b^2, d^2, g^2) are the alternative optimality cut coefficients corresponding to two alternative optimal dual solutions, a cut generated from coefficients (b^1, d^1, g^1) dominates the cut generated from coefficients (b^2, d^2, g^2) if

$$\sum_{z\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}} b^1_{zk(ij)}U_{ij}y^*_{zk} + \sum_{(i,j)\in A} d^1_{ij}U_{ij}x^*_{\ell(i,j)} + g^1 > \sum_{z\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}} b^2_{zk(ij)}U_{ij}y^*_{zk} + \sum_{(i,j)\in\mathcal{A}} d^2_{ij}U_{ij}x^*_{\ell(i,j)} + g^2$$

where y^*_{zk} and x^*_ℓ are the optimal solutions. As it is not possible to know the optimal solution *a priori,* Magnanti and Wong [87] introduced a core point—a point in the relative interior of the convex hull of the feasible region to use as a proxy for the optimal solution. However, as there are challenges in finding a core point at each iteration of the algorithm [96, 108], Papadakos [109] showed that an approximate core point— a convex combination of the core point in the previous iteration and the current master problem solution—can be used to generate Pareto-optimal cuts. The master problem solution of the first iteration is taken as the initial core point.

Furthermore, Papadakos [109] proposed a modified Magnanti-Wong (MMW) problem to avoid the computational complexities associated with solving the subproblem-dependent Magnanti-Wong problem in generating the Pareto-optimal cuts. In this research, we solve the MMW problem that does not depend on the solutions of the scenario sub-problems. We solve the MMW problem (3.24) for each scenario ω at each iteration of our algorithm using the approximate core points to generate Pareto-optimal cuts. This formulation is the dual

of the primal scenario sub-problem (3.19), with the difference that the objective function is evaluated at the core point $(\boldsymbol{x}^0, \mathbf{y}^0)$. A new core point at each iteration e is computed as follows:

$$y_{zk}^{0e} \leftarrow \lambda y_{zk}^{e-1} + (1-\lambda)\hat{y}_{zk}^{e}$$
$$x_{\ell}^{0e} \leftarrow \lambda x_{\ell}^{e-1} + (1-\lambda)\hat{x}_{\ell}^{e}$$

where $0 < \lambda < 1$. $(y_{zk}^{0e},\ x_{\ell}^{0e})$ represent the core point at iteration e, and $\hat{x}_{\ell}^{e}$ and $\hat{y}_{zk}^{e}$ are the master problem solutions at iteration e. Computational results demonstrate that the value of $\lambda = 0.15$ provides the best performance of the Pareto-optimal cuts.

We solve the sub-problem-independent Magnanti-Wong problem (3.24) for each scenario ω at each iteration of the L-shaped algorithm to construct the Pareto-optimality cuts. This formulation is the dual of the primal scenario sub-problem (3.19) with the difference that the objective function is evaluated at the core point $(\boldsymbol{x}^0, \mathbf{y}^0)$. In this formulation, constraints (3.24a)–(3.24j) force the dual feasible region of the sub-problem corresponding to scenario ω.

Valid Inequality

We observed that at the initial iterations of our L-shaped algorithm, the rate of improvement of the lower bound is vey low. Because, at the beginning, with very few cuts, the master problem constructs a very small number of links and invests protection to few or no facilities. This eventually results in a small lower bound. With very few links opened and protection investment, demands in the scenario sub-problems are mostly satisfied by the dummy facility with a high transportation cost, resulting in a very large upper bound. Therefore, to improve the master problem solution (lower bound) at the initial iterations, we add a valid inequality to force the master problem to open more links and invest in more facility protection.

$$\max \quad \sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}} \pi_{rk(i,j)}U_{ij}y^0_{rk} + \sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}} \mu_{rk(i,j)}\left[-U_{ij}(1-y^0_{rk})\right] + \sum_{(i,j)\in\mathcal{A}_p}\gamma_{ij}U_{ij}x^0_{\ell(i,j)}$$
$$+ \sum_{n\in\mathcal{N}_\mathcal{D}}\alpha_n d_n + \sum_{z\in\mathcal{N}_\mathcal{F}}\beta_z a^\omega_{zs} + \psi C_D$$

$$\text{s.t.} \quad \rho_k + \phi_{1k} + \nu_2 \le 0 \quad \forall k\in\mathcal{K} \tag{3.24a}$$

$$-\frac{1}{max\{\mathbb{P}^\omega_{rs(r,\omega)m(r,\omega)k},\epsilon\}}\nu_r + \nu_{r+1} + \phi_{rk} \le 0 \quad \forall r = 2,..,\mid\mathcal{N}_\mathcal{F}\mid -1, k\in\mathcal{K} \tag{3.24b}$$

$$-\frac{1}{max\{\mathbb{P}^\omega_{|\mathcal{N}_\mathcal{F}|s(|\mathcal{N}_\mathcal{F}|,\omega)m(|\mathcal{N}_\mathcal{F}|,\omega)k},\epsilon\}}\nu_{|\mathcal{N}_\mathcal{F}|} + \phi_{|\mathcal{N}_\mathcal{F}|k} \le 1 \quad \forall k\in\mathcal{K} \tag{3.24c}$$

$$-\mathbb{P}^\omega_{1s(1,\omega)m(1,\omega)k}t_{ij}\rho_k - t_{ij}\phi_{1k} + \delta_{1k(i,j)} + \pi_{1k(i,j)}$$
$$+\mu_{1k(i,j)} \le 0 \quad \forall k\in\mathcal{K}, (i,j)\in\mathcal{A} \tag{3.24d}$$

$$-t_{ij}\phi_{rk} + \delta_{rk(i,j)} + \pi_{rk(i,j)} + \mu_{rk(i,j)} \le 0 \quad \forall r = 2,...,\mid\mathcal{N}_\mathcal{F}\mid, k\in\mathcal{K}, (i,j)\in\mathcal{A} \tag{3.24e}$$

$$-\sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\delta_{rk(i,j)} - \sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\mu_{rk(i,j)} + \gamma_{ij} - \alpha_i - \beta_j \le 0 \quad \forall(i,j)\in\mathcal{A}_{p1} \tag{3.24f}$$

$$-\sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\delta_{rk(i,j)} - \sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\mu_{rk(i,j)} + \gamma_{ij} + \alpha_j + \beta_i \le 0 \quad \forall(i,j)\in\mathcal{A}_{p2} \tag{3.24g}$$

$$-\sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\delta_{rk(i,j)} - \sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\mu_{rk(i,j)} + \gamma_{ij} + \alpha_j - \alpha_i \le 0 \quad \forall(i,j)\in\mathcal{A}_{p3} \tag{3.24h}$$

$$-\sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\delta_{rk(i,j)} - \sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\mu_{rk(i,j)} + \gamma_{ij} + \beta_i - \beta_j \le 0 \quad \forall(i,j)\in\mathcal{A}_{p4} \tag{3.24i}$$

$$-\sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\delta_{rk(i,j)} - \sum_{r\in\mathcal{N}_\mathcal{F}}\sum_{k\in\mathcal{K}}\mu_{rk(i,j)} + \alpha_j + \psi \le 0 \quad \forall(i,j)\in\mathcal{A}\setminus\mathcal{A}_p \tag{3.24j}$$

$$\rho,\nu,\alpha\in\mathbb{R};\ \phi,\delta,\pi,\gamma,\beta,\psi\in\mathbb{R}^-,\mu\in\mathbb{R}^+ \tag{3.24k}$$

According to Birge and Louveaux [21], the following basic inequalities are valid in a stochastic program:

$$WS \le RP$$

$$EV \le WS$$

Therefore, $EV \le RP$. Here WS, RP, and EV denote optimal objective values of the wait-and-see, stochastic program, and mean value problem, respectively. The mean value

problem is the deterministic model that considers the expected value of the random variables. We use this valid inequality ($EV \leq RP$) to force the master problem to generate a solution that results in a larger objective value than the objective value of the mean value problem (1) in Appendix B. This ensures that more number of links are opened and protections are allocated to the facilities, resulting in a larger lower bound at the initial iterations of the L-shaped algorithm.

We add the following constraints to the master problem in the single-cut L-shaped algorithm:

$$\theta \geq \sum_{(i,j)\in\mathcal{A}} t_{ij} f'_{ij} \tag{3.25}$$

$$f'_{ij} \leq U_\ell x_{\ell(i,j)} \qquad \forall (i,j) \in \mathcal{A} \tag{3.26}$$

$$\sum_{(i,j)\in\mathcal{RS}(n)} f'_{ji} - \sum_{(i,j)\in\mathcal{FS}(n)} f'_{ij} = d_n \qquad \forall n \in \mathcal{N}_\mathcal{D} \tag{3.27}$$

$$\sum_{(i,j)\in\mathcal{FS}(z)} f'_{ij} - \sum_{(i,j)\in\mathcal{RS}(z)} f'_{ji} \leq \sum_{k\in\mathcal{K}} \left(\mathbb{E}\left[a_{zs} | y_{zk} = 1\right] y_{zk}\right) \qquad \forall z \in \mathcal{N}_\mathcal{F} \tag{3.28}$$

$$\sum_{(i,j)\in\mathcal{FS}(D)} f'_{ij} \leq C_D \tag{3.29}$$

$$f'_{ij} \geq 0 \qquad \forall (i,j) \in \mathcal{A} \tag{3.30}$$

The constraint (3.25) ensures that the master problem objective value is at least as large as the objective value of the mean value problem. The variable f'_{ij} is an auxiliary variable representing the flow through an arc (i,j). Constraints (3.26) force that no flow is possible through an arc (i,j) if the corresponding link ℓ is not opened. Constraints (3.27)–(3.28) are the flow balance constraints of the demand points and the facilities, respectively. Constraint (3.29) is the dummy facility's capacity constraint.

In the multi-cut version of the L-shaped algorithm, we add the following constraints to the master problem:

$$\sum_{\omega\in\Omega} \mathbb{P}_h^\omega \theta^\omega \geq \sum_{(i,j)\in\mathcal{A}} \quad t_{ij} f'_{ij}$$

$$(3.26) - (3.30)$$

3.4.3 Estimating the Probability of Capacity States

In this research, we implement several predictive modeling techniques that can analyze historical data to estimate the probabilities $\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)k}$ for a given protection level k and disruption intensity level m. The predictive modeling techniques are described as follows:

Multinomial Logistic Regression

In this research, each facility can be in one of the possible capacity states defined by the set $\mathcal{S}$. As multiple capacity states are possible for each facility, the prediction of capacity states can be considered as a multi-class classification problem. Therefore, we can use a multinomial logistic regression (MLR) model to predict the probability $\mathbb{P}^{\omega}{}_{zs(z,\omega)m(z,\omega)k}$ of a facility to arrive in capacity state s (class s) for a given protection and disruption intensity levels. In MLR model, the response variable is the capacity states ($s = 0, 1, 2, \ldots, |\mathcal{S}|$) of the facilities whereas the predictor variables are the protection amount (in U.S. dollars) (y_z) allocated to facility z and the disruption intensity level m. The estimated probability that a facility z is in capacity state s for a given protection level k or given protection y_z and the disruption intensity level m can be computed from the following equations:

$$\mathbb{P}_{mk}(s) = \frac{exp(\beta_{0s}+\beta_{1s}y_z+\beta_{2s}m)}{1+\sum_{l}^{|\mathcal{S}|-1} exp(\beta_{0l}+\beta_{1l}y_z+\beta_{2l}m)} \quad \forall s = 1, 2, ..., |\mathcal{S}| - 1 \tag{3.31}$$

$$\mathbb{P}_{mk}(|\mathcal{S}|) = \frac{1}{1+\sum_{l}^{|\mathcal{S}|-1} exp(\beta_{0l}+\beta_{1l}y_z+\beta_{2l}m)} \tag{3.32}$$

where $\beta_{0s}, \beta_{1s}, \text{and } \beta_{2s}$ are the parameter estimates of the logistic regression model corresponding to class s or capacity state s. Equation (3.31) computes the probability $\mathbb{P}_{mk}(s)$ that a facility has post-disruption capacity state s ($\forall s = 1, 2, ..., |\mathcal{S}| - 1$) for given protection level k and disruption intensity level m. The probability that a facility arrives in capacity state $|\mathcal{S}|$ (the last class) is computed from Eq. (3.32).

We fit a MLR model on the synthetic data that contain the protection amounts, disruption intensity levels, and the corresponding capacity states of the facilities. The MLR model builds a relationship between the protection amounts, disruption intensity levels, and capacity states of the facilities. We discretize the entire range of the protection amounts into

a number of discrete protection levels by dividing the range by the number of protection levels ($| \mathcal{K} |$) specified by the decision-maker and then take the upper bounds of the resulting ranges as the cost (C_k) of protection resources corresponding to the discrete protection levels.

We use the fitted MLR model to predict the probabilities, $\mathbb{P}_{mk}(s)$, corresponding to the capacity states for given protection (C_k) and disruption intensity levels (m). These probabilities $\mathbb{P}_{mk}(s)$ are used as ${\mathbb{P}^{\omega}}_{zs(z,\omega)m(z,\omega)k}$ in the stochastic programming model, where ${\mathbb{P}^{\omega}}_{z0(z,\omega)1(z,\omega)k}$ is the probability that facility z is in capacity state 0 in scenario ω given that C_k amount (U.S. dollar) or k level of protection is invested to facility z and the facility is exposed to disruption intensity level 1.

Discriminant Analysis

Discriminant analysis is a popular method for multi-class classification. In discriminant analysis, the distribution of the predictors are modeled separately for each of the response classes and Bayes' theorem is used to sort the observations into different classes [63]. Two popular discriminant analysis methods are linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA). Both LDA and QDA assume that the predictor variables (X)—protection investments, and disruption intensity levels—in each class are drawn from a multivariate Gaussian distribution.

Linear Discriminant Analysis (LDA)

It is assumed in LDA that each class has a mean vector (μ_s) and a common covariance matrix (Σ) for all the $|\mathcal{S}|$ classes. The prior probability that a randomly selected observation is from class s is denoted by π_s. The multivariate Gaussian density function for an observation (x^{obs}) that comes from class s is defined as follows:

$$f_s\left(X = x^{obs}\right) = \frac{1}{(2\pi)^{pr/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}\left(x^{obs} - \mu_s\right)^T \Sigma^{-1}\left(x^{obs} - \mu_s\right)\right) \tag{3.33}$$

The probability that an observation is in class s, given the predictor X, or equivalently, a facility is in capacity state s for given protection and disruption intensity levels can be obtained from the following formula:

$$\mathbb{P}(\text{capacity state} = s | X = x^{obs}) = \frac{\pi_s f_s(x^{obs})}{\sum_{s=1}^{|\mathcal{S}|} \pi_s f_s(x^{obs})} \tag{3.34}$$

Using the estimates of prior probabilities, mean vectors, covariance matrix, and density function, we can compute the probabilities of the capacity states from LDA (Eq. 3.34) that are used in the mathematical model.

Quadratic Discriminant Analysis (QDA)

Unlike LDA, in QDA each class has its own covariance matrix Σ_s, which leads to quadratic decision boundaries [63]. The multivariate Gaussian density function for an observation (x^{obs}) that comes from class s is defined as follows:

$$f_s\left(X = x^{obs}\right) = \frac{1}{(2\pi)^{pr/2}|\Sigma_s|^{1/2}} exp\left(-\frac{1}{2}\left(x^{obs} - \mu_s\right)^T \Sigma_s^{-1}\left(x^{obs} - \mu_s\right)\right) \tag{3.35}$$

Using the estimates of prior probabilities, mean vectors, covariance matrix for each class, and density function, we can compute the probabilities from QDA using Eq. (3.34).

3.5 Computational Results and Managerial Insights

In this section, we evaluate the performance of the stochastic programming model in minimizing the expected PDTC by optimally investing the limited protection resources among the facilities and constructing links in the network. Our main focus is to present the numerical results that provide insights into modeling imperfect, multi-level protection and disruption, as well as multiple capacity states of the facilities in a combined network design and facility protection problem. Also, we conduct numerical experiments to explore the findings from our stochastic programming problem with both exogenous and endogenous uncertainty. The numerical results provide insights into the following research questions: (1) how does the decision maker's budget affect the expected PDTC, (2) how does the number of protection levels and the number of capacity states of the facilities used in the model affect the expected PDTC, (3) how sensitive is the stochastic programming model to changes in

the number of protection levels and the number of capacity states of the facilities used in the model, (4) how robust is the stochastic programming solution compared to the solution from the deterministic model, and (5) how sensitive is the stochastic programming model to the estimation error in the probabilities resulting from using different predictive models?

3.5.1 Experimental Setup

We implemented the basic and the accelerated L-shaped decomposition algorithms in Python 2.7 with Gurobi solver [48], using Gurobi to solve the mixed-integer master problem and the linear sub-problems. The prediction models estimating the probabilities, $\mathbb{P}_{zs(z,\omega)m(z,\omega)k}$, were implemented using Python's Scikit-learn package. We conducted numerical experiments on networks that were generated based on the southeastern United States (according to American Association of Geographers), following the similar procedure as Daskin [32]. We consider cities in the southeastern United States as nodes in the networks. Based on a threshold population, we generate different-sized networks. For example, considering cities with at least 185,000 population results in a 32-node network. Demand of each node is set to the population of the corresponding city divided by 10^4. In adding links between nodes, we consider 5 neighbors of each node that are within 250 miles, and the links do not intersect each other. Transportation cost of per unit flow is taken as the length of each link. Link construction cost C_ℓ is proportional to the length (L_ℓ) of each link, $C_\ell = uL_\ell$.

As we assume that the facilities are already located in the network, we solve the capacitated facility location and network design model of Melkote and Daskin [95] for the generated networks to find the optimal location of the facilities. In solving the Melkote and Daskin [95] model, we use the following parameters: median home value (obtained from Zillow.com, 2019) in each city as the possible facility setup cost at each node and a certain percentage of the total demand as the possible capacity of the facilities. The networks with the existing facilities are used in the computational experiments of this research. The 16-node, 25-node, and 32-node networks used in this research are demonstrated in Appendices D.1, D.2, and D.3, respectively. Figures D.2, D.3, and D.4 in Appendix D.1 show the changes in the optimal solution as the decision maker's budget changes for a 16-node network. In the numerical experiments, we used the parameter values as shown in Table 3.6.

Table 3.6: Parameters and their values used in the computational experiments.

Parameter	**Values Used**
Network size (Number of nodes,$\|\mathcal{N}\|$)	16, 25, 32
Number of facilities, $\|\mathcal{N}_{\mathcal{F}}\|$	4, 5, 6
Number of protection levels, $\|\mathcal{K}\|$	2, 3, 4, 5
Number of capacity states, $\|\mathcal{S}\|$	2, 3, 4
Number of disruption intensity levels	3
Link construction cost per unit length, u	\$10
Penalty of unmet demand	10
Budget, B(low, high)	(\$20,000, \$120,000)

In this research, we generate synthetic datasets of the capacity states used for the prediction models. We randomly generate the protection investment amounts within a range from a uniform distribution, $U(\$0, \$15000)$. Also, we randomly generate the disruption intensity levels in each observation of the dataset for a given number of disruption intensity levels. For each observation in the synthetic data, we use the binomial distribution, $binom(|\mathcal{S}|, p_{mk})$ to compute the capacity state for a given protection and disruption intensity level. Here, $p_{mk} = \left(\frac{allocation\,amount}{max.\,allocation\,amount}\right)^{\frac{m}{|\mathcal{M}|-1}}$ is the probability that a binomial trial is a success. The capacity amounts of the facilities corresponding to the discrete capacity states are computed using the formula, $a_{zs} = \frac{s}{|\mathcal{S}|-1} a_z$.

Modeling Disruption Events

We model the random disruptions using the disruption events as shown in Table 3.7 which also includes a "No disruption" event with a probability of 0.75. We consider actual geographic locations of the disruption events in the southeastern United States defined by latitude and longitude. In each scenario of our stochastic programming model, one of the disruption events is realized. Depending on the distance of a facility from the geographic center of a disruption, the facility is exposed to one of the three levels: low, high, and no intensity. All facilities within the radius 1 of a disruption are exposed to high intensity or level 2 intensity. The facilities outside radius 1 but within radius 2 are in low intensity or level 1 intensity. All other facilities outside radius 2 are exposed to level 0 intensity and are not affected by the given disruption.

3.5.2 Runtime of Algorithms

Though the primary goal of this chapter is to provide insights to the problem, we also evaluate the performance of the accelerated L-shaped decomposition algorithm. This section presents the runtime of the accelerated L-shaped decomposition algorithm and a comparison of the runtime of the basic L-shaped algorithm and the acceleration techniques. To compute the runtime of the algorithms, we ran experiments on a personal computer running MacOS with Intel Core i9 @2.9 GHz with 16GB of installed RAM.

Table 3.7: Disruption events.

Index	Type	Center	Lat	Lon	Radius1 (miles)	Radius2 (miles)	Prob.
0	No disruption	-	-	-	-	-	0.75
1	Hurricane	Tampa, FL	27.95	-82.46	150	500	0.10
2	Snow storm	Raleigh, NC	35.78	-78.64	100	500	0.05
3	Tornado	Huntsville, AL	34.73	-86.59	50	250	0.10

We observed from the computational experiments that the basic L-shaped algorithm is slow in convergence. To enhance the computational efficiency of the basic (*Single-Cut*) L-shaped decomposition algorithm, we implemented several acceleration techniques described in Section 3.4.2. We conducted experiments to assess the effects of the acceleration techniques in speeding up the basic algorithm. The runtime of the basic algorithm and different combinations of the acceleration techniques for two different networks are presented in Table 3.8. The acceleration techniques are denoted as follows: *KI*- knapsack inequality, *nLimit*- node limit in the master problem solver, *TR*- trust region, *MCut*- multi-cut L-shaped decomposition algorithm, *PCut*- Pareto-optimality cut, *VI*- valid inequality. In the runtime experiments, we allowed the algorithms to run for 2 hours. The "Time", "Gap", and "Iteration" columns represent the runtime of the algorithms, optimality gap, and number of iterations at termination, respectively.

We see from Table 3.8 that the *Single-Cut* algorithm cannot solve both of the network instances to optimality within 2 hours. When applied to the single-cut algorithm individually, the acceleration techniques, *KI*, and *TR* do not enhance the algorithm's speed*; VI* and *nLimit* methods speed up the algorithm. Especially, the *VI* significantly speeds up the algorithm. The *MCut* L-shaped algorithm substantially outperforms the *Single-Cut* algorithm in runtime. Though the *MCut* algorithm cannot solve both the network instances within 2 hours, the optimality gap of *MCut* at termination is much smaller than the

optimality gap of the single-cut algorithm. Unlike *Single-Cut,* in *MCut,* we add an optimality cut for each scenario to the master problem that results in adding a set of $|\Omega|$ cuts at each iteration. In our *MCut* L-shaped algorithm, we observed that the cut coefficients are different in different scenarios, resulting the right hand sides of the cuts (3.20a) to be different. This forces the lower bound of the optimal objective value to increase in a faster rate than the *Single-Cut* version, eventually faster the convergence. When *TR* is applied with the *MCut*, it slightly improves the computational efficiency of the algorithm. The *VI* and *nLimit* methods in conjunction with the *MCut* demonstrate much better performance in speed up compared to using these methods with *Single-Cut*. The *PCut* method in conjunction with the *MCut* provides small improvements in algorithm runtime. From Table 3.8, we see that the combination of *MCut, TR, nLimit, VI,* and *PCut* methods provide the best enhancement

Table 3.8: Comparison of acceleration techniques. Other parameters: $B = \$60,000, |\mathcal{S}| = 3, |\mathcal{K}| = 3$.

Acceleration Technique	$\|\mathcal{N}\| = 25, \|\mathcal{N}_\mathcal{F}\| = 4$			$\|\mathcal{N}\| = 25, \|\mathcal{N}_\mathcal{F}\| = 5$		
	Time	**Gap(%)**	**Iteration**	**Time**	**Gap(%)**	**Iteration**
Single-Cut	>7200	100	171	>7200	100	69
Single-Cut + KI	>7200	100	183	>7200	100	63
Single-Cut + nLimit	>7200	15.55	119	>7200	100	67
Single-Cut + TR	>7200	100	247	>7200	100	64
Single-Cut + VI	6225.45	0	200	>7200	43.97	69
Single-Cut + TR + KI + nLimit + VI	1368.09	0	45	6826.57	0	64
MCut	>7200	24.59	109	>7200	98.02	47
MCut + KI	>7200	33.23	112	>7200	99.03	43
MCut + nLimit	2948.97	0	65	>7200	38.84	49
MCut + TR	>7200	19.90	111	>7200	97.49	52
MCut + VI	3124.89	0	49	>7200	33.49	42
MCut + PCut	>7200	22.94	107	>7200	97.74	46
MCut + TR + KI + nLimit + VI + PCut	495.57	0	14	2875.62	0	20
MCut + TR + nLimit + VI+ PCut	421.71	0	14	1882.78	0	14

in algorithm runtime. Therefore, we used this combination in all of the experiments unless otherwise specified.

We conducted computational experiments with the three different network sizes—16, 25, and 32-node networks—where the number of facilities ranges from 4 to 6. In the experiments, we vary the decision maker's budget, number of protection levels, and number of capacity states of facilities into four, four, and three levels, respectively. The runtimes of the accelerated L-shaped decomposition algorithm ($MCut + TR + nLimit + VI + PCut$) are demonstrated in Figures 3.1, and 3.2 and Table E.12 (Appendix E).

Figure 3.1 demonstrates the effect of the number of facilities and the decision maker's budget on algorithm runtime for the three different network sizes. We see that the runtime increases with the network size. Algorithm runtime increases as the decision maker's budget increases from \$30,000 to \$70,000 and then decreases at \$90,000 for all the networks. With more budget, the decision-maker can allocate higher level of protection resources and construct more links in the transportation network. This requires the decision-maker to explore more combinations of protection levels and facilities as well as more candidate links which eventually results in an increased runtime. However, with a sufficiently large budget, decision making is relatively easy. Because, at that point, the decision-maker can just allocate the highest level of protection and construct most of the links in the network that reduces the necessity of exploring a large number of candidate solutions. We see that at each budget level, runtime increases exponentially as the number of facility in the networks increases. Increasing the number of facilities in the transportation network increases the binary variables and the number of scenarios in the stochastic programming model that leads to a substantially higher runtime.

We see from Figure 3.2a that the algorithm runtime increases as the number of protection levels used in the model increases. Increasing the number of protection levels adds more binary variables and constraints in the stochastic mixed-integer programming model that has an exponential runtime. Therefore, runtime increases as the number of protection levels modeled increases for all three network sizes.

Figure 3.2b demonstrates that the algorithm runtime increases substantially as the number of possible PDCSs of each facility increases. Increasing the number of possible

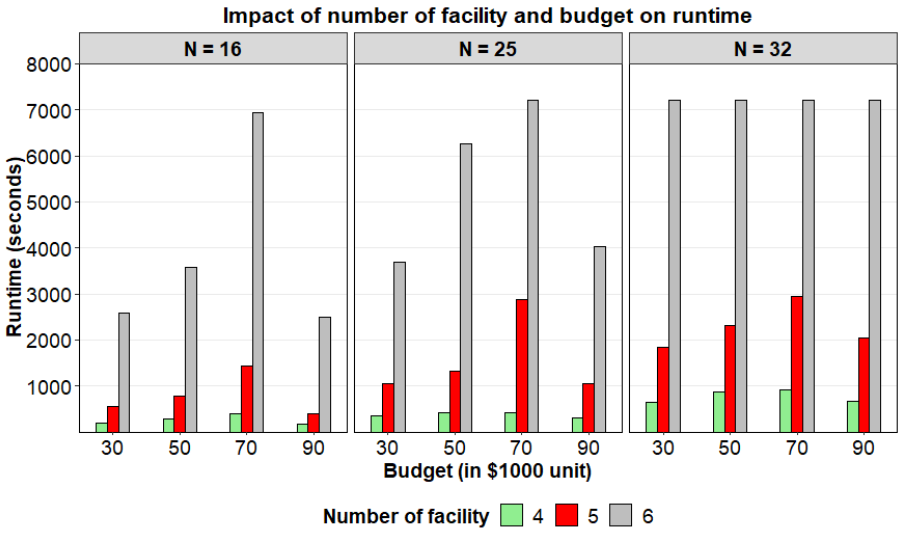
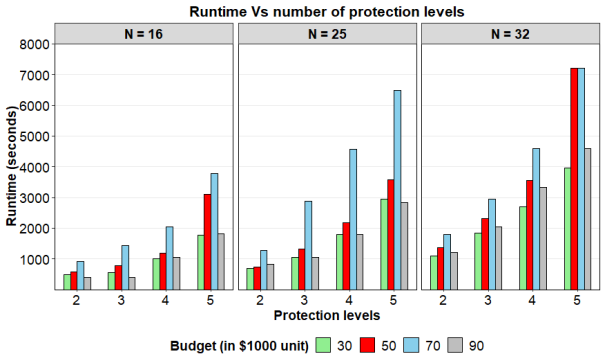
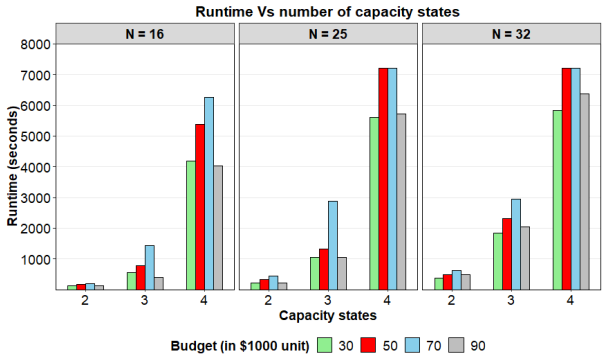


Figure 3.1: Impact of decision maker's budget and number of facility on runtime. Other parameters: $|\mathcal{S}| = 3, |\mathcal{K}| = 3$.



(a) Runtime Vs $|\mathcal{K}|$. $|\mathcal{N}_{\mathcal{F}}| = 5, |\mathcal{S}| = 3$.



(b) Runtime Vs $|\mathcal{S}|$. $|\mathcal{N}_{\mathcal{F}}| = 5, |\mathcal{K}| = 3$

Figure 3.2: Impact of number of protection levels and capacity states on runtime.

PDCSs of each facility significantly increases the number of scenarios of the stochastic programming model. This increased number of scenarios in the stochastic programming model contribute to increase the algorithm runtime. We see from Figure 3.2b and Table E.12 (Appendix E) that in 25- and 32-node networks, as the number of scenarios becomes large (≥ 3000), the L-shaped algorithm cannot solve the problem instances within 2 hours.

3.5.3 Effect of Budget on Post-disruption Transportation Cost

We conducted experiments to assess the effect of the decision maker's budget on the expected PDTC in a network. Figure 3.3 demonstrates the variation of the expected PDTC with the decision maker's budget for three different networks, each with 25- nodes as well as the average expected PDTC of the three networks.

We see that the expected PDTC decreases as the decision maker's budget increases, which is very intuitive. With increased budget, the decision maker can open more links in the network and allocate more protection resources to the facilities, increasing the likelihood of higher PDCS of the facilities. With higher post-disruption capacity of the facilities, the decision-maker can better satisfy the demands of the customer locations and thereby avoid the high penalty cost of unmet demand, which eventually reduces the transportation cost.

However, the expected PDTC does not decrease uniformly as the budget increases. Initially there is no existing link in the network, and no protection resource is allocated to the facilities. At this point, with an additional budget increment, the link construction and protection investment can yield a substantially higher rate of reduction in the expected PDTC. However, at a certain point in the budget, the rate of decrease in the expected PDTC diminishes. At this point, the network has some opened links, and the facilities are protected to some extent. Therefore, with a small budget increment, opening a few more links or allocating some protection resources cannot reduce the expected PDTC at a higher rate.

Transportation network managers can use our model along with their own datasets to produce a figure like Figure 3.3 to decide on the amount to invest in network construction and facility protection. For example, for network 1, a manager would be well-advised to

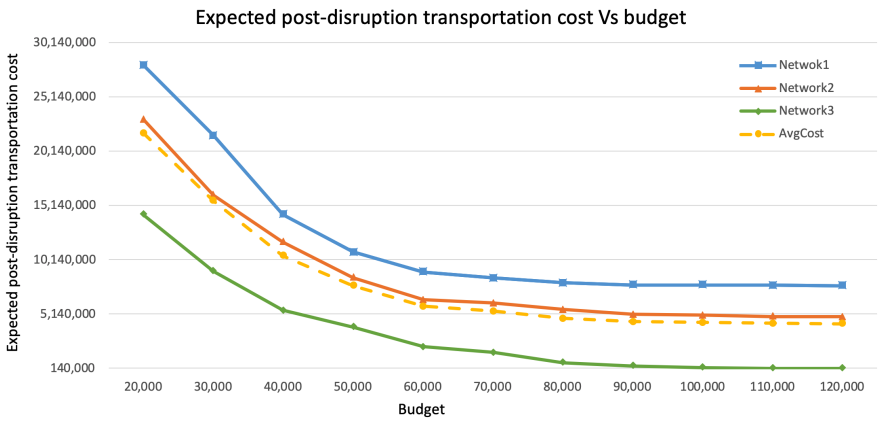


Figure 3.3: Variation of expected post-disruption transportation cost with decision maker's budget for three 25-node networks.

choose to invest $80,000, as the expected PDTC variation curve of this network levels off after this point.

3.5.4 Significance of Modeling Multiple Protection Investment Levels

Our stochastic programming model allows network owners to change the number of protection investment levels, $|\mathcal{K}|$, to exploit the model fidelity on the solution. To demonstrate the effects of changing the number of protection levels, we introduce a metric called *relative difference*. The relative difference for protection levels measures the percentage by which the expected PDTC changes due to using a different number of protection levels $|\mathcal{K}|$ rather than a reference number of protection levels $|\mathcal{K}^*|$. The relative difference can be expressed as:

$$RD^k = \frac{Q(|\mathcal{K}^*|) - Q(|\mathcal{K}|)}{Q(|\mathcal{K}^*|)} \times 100\% \tag{3.36}$$

Here, $Q(|\mathcal{K}^*|)$ and $Q(|\mathcal{K}|)$ represent the expected PDTC corresponding to using $|\mathcal{K}^*|$ and $|\mathcal{K}|$ number of protection levels in the model. A positive value of RD^k demonstrates the percentage reduction of the expected PDTC due to changing the number of protection levels from $|\mathcal{K}^*|$ to $|\mathcal{K}|$. Therefore, a positive value of RD^k is a measure of improvement in the decision due to increasing the number of protection levels from $|\mathcal{K}^*|$ to $|\mathcal{K}|$. Table 3.9 shows the relative difference in the expected PDTC due to increasing the number of protection levels from the reference number $|\mathcal{K}^*|$ at different decision maker's budgets. The experiments in this and the following sub-sections were carried out with 16-node networks having 4 facilities and M-Cut L-shaped algorithm with TR cuts, Pareto-optimality cuts, and valid inequality.

We see from Table 3.9 that RD^k increases and thus the expected PDTC decreases at different budget levels as the number of protection levels increases from $|\mathcal{K}^*|$ to $|\mathcal{K}|$. With a larger number of protection levels, the costs of protection resources corresponding to the discrete protection levels are more refined. This enables the model to allocate more accurate

Table 3.9: Relative difference for changing the number of protection levels from $|\mathcal{K}^*| = 2$.

$\lvert\mathcal{K}\rvert$	**Budget**				**Average**
	$40,000	**$60,000**	**$80,000**	**$100,000**	
2	0%	0%	0%	0%	0%
3	16.10%	21.21%	8.71%	0.56%	11.65%
4	21.59%	26.75%	14.51%	5.05%	16.98%
5	22.31%	27.69%	16.88%	7.95%	18.71%

levels of protection resources to the facilities. As a result, with the same amount of budget, a more refined and accurate protection allocation can be chosen, reducing the expected PDTC.

However, we see that the rate of percentage reduction of the expected PDTC diminishes as the number of protection levels $|\mathcal{K}|$ increases. Table 3.9 shows that the percentage reduction in the expected PDTC diminishes considerably after the number of protection levels, $|\mathcal{K}| = 4$ and becomes only $1.73\%\,(= 18.71 - 16.98)$ between $|\mathcal{K}| = 4$ and $|\mathcal{K}| = 5$. As increasing the number of protection levels beyond a sufficiently large value provides very small improvement in reducing the expected PDTC while increasing the computational complexity and runtime substantially (Figure 3.2a), it is worth considering a trade-off between the improvement in solution and runtime. A sufficiently large value of $|\mathcal{K}|$, for instance $|\mathcal{K}| = 4$ provides sufficient improvement in cost reduction while resulting in a reasonable runtime.

Furthermore, we observe that the effect of changing the number of protection levels on expected PDTC changes as the decision maker's budget changes. Table 3.9 reveals that the percentage reduction in the expected PDTC due to changing $|\mathcal{K}|$ increases as budget increases from \$40,000 to \$60,000, and then decreases until \$100,000 where the percentage reduction is the lowest. This low percentage reduction is because with large budgets, it is possible for some facilities to receive the maximum protection amount, making the number of protection levels insignificant.

3.5.5 Significance of Modeling Multiple Capacity States

In our stochastic programming model, the network owners can change the number of PDCSs, $|\mathcal{S}|$, of the facilities. To investigate the sensitivity of the model to changes in $|\mathcal{S}|$, we compute the relative difference for the number of PDCSs (RD^s). The metric RD^s is expressed as:

$$RD^s = \frac{Q_{s^*}(y(|\mathcal{S}|)) - Q_{s^*}(y(|\mathcal{S}^*|))}{Q_{s^*}(y(|\mathcal{S}^*|))} \tag{3.37}$$

Here, $y(|\mathcal{S}|)$ and $y(|\mathcal{S}^*|)$ are the optimal solutions obtained from using $|\mathcal{S}|$ and $|\mathcal{S}^*|$ number of PDCSs, respectively. $Q_{s^*}(y(|\mathcal{S}|))$ is the objective value that we get by evaluating the solution $y(|\mathcal{S}|)$ using $|\mathcal{S}^*|$ PDCSs. The metric RD^s represents the percentage change

in the expected PDTC due to using a different number of PDCSs than $|\mathcal{S}^*|$. A positive value of RD^s provides a measure of the cost of using a smaller number of PDCSs $|\mathcal{S}|$ from $|\mathcal{S}^*|$. Table 3.10 shows that the RD^s increases as the number of PDCSs decreases from $|\mathcal{S}^*|$, meaning that using a smaller number of PDCSs leads to a decision that results a higher expected PDTC. As the number of PDCSs increases, the capacity amounts corresponding to the discrete capacity states become more refined. Therefore, the greater the number of PDCSs, the more refined is the model and produces better quality solutions.

However, unlike the effect of the number of protection levels, we see that the model is not substantially sensitive to changes in the number of PDCSs. The effects of multiple PDCSs in this combined NDIP problem are consistent with the findings of Medal et al. [94] who modeled the infrastructure protection with endogenous uncertainty only. We see from Table 3.10 that the percentage increase in the expected PDTC due to using $|\mathcal{S}| = 3$ rather than $|\mathcal{S}^*| = 4$ is very small. Decreasing the number of PDCSs from $|\mathcal{S}^*| = 4$ to $|\mathcal{S}| = 2$ demonstrates some noticeable increment in RD^s.

However, increasing the number of PDCSs in the stochastic programming model increases the algorithm complexity and runtime substantially (Figure 3.2b). Therefore, the decision maker can consider a trade-off between model accuracy and runtime and can choose a value of $|\mathcal{S}|$ that provides sufficiently high accuracy and reasonable runtime. Based on the numerical results, using $|\mathcal{S}| = 3$ provides satisfactory model accuracy and a reasonable runtime.

3.5.6 Significance of Using a Stochastic Model

We analyze the benefit of modeling uncertainty in the PDCSs of the facilities. To accomplish this finding, we formulate a *mean value problem* (MVP) and compare the performance of the solutions obtained from the MVP with the solutions obtained from our stochastic programming model. The formulation of the MVP of our PNDIPP is presented in Appendix B. This model uses the expected capacity amount of each facility over all the scenarios of the stochastic network design and facility protection problem.

A metric called *value of stochastic solution* (VSS) [21] measures the cost of ignoring uncertainty in the PDCSs of the facilities when generating link construction and protection decision. The VSS can be expressed as follows:

Table 3.10: Relative difference for changing the number of capacity levels from $|\mathcal{S}^*| = 4$.

$\lvert\mathcal{S}\rvert$	**Budget**				**Average**
	$40,000	**$60,000**	**$80,000**	**$100,000**	
4	0%	0%	0%	0%	0%
3	0.28%	0.43%	0%	0.07%	0.20%
2	2.05%	0.46%	3.72%	18.55%	6.19%

$$VSS = \frac{Q(\mathbf{y}^{MVP}) - Q(\mathbf{y}^{SP})}{Q(\mathbf{y}^{SP})} \times 100\% \tag{3.38}$$

Here, $Q(\mathbf{y}^{MVP})$ and $Q(\mathbf{y}^{SP})$ are the expected PDTCs resulting from using the optimal solutions obtained from MVP and stochastic programming model, respectively. Tables 3.11 and 3.12 demonstrate the variation of VSS as the number of protection levels and PDCSs in the model changes for different budget levels. We see that VSS increases substantially as the decision maker's budget increases within the same level of $|\mathcal{K}|$ and $|\mathcal{S}|$. With more budget, the decision-maker can construct more links and allocate higher level of protection resources to the facilities. However, this can lead to more erroneous decisions of link construction and protection investment by the MVP. Therefore, while the stochastic programming model can generate better solutions with the additional budget, the MVP model generates poorer quality solutions, which eventually results in a larger VSS. In contrast, with a sufficiently small budget (e.g., \$20,000), the VSS is consistently zero over different levels of $|\mathcal{K}|$ and $|\mathcal{S}|$. Because, with a very small budget, the decision-maker can only construct few links and make small investment in facility protection. This restricts the scope of the stochastic programming model to generate a better solution than the MVP. Table 3.11 reveals that at higher budget levels, the VSS increases as number of protection levels increases within the same budget level.

Table 3.12 demonstrates that the VSS increases as the number of PDCSs of the facilities increases within the same budget level; at the higher budget levels, the VSS increases substantially. Increasing the number of possible PDCSs means increasing the range of uncertainty. As the stochastic programming model accounts for the uncertainty in the PDCS, the solution obtained from stochastic programming model is robust against uncertainty. On the other hand, the MVP considers only the expected value of the post-disruption capacity of each facility. Therefore, as the uncertainty in post-disruption capacity increases, the solution of the MVP becomes more erroneous, leading to a very large VSS.

In our combined NDIP problem with both exogenous and endogenous uncertainty, the VSS is substantially different from the results of Medal et al. [94], who modeled a facility

Table 3.11: Variation of VSS with number of protection levels.

$\|\mathcal{K}\|$	**Budget**				**Average**
	$20,000	**$40,000**	**$60,000**	**$80,000**	
2	0%	116.51%	263.25%	720.66%	275.10%
3	0%	160.81%	361.03%	754.58%	319.11%
4	0%	179.09%	395.36%	829.87%	351.08%

Table 3.12: Variation of VSS with number of capacity states

$\|\mathcal{S}\|$	**Budget**				**Average**
	$20,000	**$40,000**	**$60,000**	**$80,000**	
2	0%	116.51%	263.25%	720.66%	275.10%
3	0%	174.22%	414.29%	1326.19%	478.68%
4	0%	215.61%	430.84%	1901.79%	637.06%

fortification problem with endogenous uncertainty only. In Medal et al. [94], the mean value model performs very well compared to the stochastic programming model, which leads to a very small VSS value.

3.5.7 Estimation Error of Predictive Modeling Techniques

We stated earlier that we computed the capacity states of the facilities using a binomial distribution when generating the synthetic input dataset. We fitted the predictive modeling techniques using the synthetic dataset to estimate the probabilities of PDCSs of the facilities for given protection amounts and disruption intensity levels. In the computational experiments, we used the multinomial logistic regression (MLR) to estimate the probabilities of PDCSs. We compare the effects of the estimation error in the probabilities on the optimal objective value resulting from MLR with other classification techniques—LDA and QDA. Another research question we want to explore: how much is the effect of the estimation error on the optimal solution and optimal objective value due to using the predictive modeling techniques compared to using the original binomial distribution? To answer this question, we conducted numerical experiments to measure the percentage change in the expected PDTC due to using the predictive modeling techniques versus the true binomial distribution. We ran the experiments for different levels of defender's budget, number of protection levels, and number of PDCSs.

Table 3.13 demonstrates the percentage change in the expected PDTC due to the estimation error from MLR, LDA, and QDA. To avoid the effect of randomness of the synthetic data generation, we conducted the experiments with three different replications of the synthetic dataset and took their average. We see from Table 3.13 that the effect of estimation error on the objective value is smallest in QDA; the average $\%Higher$ over different budget, protection, and capacity levels from MLR, LDA, and QDA are 6.34, 6.38, and 3.12, respectively. However, in few instances, the objective value obtained from using QDA is higher than the objective value obtained from using MLR and LDA. In MLR and LDA, the decision boundaries are linear, whereas QDA can construct quadratic decision boundaries. Therefore, in the classification problems where the actual class boundaries are non-linear, QDA outperforms the MLR and LDA. Decision-makers need to be careful in selecting a

Table 3.13: Effect of estimation error on the optimal objective value. "% Higher" represents the percentage by which the objective value is higher due to using the predictive modeling techniques rather than using the ground-truth binomial probabilities.

B	$\lvert\mathcal{S}\rvert$	$\lvert\mathcal{K}\rvert$	% Higher		
			MLR	**LDA**	**QDA**
40000	2	2	1.53	0	0
		3	9.42	10.06	2.58
		4	3.42	3.42	6.17
	3	2	0.25	0.25	0
		3	16.34	16.59	2.62
		4	3.63	3.72	3.53
	4	2	0.22	2.58	0
		3	15.10	15.63	0.02
		4	0	2.42	0.64
60000	2	2	6.13	9.89	7.68
		3	11.73	15.19	0
		4	11.32	13.77	4.79
	3	2	12.01	14.72	14.72
		3	4.92	4.92	11.48
		4	12.24	13.99	12.27
	4	2	0	2.51	0.28
		3	0	0	0
		4	0	0	0
80000	2	2	1.40	1.70	0
		3	13.74	1.90	3.02
		4	14.49	15.44	8.48
	3	2	1.19	1.96	1.16
		3	8.71	8.72	4.38
		4	3.93	0	0
	4	2	0	2.41	0
		3	9.50	8.81	0.43
		4	9.99	1.56	0
Average			6.34	6.38	3.12

predictive modeling technique to estimate the parameters of a stochastic programming model with both endogenous and exogenous uncertainty, and QDA appears to be a good choice.

3.6 Conclusions

We studied an integrated network design and infrastructure protection (NDIP) problem where the effect of protection on facilities subject to random disruptions are uncertain, imperfect, and multi-level. In this problem, the decision-maker, with a limited budget, seeks to optimally construct links in a network of facilities and demand locations and allocate protection resources to the facilities before the disruptions are realized so that customer demands are satisfied with minimal expected PDTC after disruption. We assume that the post-disruption capacity state (PDCS) of a facility is unknown to the decision-maker for a given protection. However, the PDCS of a facility depends probabilistically on the protection and the disruption intensity level. We implemented multiple predictive modeling techniques that can analyze historical data to estimate the probability of a facility's PDCS for given protection and disruption intensity level.

We formulated the problem as a two-stage stochastic programming model with both exogenous and endogenous uncertainty. The protection decision affects the scenario probabilities, making the uncertainty endogenous to the model, whereas the link construction decision directly affects the recourse decision as in exogenous uncertainty. We implemented an L-shaped decomposition algorithm with multiple acceleration techniques to solve the model. Computational experiments show that the multi-cut L-shaped decomposition algorithm with trust region cut, Pareto-optimality cut, master problem node limit, and a valid inequality provides the best runtime. The algorithm runtime increases as the number of facilities, number of protection levels, number of capacity states, and the decision maker's budget increases. The larger the budget of the decision-maker, the smaller is the expected PDTC.

The numerical results demonstrate that the stochastic NDIP model is sensitive to the number of protection levels used in the model. We see that the expected PDTC decreases as the number of protection levels used in the model increases, while increasing the

computational complexity and algorithm runtime substantially as the number of protection levels becomes sufficiently large. Therefore, practitioners should use a moderately large number of protection levels such as $|\mathcal{K}| = 4$ that provides sufficiently high model accuracy and a reasonable runtime. Also, as the decision maker's budget becomes sufficiently large, the effect of increasing $|\mathcal{K}|$ becomes small. Unlike the effect of $|\mathcal{K}|$, the model is not substantially sensitive to changes in the number of PDCSs of the facilities. The expected PDTC increases by a small amount as the number of PDCSs used in the model decreases from a reference number of PDCSs. However, increasing the number of PDCSs substantially increases the algorithm runtime. Therefore, we recommend that transportation network managers use about $|\mathcal{S}| = 3$ capacity levels, depending on the application.

The *value of stochastic solution* (VSS) metric reveals that the mean value model performs very poorly compared to the stochastic programming model. The VSS increases as the number of protection levels, PDCSs, and the decision maker's budget increases. Numerical results demonstrate that the estimation error of the probabilities of PDCSs from using the predictive modeling techniques affects the optimal solution and the optimal objective value of the stochastic programming model. We see that the average effect of estimation error on the expected PDTC is lowest for quadratic discriminant analysis. Therefore, network managers need to be careful in using prediction models to estimate the probabilities. A prediction model capable of providing better estimation when the dataset have complex class boundaries may be a good choice.

3.6.1 Future Work

In this study, we see that the runtime of the accelerated L-shaped decomposition is high when there are a very large number of scenarios. Thus, a future extension of this research is to implement a sample average approximation algorithm to solve the problem with large number of scenarios.

We assume that the PDCSs of a facility are independent of another facility. However, failure of facilities can be interdependent, especially in power systems, where failure of a substation or a transmission line can cause the failure of another. A possible future extension of our model could be relaxing this assumption to incorporate the interdependence among

the facilities in the network. Similarly, the disruption events in a small geographic location can be related to each other. Another possible extension of this research would be to account for this possible correlation among the disruption events.

A limiting assumption of this research is that the links are reliable when subject to disruptions. However, link capacities are often degraded under major disruptions. This research can be further extended to incorporate the possibility of link failures.

Chapter 4

Submodular Optimization Under Endogenous Uncertainty: An Application to Complex System Reliability

4.1 Introduction

This chapter studies continuous submodular optimization under endogenous uncertainty with an application to complex system reliability allocation. Specifically, we study the submodularity of the SPEU framework's objective function for some probability distributions that have not been explored in the SPEU literature to date. We study a special case of a complex system reliability allocation problem, where the system is submodular. A system is submodular if that demonstrates the diminishing return property. A submodular system's performance improves more for an additional component's availability (survivability) if a fewer number of components are available so far than if larger number of components are already available [72]. Therefore, as the more number of components survive for a specified duration, the less improvement in performance can be obtained from an additional component's availability. In this problem, a decision-maker seeks to optimally invest

resources among the system's components to maximize the probability that the system maintains a specified performance for a time horizon. We assume that the decision-maker does not know whether a component of the system survives for the specified time horizon or not. However, to be realistic, we assume that the likelihood of a component's survival increases as the decision-maker invests more resources on that component, making the component's lifetime a probabilistic function of the investment decisions. We model the complex system reliability allocation problem as a stochastic program with endogenous uncertainty to incorporate these assumptions. This research is the first to study the reliability allocation of a complex submodular system and model it using the SPEU framework, where the uncertainty in a component lifetime is endogenous to the model.

Our goals in this research are to (1) analyze the submodularity of the SPEU framework's objective function for some probability distributions that have not been explored in the SPEU literature to date, (2) analyze the benefit of submodularity in solving large-scale complex system reliability allocation problem.

4.1.1 Motivation

Many real-life infrastructure systems such as water supply, oil/gas transportation, telecommunications, and power transmission networks are complex systems [68]. Complex systems refer to those networks where the components cannot be connected in a series-parallel way [77]. These networks need to design/protect to ensure that they can maintain a specified level of performance. The reliability allocation of complex systems has been studied much less than series-parallel systems, especially with uncertainty in component survivability. As modeling and solving the reliability allocation problems for complex systems using the exact mathematical optimization approach is computationally challenging, particularly for large-sized networks, studies often use heuristics and meta-heuristics algorithms for solving those problems. These heuristics and meta-heuristics algorithms often provide poor quality solutions. They cannot give any worst-case performance guarantee, which raises the necessity to design/implement approximation algorithms that can solve large-scale problems while providing a worst-case performance guarantee.

Submodularity property is useful in providing a performance guarantee of approximation algorithms. Though some research studied submodularity in the SPEU literature, they limit their analysis to one or two probability distributions having special properties (e.g., semigroup property). But, many other widely used probability distributions do not possess this particular property. Therefore, there is a need to conduct further analysis of the SPEU framework's submodularity for those probability distributions and explore the benefit of submodularity in application areas such as complex system reliability.

4.1.2 Related Literature

We provide a review of the literature related to our study from the perspective of complex system reliability allocation, continuous submodular optimization, and the submodularity of the SPEU framework in this sub-section.

The complex system reliability problem has not been studied as extensively as for purely series, purely parallel, and series-parallel systems. The reliability of complex systems can be enhanced by increasing the reliability of components consuming certain resources or adding redundancy to the components [77]. Most recent studies in the literature used heuristics or meta-heuristics methods to solve complex systems reliability optimization problems. However, early works on complex systems reliability literature used mathematical programming frameworks to model the complex systems reliability problem and developed/implemented exact solution approaches to solve the problems to optimality. Among the early studies, the Lagrangian relaxation approach was introduced to solve the nonlinear models of complex systems reliability allocation problems in different engineering applications [60] and a complex distributed system [115]. Kim and Frair [67] modeled a redundancy allocation problem in a complex system as a nonlinear model and then reformulated the model before solving it using a quasi-Newton method. Li et al. [77] studied a reliability maximization problem using redundancy for the components in a complex system, where the authors modeled the problem as a mixed-integer nonlinear program. The authors proposed an exact decomposition algorithm for solving the resulting model.

Despite the above-mentioned exact optimization approaches attempt to provide optimal solutions to the complex systems reliability optimization problems, their application often

limits the structure and size of the problem to be solved, eventually restricting the benefit of these approaches [31]. Therefore, recent studies focused on designing/implementing heuristics and meta-heuristics algorithms to solve large-sized problems in the complex systems reliability literature. Ravi et al. [121] studied a multi-objective complex system reliability optimization problem and implemented a variant of simulated annealing meta-heuristics—threshold accepting—to solve the problem. Among the metaheuristic methods, genetic algorithm (GA) has been used in solving the complex system reliability optimization problems. Kumar et al. [74] presented a multi-objective GA to optimize the reliability of a complex telecommunication network. Sheikhalishahi et al. [129] proposed a hybrid GA-particle swarm optimization (PSO) for solving a reliability-redundancy allocation problem for complex systems. Some studies implemented PSO in solving complex systems reliability optimization problems, such as PSO based on Monte Carlo simulation for reliability optimization [145] and PSO for redundancy allocation [107] of complex systems. Recently, Kumar et al. [73] proposed a new metaheuristics algorithm—gray wolf—for complex system reliability optimization.

Though it is essential to account for the underlying uncertainty, few studies have explicitly modeled uncertainty in complex system reliability optimization. Marseguerra et al. [89] and Marseguerra et al. [90] explicitly modeled the uncertainty in parameters—failure and repair rate of components—in reliability optimization problem of complex systems, where the authors used a multi-objective GA to solve their problem.

Though submodularity has been studied extensively for the discrete case in submodular optimization literature, continuous submodular optimization has also gained attention recently. Bian et al. [19] studied the submodularity of a continuous function and proposed a variant of conditional gradient method (a.k.a., Frank-Wolfe) that provides $(1 - 1/e) - \epsilon$ approximation guarantee in maximizing a monotone and diminishing return (DR) submodular function subject to down-closed convex constraints. Hassani et al. [51] studied a stochastic continuous submodular optimization problem that maximizes the objective function expressed as an expectation of submodular second-stage functions. The authors proposed a stochastic gradient method providing a worst-case performance guarantee for maximizing a monotone continuous DR-submodular function under general convex

constraint. Mokhtari et al. [103] is another study on maximizing a monotone and continuous DR-submodular function subject to general convex constraint that proposed a variant of the Frank-Wolfe algorithm. As discrete greedy algorithms usually fail to provide the tightest guarantees for many classes of feasibility constraints, many studies have used the continuous relaxations of submodular functions in discrete domains [103]. Studies obtained continuous relaxations through multilinear extension and proposed continuous greedy algorithms in maximizing monotone DR-submodular functions subject to matroid constraint with an approximation guarantee (e.g., [26, 27, 135, 141]).

In the SPEU literature, Karaesmen and Van Ryzin [65] analyzed the submodularity of the objective function of a two-stage stochastic program for an airline revenue management problem that seeks to determine the number of overbooking levels in different inventory classes. They modeled uncertainty using binomial distributions, where all the binomial random variates have the same probability of success. The authors proved that the expected value function (objective function) $f(\theta)$ is submodular in θ and componentwise concave in θ_i if the random variables θ_i have semigroup property—if two independent random variables Y_1 and Y_2 are stochastically equivalent to $x(\theta_1)$ and $x(\theta_2)$, respectively, then $Y_1 + Y_2$ is stochastically equivalent to $x(\theta_1 + \theta_2)$. Later, Medal et al. [94] studied a two-stage stochastic program with endogenous uncertainty similar in structure to the problem studied by Karaesmen and Van Ryzin [65] that uncertainty was modeled using independent binomial random variables. Unlike Karaesmen and Van Ryzin [65], in Medal et al. [94], submodularity was studied for binomial random variables having different probabilities of success. Also, Medal et al. [94] studied the problem in which the allocation of resources to the facilities was discrete.

4.1.3 Contributions

In summary of the existing literature, uncertainty in system parameters has not been considered much, and mostly heuristics and meta-heuristics methods lacking performance guarantee are used in complex systems reliability optimization problems. Moreover, the theoretical properties of the complex system reliability allocation problem have not been explored to date. But, the theoretical analysis provides a better understanding of the problem

structure that can facilitate solution approaches. Also, in SPEU literature, a few studies analyzed the submodularity for binomial distribution having semigroup property, whereas other widely used probability distributions such as exponential have not been investigated.

Therefore, this chapter studies a complex submodular systems reliability optimization problem under uncertainty in the component's lifetime. We present a two-stage SPEU model to maximize a complex system's reliability by allocating resources among the system components. This research extends the complex system reliability allocation and SPEU literature by establishing the mathematical formulation's theoretical properties that enable solving large-scale complex systems reliability problems and provides insights into the problems. Specifically, we made the following contributions: (1) proved that the reliability maximization objective function of the SPEU model is submodular when the lifetimes of the network components follow exponential distributions, (2) implemented a continuous greedy approximation algorithm capable of providing worst-case performance guarantee utilizing the submodularity and provided computational experiments to demonstrate the performance of the algorithm in solving large-scale reliability maximization problems, effects of model parameters on the runtime and solution quality of the algorithm.

4.2 Problem Description

In this problem, with a limited budget, a decision-maker seeks to optimally invest resources to the components of a complex submodular system to maximize reliability—the probability that the network provides a minimum required utility for a specified time horizon. A real-application of this type of system is sensor placement in a water supply network, where the goal is to maximize the probability that the network is capable of satisfying the demand of a certain amount of population for a specified time horizon. For instance, consider a water distribution network where sensors are placed to detect malicious contaminations and to ensure the supply of pure drinking water to an area for a specified time horizon. The utility (demand coverage) provided by this network demonstrates a diminishing return in the number of sensors' survival over the specified time horizon; the more sensors survive over the specified time horizon, the less demand coverage can be obtained from an additional sensor's

survival. This network structure is complex as it is not possible to represent the network as a series-parallel combination of the components.

As the network components are exposed to the natural environment, their performance degrades over time and eventually become inoperable. Decision-makers can invest their limited resources to reduce the rate of degradation and enhance the components' lifetime. However, the lifetime of the network components is usually uncertain as it is difficult to accurately estimate the effects of numerous environmental factors on their performance. To reflect this phenomenon, we assume that the decision-maker does not know whether a component survives for a given time horizon. Therefore, each component can arrive in either of the two possible (binary) states—survival or failure—after the specified time horizon.

To represent uncertainty in the components' states after a specified time horizon, we construct a set of scenarios. Each scenario represents a particular realization of all the components' survival/failure states. The total number of scenarios is equal to the number of possible combinations of the components' states. After realizing the survival/failure states of the components in a scenario, we can determine the utility provided by the network. We assume that components' lifetimes follow exponential distributions. To be realistic, we assume that if a decision-maker invests more resources on a component, such as installs an expensive but better quality sensor in the water supply network by incurring more cost, the survival probability of that component increases. Therefore, the survival probability of a component is a function of investment decisions, making the uncertainty in this problem to be decision-dependent (endogenous).

4.3 Mathematical Formulation

We formulate the reliability maximization problem as a two-stage stochastic program with endogenous uncertainty. The necessary sets, parameters, and variables supporting the mathematical formulations are listed in Table 4.1.

The first-stage model (4.2) seeks to maximize the probability that the minimum required utility η is maintained by the network for a time horizon T by optimally investing the limited budget B among the network components. We assume that each component's lifetime is

Table 4.1: Notation.

(a) Sets

Sets	Description
$\mathcal{L}$	Set of all components indexed by ℓ
Ω	Set of all scenarios indexed by ω
L^{ω}	Set of all components that survived in scenario ω

(b) Parameters

Parameters	Description
B	Decision maker's budget
η	Minimum required utility to be provided by the network
T	Specified time horizon

(c) Variables

Variables	Description
y_{ℓ}	Amount of investment on component ℓ
λ_{ℓ}	Failure rate of component ℓ

uncertain, making the survival/failure state of a component at the end of time horizon T uncertain. To represent uncertainty in the components' states, we construct a finite set of scenarios, where $\omega \in \Omega$ represents a particular realization of the components' survival/failure states. As each component's lifetime follows an exponential distribution, we can compute the probability that component ℓ survives the time horizon T using the survival function $e^{-\lambda_\ell T}$, where λ_ℓ is the failure rate. A key feature of this reliability maximization problem is that the survival probability of component ℓ is a function of the investment amount y_ℓ on that component. To represent this decision-dependent (endogenous) probability structure, we model the failure rate λ_ℓ of component ℓ to be a function of the investment amount y_ℓ, i.e., $\lambda_\ell = \frac{1}{y_\ell}$. Therefore, we can consider λ_ℓ to be a decision variable replacing y_ℓ. In this case, failure rates λ of the components affect the probability of scenario ω expressed as in Eq. (4.1).

$$\mathbb{P}^\omega(\boldsymbol{\lambda}) = \prod_{\ell \in L^\omega} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \tag{4.1}$$

Equation (4.1) shows that the probability of a scenario ω is a function of the failure rates $\boldsymbol{\lambda}$. The smaller the value of λ_ℓ for component ℓ, the larger the probability of survival. Therefore, the goal of our stochastic programming model is to find the optimal value of $\boldsymbol{\lambda}$ that maximizes the probability that the minimum required utility η is maintained by the network for a time horizon T under uncertainty in components' states.

The first-stage of our two-stage stochastic program is presented below:

$$\max \bar{h}(\boldsymbol{\lambda}) = \sum_{\omega \in \Omega} \prod_{\ell \in L^\omega} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega) \tag{4.2a}$$

$$\text{s.t.} \quad \sum_{\ell \in \mathcal{L}} \lambda_\ell^{-1} \leq B \tag{4.2b}$$

$$\lambda_\ell \geq 0 \qquad \forall \ell \in \mathcal{L} \tag{4.2c}$$

The objective function (4.2a) seeks to maximize the total probability that the minimum required utility η is maintained by the network for the time horizon T. The indicator

function, $\mathbf{1}_\eta(\omega)$, ensures that the objective function (4.2a) only accounts for the probabilities of the scenarios where the second-stage objective value (utility provided by the network) g^ω is at least the minimum required utility η, i.e., $g^\omega \geq \eta$. The indicator function is defined as follows:

$$\mathbf{1}_\eta(\omega) = \begin{cases} 1 & \text{if } g^\omega \geq \eta \\ 0 & \text{otherwise} \end{cases}$$

The constraint (4.2b) ensures that the total investment on the components in reducing their failure rates cannot exceed the decision maker's budget.

Realizing the survival/failure states of the components after duration T, we solve the second-stage model for each scenario ω to obtain the utility (g^ω) provided by the network in each scenario. As the second-stage model is specific to a particular application problem, we present the second-stage model for a case study in Section 4.6.1.

4.4 Theoretical Properties

We analyze the submodularity property of the first-stage reliability maximization objective function (4.2a). As we study a submodular system, the second-stage objective function in scenario ω (g^ω) is submodular. We discuss the submodularity of the second-stage objective function specific to the application problem in Section 4.6.1.

To establish the proof of submodularity of the first-stage objective function (4.2a), we need the following definition of submodularity for a differentiable function taken from Topkis [138].

Definition 1. (Submodularity) A twice differentiable function $f(\cdot)$ is submodular if and only if all off-diagonal entries of its Hessian matrix are non-positive, i.e., $\forall \boldsymbol{x} \in X, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} \leq 0, \quad \forall i \neq j$.

Theorem 1 establishes the submodularity of the reliability function $\bar{h}(\boldsymbol{\lambda})$ defined in Eq. (4.2a), when the components' lifetimes are exponentially distributed.

Theorem 1. *Given that lifetime of the network components are exponentially distributed with failure rate* $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{|\mathcal{L}|})$*, the reliability function,* $\bar{h}(\boldsymbol{\lambda})$ *(4.2a) is submodular in* $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{|\mathcal{L}|})$.

Proof. The reliability function is as follows:

$$\bar{h}(\boldsymbol{\lambda}) = \sum_{\omega\in\Omega} \prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)$$

Classifying the scenarios into two groups such that in one group component i survives ($\omega_i = 1$), whereas in another group it fails ($\omega_i = 0$), we have

$$\begin{aligned}
\bar{h}(\boldsymbol{\lambda}) &= \sum_{\omega\in\Omega(\omega_i=1)} \prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^1 \\
&\quad + \sum_{\omega\in\Omega(\omega_i=0)} \prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^0 \\
\frac{\partial\bar{h}(\boldsymbol{\lambda})}{\partial\lambda_i} &= -\sum_{\omega\in\Omega(\omega_i=1)} T \prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^1 \\
&\quad + \sum_{\omega\in\Omega(\omega_i=0)} T e^{-\lambda_i T} \prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega\cup\{i\}} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^0 \\
&= \tfrac{|\Omega|}{2} T \Big[\sum_{\omega\in\Omega(\omega_i=0)} e^{-\lambda_i T} \prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega\cup\{i\}} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^0 \\
&\quad - \sum_{\omega\in\Omega(\omega_i=1)} e^{-\lambda_i T} \prod_{\ell\in L^\omega\setminus\{i\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^1 \Big]
\end{aligned}$$

Set, $\frac{|\Omega|}{2} T = M$. Pairing the scenarios from the two groups, where each pair (ω^1, ω^2) is formed as $\omega^1 \in \Omega_{(\omega_i=0)}$ and $\omega^2 \in \Omega_{(\omega_i=1)}$ such that only the state of component i is different between ω^1 and ω^2, i.e., $\omega_\ell \in \omega^1 = \omega_\ell \in \omega^2 \quad \forall \ell \in \mathcal{L} \setminus \{i\}$. Then, we have

$$\prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega\cup\{i\}} \left(1 - e^{-\lambda_\ell T}\right) = \prod_{\ell\in L^\omega\setminus\{i\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right)$$

Therefore, $\frac{\partial\bar{h}(\boldsymbol{\lambda})}{\partial\lambda_i}$ can be written as a sum of the scenario pairs

$$\frac{\partial\bar{h}(\boldsymbol{\lambda})}{\partial\lambda_i} = M \left[e^{-\lambda_i T} \prod_{\ell\in L^\omega} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^\omega\cup\{i\}} \left(1 - e^{-\lambda_\ell T}\right) \left(\mathbf{1}_\eta(\omega)^0 - \mathbf{1}_\eta(\omega)^1\right) + \ldots \right]$$

As g^{ω} is nondecreasing submodular in ω, $\mathbf{1}_{\eta}(\omega)$ is also nondecreasing submodular in ω, making $(\mathbf{1}_{\eta}(\omega)^0 - \mathbf{1}_{\eta}(\omega)^1) \leq 0$ and thus $\frac{\partial \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i} \leq 0$.

Now, classifying each of the two groups of scenarios, $\Omega_{(\omega_i=0)}$ and $\Omega_{(\omega_i=1)}$, further into two groups such that component j survives ($\omega_j = 1$) in one group whereas fails ($\omega_j = 0$) in another, we have

$$\begin{aligned}
\frac{\partial \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i} = M\Big[& \textstyle\sum_{\omega\in\Omega(\omega_i=0,\omega_j=0)} e^{-\lambda_i T} \prod_{\ell\in L^{\omega}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{i\}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{00} \\
& + \textstyle\sum_{\omega\in\Omega(\omega_i=0,\omega_j=1)} e^{-\lambda_i T} \prod_{\ell\in L^{\omega}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{i\}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{01} \\
& - \textstyle\sum_{\omega\in\Omega(\omega_i=1,\omega_j=0)} \prod_{\ell\in L^{\omega}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{10} \\
& - \textstyle\sum_{\omega\in\Omega(\omega_i=1,\omega_j=1)} \prod_{\ell\in L^{\omega}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{11} \Big] \\
\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i \partial \lambda_j} = M\Big[& \textstyle\sum_{\omega\in\Omega(\omega_i=0,\omega_j=0)} T e^{-\lambda_i T} e^{-\lambda_j T} \prod_{\ell\in L^{\omega}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{i,j\}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{00} \\
- & \textstyle\sum_{\omega\in\Omega(\omega_i=0,\omega_j=1)} T e^{-\lambda_i T} e^{-\lambda_j T} \prod_{\ell\in L^{\omega}\setminus\{j\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{i\}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{01} \\
& - \textstyle\sum_{\omega\in\Omega(\omega_i=1,\omega_j=0)} T e^{-\lambda_i T} e^{-\lambda_j T} \prod_{\ell\in L^{\omega}\setminus\{i\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{j\}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{10} \\
& + \textstyle\sum_{\omega\in\Omega_{11}(\omega_i=1,\omega_j=1)} T e^{-\lambda_i T} e^{-\lambda_j T} \prod_{\ell\in L^{\omega}\setminus\{i,j\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}} \left(1-e^{-\lambda_\ell T}\right) \mathbf{1}_{\eta}(\omega)^{11} \Big]
\end{aligned}$$

Combining scenarios into groups of four scenarios $(\omega^1, \omega^2, \omega^3, \omega^4)$, where $\omega^1 \in \Omega_{(\omega_i=0,\omega_j=0)}$, $\omega^2 \in \Omega_{(\omega_i=0,\omega_j=1)}$, $\omega^3 \in \Omega_{(\omega_i=1,\omega_j=0)}$, and $\omega^4 \in \Omega_{(\omega_i=1,\omega_j=1)}$ such that

$$\omega_\ell \in \omega^1 = \omega_\ell \in \omega^2 = \omega_\ell \in \omega^3 = \omega_\ell \in \omega^4 \quad \forall \ell \in \mathcal{L} \setminus \{i,j\}$$

We have,

$$\begin{aligned}
& \textstyle\prod_{\ell\in L^{\omega}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{i,j\}} \left(1-e^{-\lambda_\ell T}\right) = \prod_{\ell\in L^{\omega}\setminus\{j\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{i\}} \left(1-e^{-\lambda_\ell T}\right) \\
& = \textstyle\prod_{\ell\in L^{\omega}\setminus\{i\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}\cup\{j\}} \left(1-e^{-\lambda_\ell T}\right) = \prod_{\ell\in L^{\omega}\setminus\{i,j\}} e^{-\lambda_\ell T} \prod_{\ell\in\mathcal{L}\setminus L^{\omega}} \left(1-e^{-\lambda_\ell T}\right)
\end{aligned}$$

Therefore,

$$\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i \partial \lambda_j} = M' \Big[e^{-\lambda_i T} e^{-\lambda_j T} \prod_{\ell \in L^\omega} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega \cup \{i,j\}} \left(1 - e^{-\lambda_\ell T}\right) \\ \left(\mathbf{1}_\eta(\omega)^{00} - \mathbf{1}_\eta(\omega)^{01} - \mathbf{1}_\eta(\omega)^{10} + \mathbf{1}_\eta(\omega)^{11}\right) + \dots \Big]$$

Where $M' = M\frac{|\Omega|}{4}T$. As $\mathbf{1}_\eta(\omega)$ is submodular, the following inequality holds,

$$\mathbf{1}_\eta(\omega)^{11} + \mathbf{1}_\eta(\omega)^{00} \leq \mathbf{1}_\eta(\omega)^{01} + \mathbf{1}_\eta(\omega)^{10}$$

Therefore,

$$\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i \partial \lambda_j} \leq 0, \qquad \forall i \neq j$$

Thus, by definition, the reliability function, $\bar{h}(\boldsymbol{\lambda})$, is submodular.

□

We also analyze whether our reliability function (4.2a) is DR-submodular. A twice differentiable function $f(\cdot)$ is DR-sumodular if and only if $\forall \boldsymbol{x} \in X, \frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} \leq 0, \forall i, j$ [20]. The proof of Theorem 1 shows that the $\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i \partial \lambda_j} \leq 0, \forall i \neq j$. Therefore, we need to check whether the diagonal entries of the Hessian matrix of $\bar{h}(\boldsymbol{\lambda})$ are non-positive, i.e., $\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i^2} \leq 0, \forall i$.

Theorem 2. *Given that lifetime of the network components are exponentially distributed with failure rate $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_{|\mathcal{L}|})$, the reliability function, $\bar{h}(\boldsymbol{\lambda})$ (4.2a) is not DR-submodular.*

Proof. From the proof of Theorem 1, we have

$$\frac{\partial \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i} = M \Big[\sum_{\omega \in \Omega(\omega_i = 0)} e^{-\lambda_i T} \prod_{\ell \in L^\omega} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega \cup \{i\}} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^0 \\ - \sum_{\omega \in \Omega(\omega_i = 1)} e^{-\lambda_i T} \prod_{\ell \in L^\omega \setminus \{i\}} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^1 \Big]$$

$$\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i^2} = M \Big[\sum_{\omega \in \Omega(\omega_i = 0)} -T e^{-\lambda_i T} \prod_{\ell \in L^\omega} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega \cup \{i\}} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^0 \\ + \quad \sum_{\omega \in \Omega(\omega_i = 1)} T e^{-\lambda_i T} \prod_{\ell \in L^\omega \setminus \{i\}} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \mathbf{1}_\eta(\omega)^1 \Big]$$

Pairing the scenarios from the two groups, where each pair (ω^1, ω^2) is formed as $\omega^1 \in \Omega_{(\omega_i=0)}$ and $\omega^2 \in \Omega_{(\omega_i=1)}$ such that only the state of component i is different between ω^1 and ω^2, i.e., $\omega_\ell \in \omega^1 = \omega_\ell \in \omega^2 \quad \forall \ell \in \mathcal{L} \setminus \{i\}$, we have

$$\prod_{\ell \in L^\omega} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega \cup \{i\}} \left(1 - e^{-\lambda_\ell T}\right) = \prod_{\ell \in L^\omega \setminus \{i\}} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right)$$

Therefore, $\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i^2}$ can be written as a sum of the scenario pairs as follows

$$\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i^2} = M' \left[e^{-\lambda_i T} \prod_{\ell \in L^\omega \setminus \{i\}} e^{-\lambda_\ell T} \prod_{\ell \in \mathcal{L} \setminus L^\omega} \left(1 - e^{-\lambda_\ell T}\right) \left(\mathbf{1}_\eta(\omega)^1 - \mathbf{1}_\eta(\omega)^0\right) + \dots \right]$$

As $\mathbf{1}_\eta(\omega)$ is nondecreasing submodular in ω, $(\mathbf{1}_\eta(\omega)^1 - \mathbf{1}_\eta(\omega)^0) \geq 0$.

Therefore,

$$\frac{\partial^2 \bar{h}(\boldsymbol{\lambda})}{\partial \lambda_i^2} \geq 0, \quad \forall i$$

Thus, by definition, the reliability function, $\bar{h}(\boldsymbol{\lambda})$, is not DR-submodular. □

4.5 Solution Approach

This section details the solution algorithm for solving our two-stage stochastic program. As we prove in Theorem 1 that the objective function (4.2a) of our SPEU model is submodular, we can use algorithms for continuous submodular function (CSF) maximization to solve our model. As discussed in Section 4.1.2, in CSF literature, there exist different algorithms for maximizing a constrained continuous submodular function with a worst-case performance guarantee. Some of the recent notable algorithms include conditional gradient (a.k.a., Frank-Wolfe) algorithm [103], stochastic gradient method [51] for continuous DR-submodular function maximization with general convex constraint, continuous greedy algorithm for continuous monotone submodular function maximization subject to down-closed constraint [141] and general matroid constraint [26, 64].

The objective function (4.2a) of our SPEU model is similar to the multilinear extension of a discrete submodular function maximized using the continuous greedy algorithm in Jegelka [64], with the difference that our objective function has probability functions instead of probabilities. Therefore, we implement the continuous greedy approximation algorithm presented in Jegelka [64] to solve our continuous submodular stochastic program (4.2). This algorithm is also similar to Calinescu et al. [26]. This continuous greedy approximation algorithm provides a worst-case performance guarantee of $(1-\frac{1}{e})OPT-\epsilon$ for maximizing a continuous DR-submodular function subject to a general matroid constraint, where $\epsilon = \frac{C_g}{2K}$. This worst-case performance improves as the number of iterations (K) increases. The algorithm is presented in Algorithm 2.

Algorithm 2 Continuous Greedy Algorithm

function CONTINUOUS GREEDY
  initialize $\boldsymbol{\lambda^0 = 0}$
  while $k \leq K$ **do**
    $\boldsymbol{\delta}^k = arg\,max_{\boldsymbol{\delta} \in conv(\mathcal{I})} \langle \boldsymbol{\delta}, \nabla \bar{h}(\boldsymbol{\lambda}^k) \rangle$
    $\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \alpha_k \boldsymbol{\delta}^k$
    $k \leftarrow k+1$
  end while
  return $\boldsymbol{\lambda}^K$
end function

At each iteration k of the algorithm, we find a direction $\boldsymbol{\delta}^k$ by solving a maximization problem over $conv(\mathcal{I})$, where $\mathcal{I}$ is the independent set of a matroid. In our problem, the $conv(\mathcal{I})$ is defined for the uniform matroid by the inequality shown in Eq. (4.2b). At each iteration k, we take linear combination of the current iterate ($\boldsymbol{\lambda}^k$) and direction $\boldsymbol{\delta}^k$ to obtain the next iterate ($\boldsymbol{\lambda}^{k+1}$). In this algorithm, the step size (α_k) is determined as $\alpha_k = \frac{1}{K}$.

4.6 Case Study

In this section, we present a case study for a complex submodular system reliability allocation problem. We conduct computational experiments to evaluate the performance of the stochastic programming model and the continuous approximation algorithm in solving a complex submodular system reliability allocation problem and provide insights into the problem.

4.6.1 Case Study Description

As a case study, we consider a sensor placement problem in a water supply network. It is crucial to detect malicious contaminations in the water supply as contaminations can affect a large population [72]. To detect malicious contaminations and to ensure the supply of pure drinking water to an area, sensors are placed in the network. We assume that the network configuration is fixed, meaning that the water supply network's sensor locations are pre-determined. This network aims to cover (satisfy) the minimum required demand for pure drinking water over a specified time horizon T. We generate networks based on the southeastern United States (according to the American Association of Geographers), where each node represents a city in the southeastern United States. We generate different-sized networks based on a threshold population, for example, considering cities with more than 100,000 population results in a 121-node network. We solve the maximum covering location model of Church and ReVelle [28] for the generated networks to determine the location of the sensors. There are two sets of nodes in the resulting network—sensor locations and demand points. Let $\mathcal{S}$ denotes the set of demand points indexed by s. Demand a_s of node s is set to the corresponding city population divided by 10^3. A 32-node network is shown in Figure F.7 (Appendix F) . This network structure is complex as it is not possible to represent the network as a series-parallel combination of the sensors.

As the sensors are exposed to the natural environment, their performance degrades over time and eventually becomes inoperable [72]. The network managers can invest their limited budget in placing better quality sensors having a longer lifetime to ensure the detection of contaminations and thus enable the supply of pure drinking water. However, as it is

difficult to accurately estimate the effects of numerous environmental factors on the sensors' performance, their lifetime is usually uncertain, making their survival/failure states of being uncertain at the end of time horizon T. Each scenario $\omega \in \Omega$ of our stochastic programming model represents a particular realization of all the sensors' survival/failure states at the end of time horizon T. To represent the survival/failure state of a sensor ℓ in scenario ω, we define the parameter e_ℓ^ω as follows

$$e_\ell^\omega = \begin{cases} 1 & \text{if sensor } \ell \text{ survives after time } T \text{ in scenario } \omega \\ 0 & \text{otherwise} \end{cases}$$

Realizing the sensors' survival/failure states in scenario ω, we can compute the amount of demand covered by the sensors survived in scenario ω. Let D be the maximum distance from a demand point s within which a sensor should exist (survive) for the demand point to be covered. We refer this distance D as *coverage radius*. Defining the distance between a demand point s and a sensor ℓ surviving in a scenario as $d_{s\ell}$, the demand point s is considered to be covered if $d_{s\ell} \leq D$. Therefore, the set of neighboring sensors covering a demand point s is defined as $\mathcal{N}_s := \{\ell \in \mathcal{L} | d_{s\ell} \leq D\}$. To represent whether a demand point s is covered in scenario ω, we define the variable x_s^ω as follows:

$$x_s^\omega = \begin{cases} 1 & \text{if the demand point } s \text{ is covered in scenario } \omega \\ 0 & \text{otherwise} \end{cases}$$

The goal of the network managers here is to optimally invest their budgets among the sensors to maximize the total probability over all scenarios in Ω that the minimum required demand coverage η is maintained in this network. Network managers can use the stochastic programming model (4.2) to compute the optimal investment decisions. Realizing the sensors' survival/failure states in scenario ω, the second-stage model (4.3) maximizes the amount of demand covered in the network.

$$g^{\omega} = \max \sum_{s \in \mathcal{S}} a_s x_s^{\omega} \tag{4.3a}$$

$$\text{s.t.} \quad \sum_{\ell \in \mathcal{N}_s} e_{\ell}^{\omega} \geq x_s^{\omega} \qquad \forall s \in \mathcal{S} \tag{4.3b}$$

$$x_s^{\omega} \in \{0, 1\} \qquad \forall s \in \mathcal{S} \tag{4.3c}$$

The objective function (4.3a) computes the maximum demand covered in scenario ω. Each constraint (4.3b) ensures that a demand point is covered in scenario ω if at least a sensor within distance D from the demand point survives in that scenario ω.

The second-stage demand coverage function g^{ω} is submodular in $\omega = (\omega_{\ell})_{\ell \in \mathcal{L}}$, where ω_{ℓ} denotes the state of sensor ℓ in scenario ω. The submodularity of the second-stage objective function (4.3a) is very intuitive. The value of demand covered g^{ω} increases as more sensors survive the time horizon T in scenario ω. But, the improvement in g^{ω} for an additional sensor's availability (survival) decreases (nonincreasing) as the number of surviving sensors so far increases.

This second-stage model's key feature is that unlike the traditional two-stage stochastic programs, second-stage computation depends on the first-stage decision variables, computation of the second-stage model (4.3) is independent of the first-stage model (4.2). Therefore, we can compute the second-stage values prior to solving the first-stage model.

4.6.2 Experimental Setup

We implemented the reliability maximization model and continuous greedy approximation algorithm for the water supply network discussed above in Python 2.7. We conduct numerical experiments to provide insights into the following research questions: (1) how large problems can be solved by the continuous greedy algorithm within a reasonable time, (2) how does the network size—number of nodes and number of sensors—affect the algorithm runtime, (3) how does the decision maker's budget affect the maximum reliability (objective value), (4) how do the model parameters—coverage radius (D) and minimum required demand coverage

(η)— affect the maximum reliability? In the numerical experiments, we used the parameter values as shown in Table 4.2.

4.6.3 Performance of Continuous Greedy Algorithm

This section presents the performance—runtime and solution quality—of the continuous greedy algorithm in solving our submodular stochastic programming model for the water supply networks. To test the computational efficiency of the algorithm, we ran experiments on a personal computer running MacOS with Intel Core i9 @2.9 GHz with 16GB of installed RAM. Figures 4.1 and 4.2 demonstrate the runtime and objective value (reliability), respectively, of the continuous greedy algorithm for solving different-sized problem instances with varying number of sensors, number of iterations, and decision maker's budget. We allowed the algorithm to run for 1 hour.

Figure 4.1 demonstrates that the continuous greedy algorithm can solve the stochastic submodular reliability allocation problem for most 69- and 121-node network instances in less than 30 minutes. We see from Figure 4.1 that runtime increases substantially as the number of sensors in a network increases. As the number of sensors increases, the number of scenarios in our stochastic programming model increases substantially. Therefore, in each iteration of the algorithm, the gradient is computed for an increased number of scenarios, resulting in a larger runtime. Figure 4.1 shows that runtime increases with the number of iterations (K) within the same budget level, which is intuitive. The algorithm runtime does not show any noticeable trend as the network size increases.

Figure 4.2 shows that the solution quality improves as the number of iterations increases within the same budget level, consistent with the algorithm's theoretical performance. As mentioned earlier, the worst-case performance $\left((1-\frac{1}{e})OPT - \frac{C_g}{2K}\right)$ improves as the number of iterations increases. However, it is evident from Figure 4.2 that the rate of improvement in solution quality diminishes as the number of iterations increases beyond a certain value while increasing runtime substantially. For instance, we see in Figure 4.1 that the algorithm cannot solve the problem instances with 18 sensors (number of scenarios, $|\Omega| > 260,000$) within 1 hour when the number of iterations goes beyond 40. Therefore, practitioners should

Table 4.2: Parameters and their values used in the computational experiments.

Parameter	**Values Used**
Network size (Number of nodes)	69, 121
Number of sensors, $\lvert\mathcal{L}\rvert$	12, 15, 18
Coverage radius, D (low, high)	(50 miles, 350 miles)
Minimum required demand coverage, η (low, high)	(10%, 90%)
Budget, B (low, high)	(10, 120)
Number of iterations, K (low, high)	(10, 100)

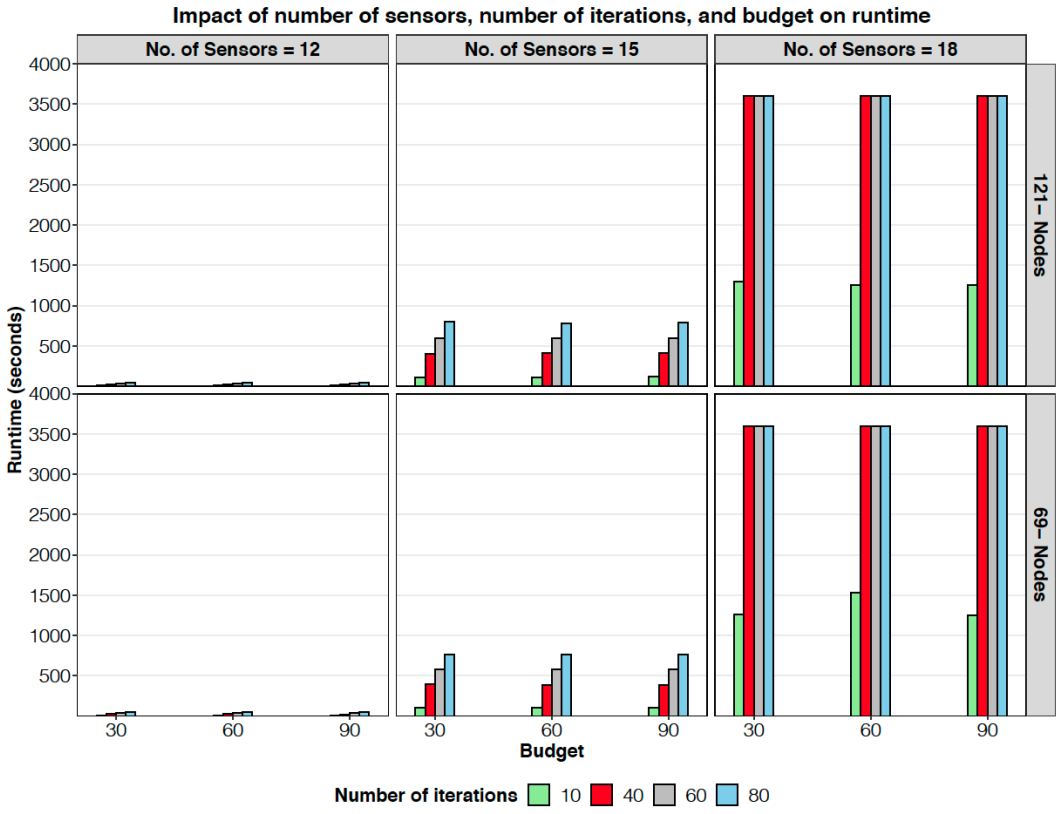


Figure 4.1: Impact of network size, number of sensors, number of iterations, and decision-maker's budget on runtime.

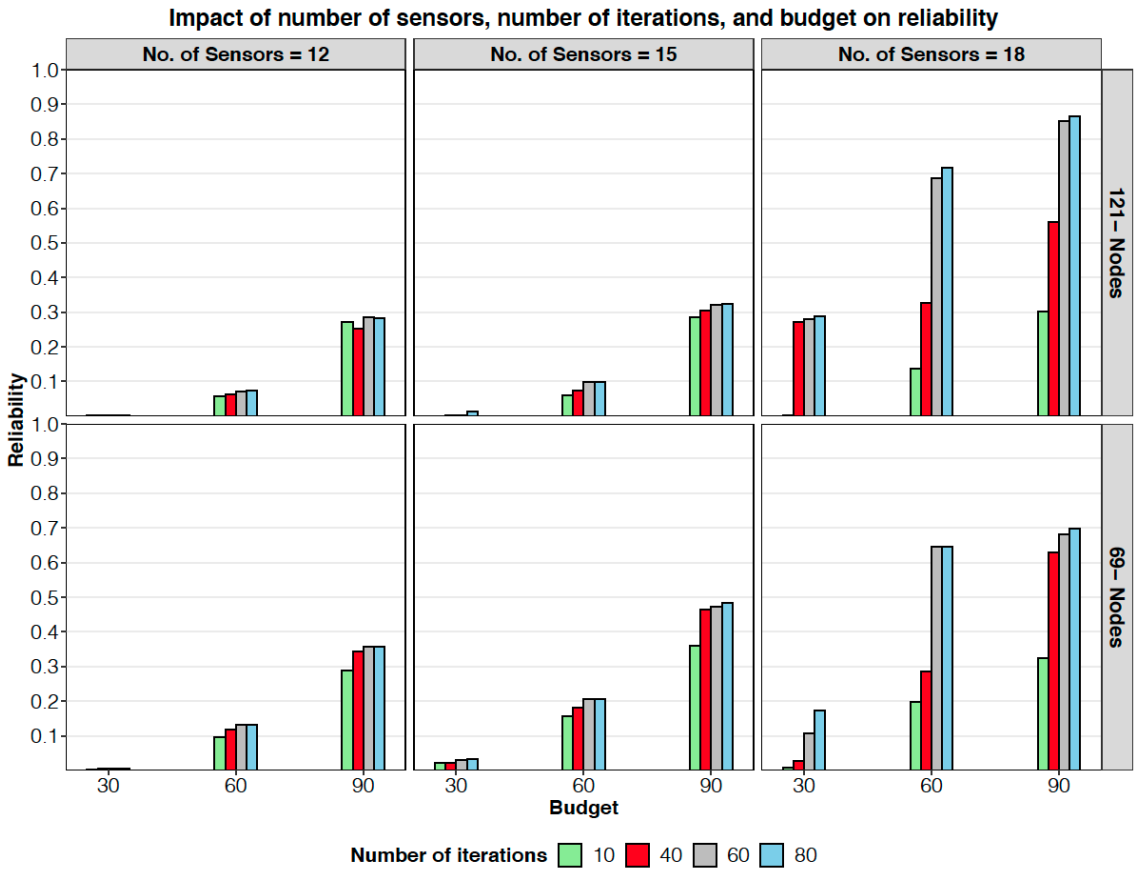


Figure 4.2: Impact of number of sensors, number of iterations, and decision maker's budget on reliability.

consider this trade-off between the runtime and solution quality in setting the algorithm's number of iterations (K) parameter.

4.6.4 Effect of Budget, Coverage Radius, and Minimum Required Demand Coverage on Reliability

Figure 4.3 demonstrates the variation of reliability with the decision maker's budget for two different networks, 69- and 121-node networks, each with 15 sensors. We see that reliability increases as the decision maker's budget increases, which is very intuitive. With an increased budget, the decision-maker can invest more resources in the sensors, reducing their failure rate and increasing the likelihood of survival. As the more sensors survive, the more demand can be covered, which increases the probability that the minimum required demand is covered in the network.

However, reliability does not increase uniformly as the budget increases. Initially, with a small amount of budget, it is impossible to make sufficient investment to increase the lifetime of the sensors substantially. As the budget increases, more investment is made to the critical sensors that cover a substantial amount of demand, increasing the reliability at a higher rate. However, as the budget becomes large enough that a large investment can be made to most of the sensors, the additional budget has less impact on reliability improvement.

Practitioners can use our model along with their own datasets to produce a figure like Figure 4.3 to decide on the amount to invest in the network sensors, depending on their desired reliability.

We demonstrate a solution for a 32-node network instance obtained from the continuous greedy algorithm in Figure F.7 (Appendix F) to clarify how a given budget is allocated to the sensors to maximize reliability. The solution demonstrates that when the budget is not large enough to invest a fair amount of resources to all sensors to reduce all the sensors' failure rate, the model tries to minimize the failure rate of the critical sensors that cover larger demand.

Table 4.3 demonstrates the variation of reliability (objective value) with coverage radius across different decision maker's budget levels. As the coverage radius in the water supply

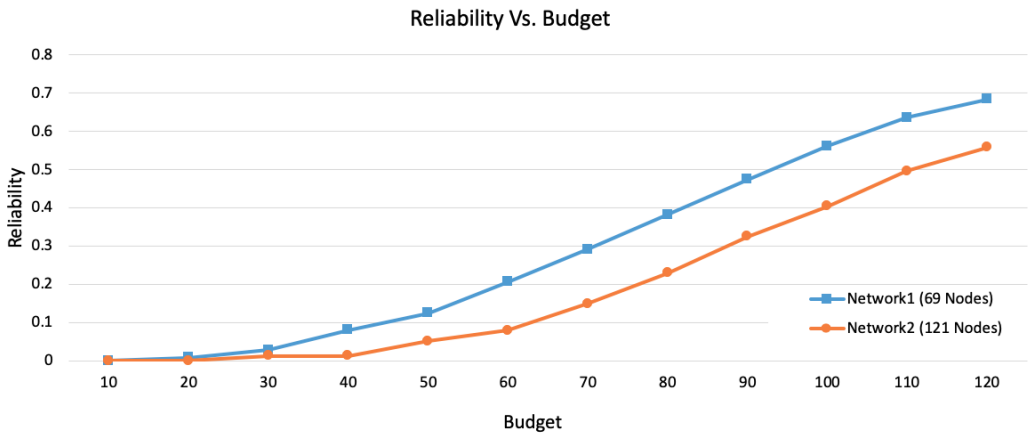


Figure 4.3: Variation of reliability with decision maker's budget for 69- and 121-node networks.

Table 4.3: Variation of reliability with coverage radius, D. Other parameters are: $K = 80$ and $\eta = 60\%$ of the total demand.

D	**Budget**			
	20	**50**	**80**	**110**
50	0	0	0	0
100	0	0.0121	0.0244	0.0935
150	0.0001	0.0346	0.2005	0.4696
200	0.0025	0.1538	0.3832	0.6414
250	0.0089	0.1489	0.4951	0.7438
300	0.0081	0.1702	0.5138	0.8086
350	0.0093	0.1784	0.5841	0.8632

network increases, a demand point can be covered by a sensor located far away from the demand point. Therefore, the larger the coverage radius, the more fraction of the total demand is covered in the scenarios, resulting in a higher probability that the minimum required demand (η) is covered by the network.

Table 4.4 demonstrates the variation of reliability with the minimum required demand coverage (η) in the network for different budget levels. Here, η is shown as a percentage of the total demand in the network. As the minimum required demand coverage (η) increases, the network fails to cover this larger η in more number of scenarios, i.e., $g^{\omega} < \eta$. Therefore, the probability of these scenarios has no contribution to the objective value of the stochastic program (reliability). Thus the larger the value of η, the smaller is the reliability within the same budget level.

4.7 Conclusions

We studied a particular case of a complex system reliability allocation problem, where the system is submodular. In this problem, the decision-maker seeks to optimally invest the limited resources to the network components to maximize the probability that the network provides a minimum required utility over a specified time horizon. We assume that components' lifetimes are uncertain to the decision-maker and distributed exponentially. The lifetime of a component depends probabilistically on the investment amount to that component.

We formulated the problem as a two-stage stochastic programming model with endogenous uncertainty, where the first-stage investment decisions affect the scenario probabilities. Utilizing the complex system's submodularity, we proved that our stochastic programming model's objective function is also submodular. This research contributes to the SPEU literature by establishing the SPEU modeling framework's submodularity when probability distribution (exponential distribution) governing uncertainty does not have semigroup property. We implemented a continuous greedy approximation algorithm to solve our submodular stochastic programming model.

Table 4.4: Variation of reliability with minimum demand to be covered η. Other parameters are: $K = 80$ and $D = 150$ miles.

η%	Budget			
	20	**50**	**80**	**110**
20	0.1552	0.4823	0.7847	0.9307
30	0.1044	0.3793	0.7077	0.8735
40	0.0389	0.1600	0.4788	0.7299
50	0.0060	0.1417	0.3987	0.6279
60	0.0001	0.0346	0.2006	0.4696
70	0.0001	0.0128	0.0779	0.2214
80	0	0.0069	0.0338	0.0894
90	0	0	0	0.001

We conduct computational experiments based on a case study of sensor placement in a water supply network. Results show that the continuous approximation algorithm can solve large problem instances within a reasonable amount of time (less than an hour). Algorithm runtime increases as the network size, the number of sensors in the water supply network, and iterations increase. System reliability increases as the decision maker's budget increases. Reliability also increases as the maximum distance (coverage radius) within which a sensor should exist to cover a demand point increases. Reliability decreases as the minimum required demand coverage in the network increases.

This research can be extended in several possible directions. We plan to provide a numerical performance comparison of the continuous approximation algorithm with a global optimization solver (e.g., BARON). A possible extension is to study the submodularity in SPEU literature for other probability distributions commonly used in reliability allocation problems, such as the Weibull distribution. Another possible extension is to establish the worst-case performance guarantee of the continuous approximation algorithm in solving the submodular SPEU framework for exponential and Weibull distributions. Also, other application areas of submodular systems can be explored, such as maximizing marketing coverage of new products, where the investment decisions can affect the underlying uncertainty as well.

Chapter 5

Conclusions

This dissertation focuses on (1) developing new SPEU models for problems having natural endogenous (decision-dependent) uncertainty, (2) presenting mathematical reformulations to linearize the nonconvex models, (3) establishing submodularity of the modeling framework for some probability distributions, (4) designing and implementing exact and approximate solution approaches utilizing the structure of the models, and (5) demonstrating the benefit of this modeling and solution approaches to different application areas. Methodologies presented in this dissertation are useful for practitioners in solving problems with endogenous uncertainty. The numerical results provide key managerial insights and policy-making implications to the practitioners in those application areas. Practitioners can also adopt the proposed models and solution approaches in other application areas.

The following sections draw more specific conclusions to each aspect of this dissertation and discuss potential future research directions.

5.1 Models and Applications of SPEU

This dissertation presents new SPEU models for both discrete and continuous decision variables. In chapters 2 and 3, we introduced new data-driven SPEU models with discrete decision variables, whereas chapter 4 presents an SPEU model with continuous decision

variables. The models presented in chapters 2 and 3 are data-driven as they integrate supervised machine learning algorithms into these stochastic programming models to estimate the probability distributions of the uncertain parameters by analyzing historical/survey data.

Chapter 2 introduces a data-driven two-stage SPEU model for a risk-based incentive structure design problem seeking to optimally allocate monetary incentives to private landowners to encourage them in reducing the risk of potential future wildfires by implementing hazardous fuel reduction treatment. This model provides the solution to a practical and challenging problem that has not been addressed to date. Our proposed model captures the realistic nature of the problem by modeling landowner behavior to be a probabilistic function of the incentive amount offered to the landowners. This data-driven model integrates a logistic regression model to estimate the conditional probability distributions of the landowners' accepting a given incentive amount. As the probability distributions of landowners' behaviors are a function of incentive allocation decisions, and thus, scenarios probabilities are a function of the decisions, the resulting stochastic optimization model is nonconvex. Results from the model based on Santa Fe National Forest data provide policy implications to the fire managers in investing their limited budgets and choosing budget allocation levels to best accomplish their wildfire risk-mitigating objectives.

Chapter 3 presents a new data-driven stochastic programming model for problems where both endogenous and exogenous uncertainties coexist. Unlike the SPEU model of chapter 2, first-stage decisions affect only the scenario probabilities, in this model, some first-stage decision variables affect the scenario probabilities (decision-dependent uncertainty) whereas, other variables affect recourse decisions as in exogenous uncertainty. The proposed model is used to formulate an integrated network design and infrastructure protection problem with the realistic assumption that the effects of protection and disruption on infrastructure are imperfect. Infrastructure is likely to have a higher post-disruption capacity as the protection investment increases and disruption intensity to which it is exposed decreases. We integrated several supervised machine learning algorithms into the stochastic optimization model to estimate the conditional probability distributions of the post-disruption capacities. As the scenario probabilities are a function of the protection investment decisions, the resulting model is nonconvex.

Computational results demonstrate that computational complexity in solving the model with endogenous and exogenous uncertainty is substantially higher than solving the model with only endogenous uncertainty, as in chapter 2. But, modeling the realistic imperfect protection and disruption using the data-driven SPEU model provides a substantially better solution than the models assuming perfect protection. Also, we found that the proposed model in chapter 3 provides a substantially better solution than the models ignoring underlying uncertainty. Another key finding is that the optimal solution of the data-driven stochastic optimization model is sensitive to the estimation error of the supervised learning algorithms used. Therefore, practitioners should consider using a supervised learning algorithm with the least estimation error.

Chapter 4 studies the submodularity of the reliability maximization objective function of a complex submodular system. We formulated the problem as a two-stage SPEU model, where the continuous investment decision variables probabilistically affect the lifetime of the system components. We conduct a case study based on a water supply network to present insights into the problem.

5.2 Theoretical Properties and Solution Approaches for SPEU

This dissertation presents mathematical reformulations for linearizing nonconvex models. It also presents simulation, exact decomposition, and approximation approaches to solve optimization problems under endogenous uncertainty.

In chapter 2, we implemented a reformulation technique to linearize the nonconvex model. We faced a challenge in solving a realistic-size problem over large landscapes because the number of scenarios in our stochastic program and the wildfire sub-scenarios in each scenario becomes very large (millions), making the model computationally intractable. To overcome this challenge, we integrated a simulation program that computes the second-stage objective value in each scenario by modeling the wildfire spread through the landscape, accounting for fuel reduction treatment information, weather and landscape characteristics. We modeled the

landscape as a directed network and wildfire spread as a network flow problem in computing the wildfire damage in each scenario. This network flow problem relates our problem to the class of stochastic network interdiction problem, as the allocation of fuel treatment resources interdicts (prevents) the spread of wildfires. Computational results suggest that this data-driven simulation-integrated SPEU model can be used to solve realistic-size problems within a reasonable time.

Chapter 3 proposes a mathematical reformulation to linearize the nonconvex model and prove that this reformulation is exact without any approximation gap. Taking advantage of the structure of the reformulated model presented in chapter 3, we implemented a L-shaped decomposition algorithm to solve the model. We introduced several valid inequalities to enhance the computational efficiency of the algorithm. We found that this exact decomposition algorithm's runtime increases exponentially as the problem size—number of nodes and facilities, number of protection levels, and capacity states—increases. The accelerated L-shaped algorithm can solve moderately large-sized problem instances (32-node network with six facilities) within a reasonable time (2 hours). Remarkably, the algorithm fails to solve larger problems when the number of scenarios in the stochastic program goes beyond 3000, i.e., $|\Omega| > 3000$. This computational limitation raises the necessity of a more efficient algorithm or further enhancement to this algorithm's computational efficiency.

Chapter 4 studies the submodularity of the objective function of the SPEU framework when the uncertain parameters follow an exponential distribution. We proved that the reliability maximization objective function is submodular but not DR-submodular for the exponential distribution. Taking advantage of this submodularity, we implemented a continuous greedy approximation algorithm capable of solving large-scale problems much faster than the exact algorithms.

5.3 Limitations and Future Work

We assumed in the mathematical models and reformulations that the elements (e.g., facilities) are independent of each other, meaning that one element's probability distribution is independent of another. But, in some application areas, such as in power systems,

components can be interdependent. For example, the failure of a substation can cause the failure of another, leading to a cascading failure. In this case, our exact reformulation does not work. This limitation spurs the necessity to develop a new SPEU model and reformulations to solve problems with interdependent elements.

Our proposed accelerated L-shaped decomposition algorithm has computational limitations in solving a very large problem when both endogenous and exogenous uncertainties coexist. A sample average approximation algorithm can be used to alleviate the issue with a large number of scenarios.

Though multi-stage stochastic programs have been extensively studied in SPEA literature, this framework has not gained much attention in SPEU literature to date. But, the multi-stage SPEU model is beneficial in many application areas, such as the multi-period version of the incentive structure design for fuel reduction treatment in the wildfire risk reduction problem studied in chapter 2. Naturally, fuel reduction treatment is a multi-period problem, as fuel accumulates over time in the landscape. Therefore, to reduce the risk of wildfires over a long time, an incentive program needs to be designed that accounts for the fuel re-growth each year and make adjustments in the decisions over the planning horizon. This requires to model the problem as a multi-stage SPEU and warrants developing new solution approaches.

A possible extension can be developing mean-risk SPEU models by incorporating risk measures such as conditional-value at risk and develop new solution approaches. Some applications of this new mean-risk modeling and solution approaches include mitigating the risk of substantially large financial damage and fatalities from catastrophic wildfires and mitigating the risk of severe cyber-attacks [15], where investment decisions can reduce the likelihood of attack success.

Theoretical properties of the SPEU framework can be studied for other commonly used probability distributions. It would also be beneficial to study whether the mean-risk objective function of the SPEU framework is submodular. Submodularity of the mean-risk function would benefit many real-life application areas, including but not limited to cyber-security, disaster mitigation, supply chain management.

Bibliography

[1] Acuna, M. A., Palma, C. D., Cui, W., Martell, D. L., and Weintraub, A. (2010). Integrated spatial fire and forest management planning. *Canadian Journal of Forest Research*, 40(12):2370–2383. 13

[2] Agee, J. K. and Skinner, C. N. (2005). Basic principles of forest fuel reduction treatments. *Forest Ecology and Management*, 211(1-2):83–96. 11

[3] Ager, A. A., Vaillant, N. M., and Finney, M. A. (2010). A comparison of landscape fuel treatment strategies to mitigate wildland fire risk in the urban interface and preserve old forest structure. *Forest Ecology and Management*, 259(8):1556–1570. 12

[4] Ahmed, S. (2000). *Strategic planning under uncertainty: Stochastic integer programming approaches*. PhD thesis, University of Illinois at Urbana-Champaign. 5, 62

[5] Aksen, D., Akca, S. Ş., and Aras, N. (2014). A bilevel partial interdiction problem with capacitated facilities and demand outsourcing. *Computers & Operations Research*, 41:346–358. 59

[6] Aksen, D., Aras, N., and Piyade, N. (2013). A bilevel p-median model for the planning and protection of critical facilities. *Journal of Heuristics*, 19(2):373–398. 58

[7] Aksen, D., Piyade, N., and Aras, N. (2010). The budget constrained r-interdiction median problem with capacity expansion. *Central European Journal of Operations Research*, 18(3):269–291. 58

[8] Apap, R. M. and Grossmann, I. E. (2017). Models and computational strategies for multistage stochastic programming under endogenous and exogenous uncertainties. *Computers & Chemical Engineering*, 103:233–274. 62

[9] Azad, N., Davoudpour, H., Saharidis, G. K., and Shiripour, M. (2014). A new model to mitigating random disruption risks of facility and transportation in supply chain network design. *The International Journal of Advanced Manufacturing Technology*, 70(9-12):1757–1774. 61

[10] Azad, N., Saharidis, G. K., Davoudpour, H., Malekly, H., and Yektamaram, S. A. (2013). Strategies for protecting supply chain networks against facility and transportation disruptions: an improved benders decomposition approach. *Annals of Operations Research*, 210(1):125–163. 61

[11] Azizi, N., Chauhan, S., Salhi, S., and Vidyarthi, N. (2016). The impact of hub failure in hub-and-spoke networks: Mathematical formulations and solution techniques. *Computers & Operations Research*, 65:174–188. 59

[12] Barrionuevo, A. and Deutsch, C. H. (2005). A distribution system brought to its knees. *New York Times*, 1:C1. 57

[13] Benders, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252. 77

[14] Bettinger, P. (2010). An overview of methods for incorporating wildfires into forest planning models. *Mathematical and Computational Forestry & Natural-Resource Sciences*, 2(1):43–52. 12

[15] Bhuiyan, T. H. (2018). Risk-averse bi-level stochastic network interdiction model for cyber-security risk management. Master's thesis, Mississippi State University. 148

[16] Bhuiyan, T. H., Medal, H. R., and Harun, S. (2020). A stochastic programming model with endogenous and exogenous uncertainty for reliable network design under random disruption. *European Journal of Operational Research*, 285(2):670–694. 8

[17] Bhuiyan, T. H., Moseley, M. C., Medal, H. R., Rashidi, E., and Grala, R. K. (2019). A stochastic programming model with endogenous uncertainty for incentivizing fuel reduction treatment under uncertain landowner behavior. *European Journal of Operational Research*, 277(2):699–718. 7, 62

[18] Bhuiyan, T. H., Nandi, A. K., Medal, H., and Halappanavar, M. (2016). Minimizing expected maximum risk from cyber-attacks with probabilistic attack success. In *IEEE Symposium on Technologies for Homeland Security*, pages 1–6. IEEE. 77

[19] Bian, A. A., Mirzasoleiman, B., Buhmann, J., and Krause, A. (2017). Guaranteed non-convex optimization: Submodular maximization over continuous domains. In *Artificial Intelligence and Statistics*, pages 111–120. 119

[20] Bian, Y., Buhmann, J. M., and Krause, A. (2020). Continuous submodular function maximization. *arXiv preprint arXiv:2006.13474*. 128

[21] Birge, J. R. and Louveaux, F. (2011). *Introduction to stochastic programming*. Springer series in operations research and Financial Engineering, New York: Springer. 81, 87, 106

[22] Birge, J. R. and Louveaux, F. V. (1988). A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research*, 34(3):384–392. 81

[23] Brusentsev, V. and Vroman, W. (2016). Wildfires in the United States. Technical report, Urban Institute. 10

[24] Burer, S. and Letchford, A. N. (2012). Non-convex mixed-integer nonlinear programming: A survey. *Surveys in Operations Research and Management Science*, 17(2):97–106. 54

[25] Busby, G. and Albers, H. J. (2010). Wildfire risk management on a landscape with public and private ownership: who pays for protection? *Environmental Management*, 45(2):296–310. 52

[26] Calinescu, G., Chekuri, C., Pal, M., and Vondrák, J. (2011). Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766. 120, 129, 130

[27] Chekuri, C., Vondrák, J., and Zenklusen, R. (2014). Submodular function maximization via the multilinear relaxation and contention resolution schemes. *SIAM Journal on Computing*, 43(6):1831–1879. 120

[28] Church, R. and ReVelle, C. (1974). The maximal covering location problem. In *Papers of the Regional Science Association*, volume 32, pages 101–118. 131

[29] Church, R. L. and Scaparra, M. P. (2007). Protecting critical assets: the r-interdiction median problem with fortification. *Geographical Analysis*, 39(2):129–146. 58, 59

[30] Clark, D. and Takahashi, Y. (2011). Quake disrupts key supply chains. *The Wall Street Journal Asia, March*, 12. 57

[31] Coit, D. W. and Zio, E. (2019). The evolution of system reliability optimization. *Reliability Engineering & System Safety*, 192:106259. 119

[32] Daskin, M. S. (1995). *Network and Discrete Location: Models, Algorithms, and Applications*. Wiley, New York. 92

[33] Dillman, D. A., Smyth, J. D., and Christian, L. M. (2009). *Internet, mail, and mixed-mode surveys: The tailored design method*. John Wiley & Sons, Hoboken, NJ. 30

[34] Du, L. and Peeta, S. (2014). A stochastic optimization model to reduce expected post-disaster response time through pre-disaster investment decisions. *Networks and Spatial Economics*, 14(2):271–295. 5, 62

[35] Eiselt, H. A. and Marianov, V. (2012). Mobile phone tower location for survival after natural disasters. *European Journal of Operational Research*, 216(3):563–572. 59

[36] Ekin, T. (2018). Integrated maintenance and production planning with endogenous uncertain yield. *Reliability Engineering & System Safety*, 179:52–61. 6, 62

[37] Ekin, T., Polson, N. G., and Soyer, R. (2017). Augmented nested sampling for stochastic programs with recourse and endogenous uncertainty. *Naval Research Logistics (NRL)*, 64(8):613–627. 6

[38] Escudero, L. F., Garín, M. A., Monge, J. F., and Unzueta, A. (2018). On preparedness resource allocation planning for natural disaster relief under endogenous uncertainty with time-consistent risk-averse management. *Computers & Operations Research*, 98:84–102. 6, 62

[39] Fang, Y.-P. and Zio, E. (2019). An adaptive robust framework for the optimization of the resilience of interdependent infrastructures under natural hazards. *European Journal of Operational Research*, 276(3):1119–1136. 60

[40] Ferreira, L., Constantino, M., and Borges, J. G. (2014). A stochastic approach to optimize Maritime pine (Pinus pinaster Ait.) stand management scheduling under fire risk. An application in Portugal. *Annals of Operations Research*, 219(1):359–377. 14

[41] Finney, M. A. (2002). Fire growth using minimum travel time methods. *Canadian Journal of Forest Research*, 32(8):1420–1424. 32

[42] Finney, M. A. (2006). An overview of flammap fire modeling capabilities. In Andrews, P. L. and Butler, B. W., editors, *Fuels Management–How to Measure Success: Conference Proceedings. 28-30 March 2006; Portland, OR. Proceedings RMRS-P-41. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station*, pages 213–220. 31

[43] Finney, M. A. (2008). A computational method for optimising fuel treatment locations. *International Journal of Wildland Fire*, 16(6):702–711. 11, 12, 13

[44] Ghaffarinasab, N. and Atayi, R. (2018). An implicit enumeration algorithm for the hub interdiction median problem with fortification. *European Journal of Operational Research*, 267(1):23–39. 59

[45] Goel, V. and Grossmann, I. E. (2004). A stochastic programming approach to planning of offshore gas field developments under uncertainty in reserves. *Computers & Chemical Engineering*, 28(8):1409–1429. 1, 2, 53, 61, 62

[46] Goel, V. and Grossmann, I. E. (2006). A class of stochastic programs with decision dependent uncertainty. *Mathematical Programming*, 108(2-3):355–394. 2

[47] Green, D., Gill, A. M., and Noble, I. (1983). Fire shapes and the adequacy of fire-spread models. *Ecological Modelling*, 20(1):33–45. 32

[48] Gurobi Optimization Inc. (2017). Gurobi optimizer reference manual. URL: http://www. gurobi. com. 22, 29, 35, 92

[49] Haines, T., Renner, C., Reams, M., and Granskog, J. (2008). The national wildfire mitigation programs database: state, county, and local efforts to reduce wildfire risk. In González-Cabán, A., editor, *Proceedings of the Second International Symposium On Fire Economics, Planning, and Policy: A Global View. General Technical Report. PSW-GTR-208. Albany, CA: U.S. Department of Agriculture, Forest Service, Pacific Southwest Research Station*, pages 505–511. 11

[50] Hamming, R. W. (1950). Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2):147–160. 83

[51] Hassani, H., Soltanolkotabi, M., and Karbasi, A. (2017). Gradient methods for submodular maximization. In *Advances in Neural Information Processing Systems*, pages 5841–5851. 119, 129

[52] Hatefi, S. and Jolai, F. (2014). Robust and reliable forward–reverse logistics network design under demand uncertainty and facility disruptions. *Applied Mathematical Modelling*, 38(9-10):2630–2647. 60

[53] Held, H. and Woodruff, D. L. (2005). Heuristics for multi-stage interdiction of stochastic networks. *Journal of Heuristics*, 11(5-6):483–500. 5

[54] Hellemo, L. (2016). *Managing uncertainty in design and operation of natural gas infrastructure*. PhD thesis, Norwegian University of Science and Technology. 62

[55] Hellemo, L., Barton, P. I., and Tomasgard, A. (2018). Decision-dependent probabilities in stochastic programs with recourse. *Computational Management Science*, 15(3-4):369–395. 6

[56] Higgins, A., Whitten, S., Slijepcevic, A., Fogarty, L., and Laredo, L. (2011). An optimisation modelling approach to seasonal resource allocation for planned burning. *International Journal of Wildland Fire*, 20(2):175–183. 13

[57] Hof, J. and Omi, P. (2003). Scheduling removals for fuels management. In Omi, P. N. and Joyce, L. A., editors, *Fire, Fuel Treatments, and Ecological Restoration: Conference Proceedings; 16-18 April 2002; Fort Collins, CO. Proceedings RMRS-P-29. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station*, pages 367–378. 13

[58] Hof, J., Omi, P. N., Bevers, M., and Laven, R. D. (2000). A timing-oriented approach to spatial allocation of fire management effort. *Forest Science*, 46(3):442–451. 13

[59] Huggett, R. J., Abt, K. L., and Shepperd, W. (2008). Efficacy of mechanical fuel treatments for reducing wildfire hazard. *Forest Policy and Economics*, 10(6):408–414. 11

[60] Hwang, C.-L., Tillman, F. A., and Kuo, W. (1979). Reliability optimization by generalized lagrangian-function and reduced-gradient methods. *IEEE Transactions on reliability*, 28(4):316–319. 118

[61] Isidore, C. (2018). Ford F-150s could be in short supply after plant fire. *CNN Money*. 57

[62] Jabbarzadeh, A., Fahimnia, B., Sheu, J.-B., and Moghadam, H. S. (2016). Designing a supply chain resilient to major disruptions and supply/demand interruptions. *Transportation Research Part B: Methodological*, 94:121–149. 59

[63] James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer. 90, 91

[64] Jegelka, S. (2015). Lecture notes on learning with combinatorial structure. http://people.csail.mit.edu/stefje/fall15/index.html. 129, 130

[65] Karaesmen, I. and Van Ryzin, G. (2004). Overbooking with substitutable inventory classes. *Operations Research*, 52(1):83–104. 7, 120

[66] Keyvanshokooh, E., Ryan, S. M., and Kabir, E. (2016). Hybrid robust and stochastic optimization for closed-loop supply chain network design using accelerated benders decomposition. *European Journal of Operational Research*, 249(1):76–92. 77

[67] Kim, J. Y. and Frair, L. (1981). Optimal reliability design for complex systems. *IEEE Transactions on Reliability*, 30(3):300–302. 118

[68] Kim, Y. and Kang, W.-H. (2013). Network reliability analysis of complex systems using a non-simulation-based method. *Reliability engineering & system safety*, 110:80–88. 117

[69] Konoshima, M., Albers, H. J., Montgomery, C. A., and Arthur, J. L. (2010). Optimal spatial patterns of fuel management and timber harvest with fire risk. *Canadian Journal of Forest Research*, 40(1):95–108. 14

[70] Konoshima, M., Montgomery, C. A., Albers, H. J., and Arthur, J. L. (2008). Spatial-endogenous fire risk and efficient fuel management and timber harvest. *Land Economics*, 84(3):449–468. 14

[71] Krasko, V. and Rebennack, S. (2017). Two-stage stochastic mixed-integer nonlinear programming model for post-wildfire debris flow hazard management: Mitigation and emergency evacuation. *European Journal of Operational Research*, 263(1):265–282. 6, 62

[72] Krause, A. and Guestrin, C. (2011). Submodularity and its applications in optimized information gathering. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(4):1–20. 116, 131

[73] Kumar, A., Pant, S., and Ram, M. (2017). System reliability optimization using gray wolf optimizer algorithm. *Quality and Reliability Engineering International*, 33(7):1327–1335. 119

[74] Kumar, R., Parida, P. P., and Gupta, M. (2002). Topological design of communication networks using multiobjective genetic optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation*, pages 425–430. 119

[75] Lappas, N. H. and Gounaris, C. E. (2016). Multi-stage adjustable robust optimization for process scheduling under uncertainty. *AIChE Journal*, 62(5):1646–1667. 2

[76] Lappas, N. H. and Gounaris, C. E. (2018). Robust optimization for decision-making under endogenous uncertainty. *Computers & Chemical Engineering*, 111:252–266. 2

[77] Li, D., Sun, X., and McKinnon, K. (2005). An exact solution method for reliability optimization in complex systems. *Annals of Operations Research*, 133(1-4):129–148. 117, 118

[78] Li, Q., Zeng, B., and Savachkin, A. (2013). Reliable facility location design under disruptions. *Computers & Operations Research*, 40(4):901–909. 58

[79] Liberatore, F. and Scaparra, M. P. (2011). Optimizing protection strategies for supply chains: comparing classic decision-making criteria in an uncertain environment. *Annals of the Association of American Geographers*, 101(6):1241–1258. 58

[80] Liberatore, F., Scaparra, M. P., and Daskin, M. S. (2011). Analysis of facility protection strategies against an uncertain number of attacks: The stochastic r-interdiction median problem with fortification. *Computers & Operations Research*, 38(1):357–366. 58

[81] Liberatore, F., Scaparra, M. P., and Daskin, M. S. (2012). Hedging against disruptions with ripple effects in location analysis. *Omega*, 40(1):21–30. 58

[82] Lim, M., Daskin, M. S., Bassamboo, A., and Chopra, S. (2010). A facility reliability problem: Formulation, properties, and algorithm. *Naval Research Logistics (NRL)*, 57(1):58–70. 58

[83] Linderoth, J. and Wright, S. (2003). Decomposition algorithms for stochastic programming on a computational grid. *Computational Optimization and Applications*, 24(2-3):207–250. 83

[84] Losada, C., Scaparra, M. P., Church, R. L., and Daskin, M. S. (2012a). The stochastic interdiction median problem with disruption intensity levels. *Annals of Operations Research*, 201(1):345–365. 59

[85] Losada, C., Scaparra, M. P., and O'Hanley, J. R. (2012b). Optimizing system resilience: a facility protection model with recovery time. *European Journal of Operational Research*, 217(3):519–530. 59

[86] Luo, F. and Mehrotra, S. (2020). Distributionally robust optimization with decision dependent ambiguity sets. *Optimization Letters*, pages 1–30. 2

[87] Magnanti, T. L. and Wong, R. T. (1981). Accelerating benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, 29(3):464–484. 85

[88] Maier, S., Pflug, G. C., and Polak, J. W. (2019). Valuing portfolios of interdependent real options under exogenous and endogenous uncertainties. *European Journal of Operational Research*, (In press). 62

[89] Marseguerra, M., Zio, E., and Podofillini, L. (2004). Optimal reliability/availability of uncertain systems via multi-objective genetic algorithms. *IEEE Transactions on Reliability*, 53(3):424–434. 119

[90] Marseguerra, M., Zio, E., Podofillini, L., and Coit, D. W. (2005). Optimal design of reliable network systems in presence of uncertainty. *IEEE Transactions on Reliability*, 54(2):243–253. 119

[91] Martell, D. L. (2015). A review of recent forest and wildland fire management decision support systems research. *Current Forestry Reports*, 1(2):128–137. 12

[92] McCormick, G. P. (1976). Computability of global solutions to factorable nonconvex programs: Part I—convex underestimating problems. *Mathematical Programming*, 10(1):147–175. 75

[93] Medal, H. R., Pohl, E. A., and Rossetti, M. D. (2014). A multi-objective integrated facility location-hardening model: Analyzing the pre-and post-disruption tradeoff. *European Journal of Operational Research*, 237(1):257–270. 58

[94] Medal, H. R., Pohl, E. A., and Rossetti, M. D. (2016). Allocating protection resources to facilities when the effect of protection is uncertain. *IIE Transactions*, 48(3):220–234. 2, 5, 7, 24, 57, 59, 62, 63, 65, 72, 106, 108, 111, 120

[95] Melkote, S. and Daskin, M. S. (2001). Capacitated facility location/network design problems. *European Journal of Operational Research*, 129(3):481–495. 92

[96] Mercier, A., Cordeau, J.-F., and Soumis, F. (2005). A computational study of benders decomposition for the integrated aircraft routing and crew scheduling problem. *Computers & Operations Research*, 32(6):1451–1476. 85

[97] Minas, J., Hearne, J., and Martell, D. (2015). An integrated optimization model for fuel management and fire suppression preparedness planning. *Annals of Operations Research*, 232(1):201–215. 13

[98] Minas, J. P., Hearne, J. W., and Handmer, J. W. (2012). A review of operations research methods applicable to wildfire management. *International Journal of Wildland Fire*, 21(3):189–196. 12

[99] Minas, J. P., Hearne, J. W., and Martell, D. L. (2014). A spatial optimisation model for multi-period landscape level fuel management to mitigate wildfire impacts. *European Journal of Operational Research*, 232(2):412–422. 13, 14

[100] Mississippi Forestry Commission (2017). Forest resource development program. https://www.mfc.ms.gov/FRDP. 10

[101] Moghaddas, J. J. and Craggs, L. (2008). A fuel treatment reduces fire severity and increases suppression efficiency in a mixed conifer forest. *International Journal of Wildland Fire*, 16(6):673–678. 11

[102] Mohammadi, M., Jula, P., and Tavakkoli-Moghaddam, R. (2019). Reliable single-allocation hub location problem with disruptions. *Transportation Research Part E: Logistics and Transportation Review*, 123:90–120. 61

[103] Mokhtari, A., Hassani, H., and Karbasi, A. (2018). Conditional gradient method for stochastic submodular maximization: Closing the gap. In *International Conference on Artificial Intelligence and Statistics*, pages 1886–1895. 120, 129

[104] O'Hanley, J. R., Church, R. L., and Gilless, J. K. (2007a). The importance of in situ site loss in nature reserve selection: balancing notions of complementarity and robustness. *Biological Conservation*, 135(2):170–180. 58

[105] O'Hanley, J. R., Church, R. L., and Gilless, J. K. (2007b). Locating and protecting critical reserve sites to minimize expected and worst-case losses. *Biological Conservation*, 134(1):130–141. 58

[106] O'Hanley, J. R., Scaparra, M. P., and García, S. (2013). Probability chains: A general linearization technique for modeling reliability in facility location and related problems. *European Journal of Operational Research*, 230(1):63–75. 72

[107] Pant, S., Anand, D., Kishor, A., and Singh, S. B. (2015). A particle swarm algorithm for optimization of complex system reliability. *International Journal of Performability Engineering*, 11(1). 119

[108] Papadakos, N. (2006). *Integrated airline scheduling: models and decomposition techniques*. PhD thesis, Imperial College London. 85

[109] Papadakos, N. (2008). Practical enhancements to the magnanti–wong method. *Operations Research Letters*, 36(4):444–449. 85

[110] Peeta, S., Salman, F. S., Gunnec, D., and Viswanath, K. (2010). Pre-disaster investment decisions for strengthening a highway network. *Computers & Operations Research*, 37(10):1708–1719. 5, 62

[111] Peng, P., Snyder, L. V., Lim, A., and Liu, Z. (2011). Reliable logistics networks design with facility disruptions. *Transportation Research Part B: Methodological*, 45(8):1190–1211. 60, 65

[112] Qin, X., Liu, X., and Tang, L. (2013). A two-stage stochastic mixed-integer program for the capacitated logistics fortification planning under accidental disruptions. *Computers & Industrial Engineering*, 65(4):614–623. 58

[113] Quealy, K. (2017). The Cost of Hurricane Harvey: Only One Recent Storm Comes Close. *The New York Times*. 57

[114] Rachmawati, R., Ozlen, M., Reinke, K. J., and Hearne, J. W. (2016). An optimisation approach for fuel treatment planning to break the connectivity of high-risk regions. *Forest Ecology and Management*, 368:94–104. 14

[115] Raghavendra, C. S. and Hariri, S. (1985). Reliability optimization in the design of distributed systems. *IEEE Transactions on software engineering*, 10(1184–1193). 118

[116] Ramamoorthy, P., Jayaswal, S., Sinha, A., and Vidyarthi, N. (2018). Multiple allocation hub interdiction and protection problems: Model formulations and solution approaches. *European Journal of Operational Research*, 270(1):230–245. 59

[117] Rashidi, E. (2016). *Optimization models and algorithms for vulnerability analysis and mitigation planning of pyro-terrorism*. PhD thesis, Mississippi State University. 30

[118] Rashidi, E., Medal, H., Gordon, J., Grala, R., and Varner, M. (2017). A maximal covering location-based model for analyzing the vulnerability of landscapes to wildfires: Assessing the worst-case scenario. *European Journal of Operational Research*, 258(3):1095–1105. 31, 32, 51

[119] Rashidi, E., Medal, H., and Hoskins, A. (2018a). An attacker-defender model for analyzing the vulnerability of initial attack in wildfire suppression. *Naval Research Logistics*, 65(2):120–134. 30

[120] Rashidi, E., Medal, H. R., and Hoskins, A. (2018b). Mitigating a pyro-terror attack using fuel treatment. *IISE Transactions*, 50(6):499–511. 13

[121] Ravi, V., Reddy, P., and Zimmermann, H.-J. (2000). Fuzzy global optimization of complex system reliability. *IEEE Transactions on Fuzzy systems*, 8(3):241–248. 119

[122] Rytwinski, A. and Crowe, K. A. (2010). A simulation-optimization model for selecting the location of fuel-breaks to minimize expected losses from forest fires. *Forest Ecology and Management*, 260(1):1–11. 14

[123] Santos, F. S. P. and Oliveira, F. (2019). An enhanced L-shaped method for optimizing periodic-review inventory control problems modeled via two-stage stochastic programming. *European Journal of Operational Research*, 275(2):677–693. 77, 79

[124] Santoso, T., Ahmed, S., Goetschalckx, M., and Shapiro, A. (2005). A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research*, 167(1):96–115. 77, 84, 85

[125] Sarhadi, H., Tulett, D. M., and Verma, M. (2017). An analytical approach to the protection planning of a rail intermodal terminal network. *European Journal of Operational Research*, 257(2):511–525. 59

[126] Scaparra, M. P. and Church, R. (2012). Protecting supply systems to mitigate potential disaster: a model to fortify capacitated facilities. *International Regional Science Review*, 35(2):188–210. 58

[127] Scaparra, M. P. and Church, R. L. (2008a). A bilevel mixed-integer program for critical infrastructure protection planning. *Computers & Operations Research*, 35(6):1905–1923. 58

[128] Scaparra, M. P. and Church, R. L. (2008b). An exact solution approach for the interdiction median problem with fortification. *European Journal of Operational Research*, 189(1):76–92. 58

[129] Sheikhalishahi, M., Ebrahimipour, V., Shiri, H., Zaman, H., and Jeihoonian, M. (2013). A hybrid ga–pso approach for reliability optimization in redundancy allocation problem. *The International Journal of Advanced Manufacturing Technology*, 68(1-4):317–338. 119

[130] Shishebori, D. and Babadi, A. Y. (2015). Robust and reliable medical services network design under uncertain environment and system disruptions. *Transportation Research Part E: Logistics and Transportation Review*, 77:268–288. 60

[131] Shishebori, D., Karimi-Nasab, M., and Snyder, L. V. (2017). A two-phase heuristic algorithm for designing reliable capacitated logistics networks under disruptions. *European Journal of Industrial Engineering*, 11(4):425–468. 60

[132] Shishebori, D., Snyder, L. V., and Jabalameli, M. S. (2014). A reliable budget-constrained fl/nd problem with unreliable facilities. *Networks and Spatial Economics*, 14(3-4):549–580. 60

[133] Snyder, L. V., Scaparra, M. P., Daskin, M. S., and Church, R. L. (2006). Planning for disruptions in supply chain networks. *Tutorials in Operations Research*, 2:234–257. 58, 60

[134] Stephens, S. L., McIver, J. D., Boerner, R. E., Fettig, C. J., Fontaine, J. B., Hartsough, B. R., Kennedy, P. L., and Schwilk, D. W. (2012). The effects of forest fuel-reduction treatments in the United States. *BioScience*, 62(6):549–560. 11

[135] Sviridenko, M., Vondrák, J., and Ward, J. (2017). Optimal approximation for submodular and supermodular optimization with bounded curvature. *Mathematics of Operations Research*, 42(4):1197–1218. 120

[136] Thompson, M. P. and Calkin, D. E. (2011). Uncertainty and risk in wildland fire management: a review. *Journal of Environmental Management*, 92(8):1895–1909. 13

[137] Thompson, M. P., Vaillant, N. M., Haas, J. R., Gebert, K. M., and Stockmann, K. D. (2013). Quantifying the potential impacts of fuel treatments on wildfire suppression costs. *Journal of Forestry*, 111(1):49–58. 11

[138] Topkis, D. M. (1978). Minimizing a submodular function on a lattice. *Operations research*, 26(2):305–321. 125

[139] Vaillant, N. M., Fites-Kaufman, J. A., and Stephens, S. L. (2009). Effectiveness of prescribed fire as a fuel treatment in californian coniferous forests. *International Journal of Wildland Fire*, 18(2):165–175. 11

[140] Van Slyke, R. M. and Wets, R. (1969). L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663. 77

[141] Vondrák, J. (2008). Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 67–74. 120, 129

[142] Wei, Y. (2012). Optimize landscape fuel treatment locations to create control opportunities for future fires. *Canadian Journal of Forest Research*, 42(6):1002–1014. 14

[143] Wei, Y. and Long, Y. (2014). Schedule fuel treatments to fragment high fire hazard fuel patches. *Mathematical and Computational Forestry & Natural-Resource Sciences*, 6(1):1–10. 14, 16

[144] Wei, Y., Rideout, D., and Kirsch, A. (2008). An optimization model for locating fuel treatments across a landscape to reduce expected fire losses. *Canadian Journal of Forest Research*, 38(4):868–877. 13

[145] Yeh, W.-C., Lin, Y.-C., Chung, Y. Y., and Chih, M. (2010). A particle swarm optimization approach based on monte carlo simulation for solving the complex network reliability problem. *IEEE Transactions on Reliability*, 59(1):212–221. 119

[146] Zhan, Y., Zheng, Q. P., Wang, J., and Pinson, P. (2016). Generation expansion planning with large amounts of wind power via decision-dependent stochastic programming. *IEEE Transactions on Power Systems*, 32(4):3015–3026. 6

[147] Zhu, Y., Zheng, Z., Zhang, X., and Cai, K. (2013). The r-interdiction median problem with probabilistic protection and its solution algorithm. *Computers & Operations Research*, 40(1):451–462. 59

Appendices

A Additional Results of Chapter 2

A.1 Parameters

Table A.1 includes the number of acres owned by each landowner j for landscapes having $|\mathcal{J}|$ =4, 6, 8, 10 total landowners, and Table A.2 shows the dollar amounts for each allocation level k when there are $|\mathcal{K}|$ =2, 4, 6, 8, 10 allocation levels.

A.2 Average Runtime

Table A.3 shows the average runtimes of the risk-based allocation experiments for the Santa Fe landscape. Cells marked with asterisk (*) represent the combinations that cannot reach optimality within 6 hours (21600 seconds) and provide sub-optimal solution at termination.

Tables A.4 and A.5 show the average runtimes of the uniform and hybrid allocation experiments, respectively, for the Santa Fe landscape. Compared to the average runtimes for the risk-based allocation experiments, discussed in Section 2.4.2, the runtimes for uniform and hybrid allocation methods are more consistent across number of cost-share allocation levels $|\mathcal{K}|$, for each number of landowners $|\mathcal{J}|$.

A.3 Impact of Budget on Expected Damage

Table A.6 shows that increasing the budget by \$20,000 from \$20,000 to \$100,000 provides, on average, a 33.18% improvement in the expected damage reduction to the landscape when using risk-based allocation. These values are also displayed in Figure 2.4 in Section 2.4.3. Cells marked with asterisk (*) represent the combinations that cannot reach optimality within 6 hours (21600 seconds). The solutions of these timed-out combinations are sub-optimal for which we use the best upper bound found at termination as expected damage.

As shown in Table A.7, budget increases of \$20,000 from \$20,000 to \$100,000 provide, on average, a 31.33% reduction in the expected damage to the landscape when using uniform allocation. Similarly, Table A.8 shows an average improvement of 33.31% from increasing the budget by \$20,000 from \$20,000 to \$100,000 when using hybrid allocation.

Table A.1: Landowner acreage for each modified landscape.

Landowner j	Number of Landowners $\|\mathcal{J}\|$			
	4	6	8	10
1	584.7	325.2	224.9	179.9
2	539.7	415.1	356.3	294.1
3	498.2	377.1	204.1	193.7
4	539.7	314.8	314.8	207.6
5	—	346.0	238.7	221.4
6	—	384.0	259.5	214.5
7	—	—	242.2	228.3
8	—	—	321.7	166.1
9	—	—	—	179.9
10	—	—	—	276.8

Table A.2: The cost-share assistance amounts associated with each allocation level k, according to the total number of levels $|\mathcal{K}|$.

Level k	Number of Levels $\lvert\mathcal{K}\rvert$				
	2	4	6	8	10
0	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
1	\$39.83	\$13.28	\$7.97	\$5.69	\$4.43
2	—	\$26.55	\$15.93	\$11.38	\$8.85
3	—	\$39.83	\$23.90	\$17.07	\$13.28
4	—	—	\$31.86	\$22.76	\$17.70
5	—	—	\$39.83	\$28.45	\$22.13
6	—	—	—	\$34.14	\$26.55
7	—	—	—	\$39.83	\$30.98
8	—	—	—	—	\$35.40
9	—	—	—	—	\$39.83

Table A.3: Risk-based allocation runtimes (in seconds) by number of landowners $|\mathcal{J}|$, cost-share budget size B, and number of cost-share allocation levels $|\mathcal{K}|$.

$\|\mathcal{J}\|$	B	$\|\mathcal{K}\|$			Average Runtime for $\|\mathcal{J}\|$
		2	4	6	
4	\$20,000	181.83	182.56	182.29	
	\$40,000	184.02	184.43	185.15	
	\$60,000	183.13	183.08	184.04	
	\$80,000	182.87	183.96	183.90	
	\$100,000	181.68	183.70	184.05	
	Average	182.70	183.55	183.89	183.38
6	\$20,000	730.60	733.48	736.28	
	\$40,000	735.81	736.16	739.83	
	\$60,000	736.97	736.80	736.07	
	\$80,000	736.07	733.81	732.81	
	\$100,000	738.11	741.00	735.62	
	Average	735.51	736.25	736.12	735.96
8	\$20,000	2821.89	2815.90	2888.43	
	\$40,000	2804.90	2857.99	3402.12	
	\$60,000	2877.44	2906.68	3284.45	
	\$80,000	2824.73	2849.41	2963.43	
	\$100,000	2825.10	2832.76	2922.51	
	Average	2830.81	2852.55	3092.19	2925.18
10	\$20,000	11558.93	12316.99	*	
	\$40,000	11593.11	17595.83	*	
	\$60,000	11620.34	17066.81	*	
	\$80,000	11677.09	13582.60	*	
	\$100,000	11710.05	12924.61	*	
	Average	11631.91	14697.37	*	16260.70*

Table A.4: Uniform allocation runtimes (in seconds) by number of landowners $|\mathcal{J}|$, budget size B, and number of allocation levels $|\mathcal{K}|$.

$\|\mathcal{J}\|$	B	$\|\mathcal{K}\|$			Average Runtime for $\|\mathcal{J}\|$
		2	4	6	
4	\$20,000	183.92	184.97	184.97	
	\$40,000	183.30	185.80	185.91	
	\$60,000	185.40	186.20	185.24	
	\$80,000	184.72	186.14	186.08	
	\$100,000	185.79	182.75	185.59	
	Average	184.63	185.17	185.56	185.12
6	\$20,000	741.66	740.81	737.86	
	\$40,000	728.32	724.13	725.49	
	\$60,000	729.08	732.72	735.20	
	\$80,000	733.00	732.17	736.18	
	\$100,000	736.73	733.09	736.38	
	Average	733.76	732.59	734.22	733.52
8	\$20,000	2863.46	2791.33	2787.83	
	\$40,000	2787.66	2787.99	2788.61	
	\$60,000	2786.87	2789.77	2786.47	
	\$80,000	2787.82	2787.17	2785.09	
	\$100,000	2782.54	2784.41	2781.10	
	Average	2801.67	2788.13	2785.82	2791.87
10	\$20,000	11383.86	11312.21	11336.02	
	\$40,000	11425.71	11378.79	11459.91	
	\$60,000	11413.46	11424.82	11379.98	
	\$80,000	11376.19	11505.71	11554.96	
	\$100,000	11476.50	11596.44	11588.25	
	Average	11415.14	11443.60	11463.82	11440.85

Table A.5: Hybrid allocation runtimes (in seconds) by number of landowners $|\mathcal{J}|$, budget size B, and number of allocation levels $|\mathcal{K}|$

$\|\mathcal{J}\|$	B	$\|\mathcal{K}\|$			Average Runtime for $\|\mathcal{J}\|$
		2	4	6	
4	\$20,000	180.54	183.39	183.66	
	\$40,000	184.74	184.66	182.32	
	\$60,000	184.52	183.95	184.79	
	\$80,000	183.33	184.27	185.65	
	\$100,000	182.57	185.09	186.21	
	Average	183.14	184.27	184.53	183.98
6	\$20,000	738.48	740.64	741.88	
	\$40,000	744.43	739.76	739.40	
	\$60,000	741.05	738.01	740.64	
	\$80,000	739.94	738.61	739.37	
	\$100,000	741.62	737.78	740.16	
	Average	741.11	738.96	740.29	740.12
8	\$20,000	2891.66	2884.88	2893.03	
	\$40,000	2889.63	2888.29	2893.67	
	\$60,000	2891.90	2887.64	2893.26	
	\$80,000	2891.07	2885.75	2888.84	
	\$100,000	2887.75	2883.89	2893.41	
	Average	2890.40	2886.09	2892.44	2889.64
10	\$20,000	11615.71	11716.40	11868.53	
	\$40,000	11635.37	11717.39	11772.88	
	\$60,000	11677.41	11763.81	11773.85	
	\$80,000	11669.92	11713.65	11779.22	
	\$100,000	11698.73	11707.93	11761.60	
	Average	11659.43	11723.84	11791.22	11724.83

Table A.6: Average percent improvement in expected damage reduction due to budget B for $|\mathcal{J}|$ landowners across $|\mathcal{K}|$ allocation levels using risk-based allocation.

$\|\mathcal{J}\|$	B	$\|\mathcal{K}\|$			Improvement from Previous	Improvement over \$20,000
		2	4	6		
4	\$20,000	149.00	137.75	131.23	—	—
	\$40,000	126.09	91.45	86.52	27.3%	27.3%
	\$60,000	82.33	58.17	55.11	35.7%	53.2%
	\$80,000	53.29	32.39	32.52	39.6%	71.7%
	\$100,000	26.27	26.23	26.40	33.3%	81.1%
6	\$20,000	132.92	123.14	124.18	—	—
	\$40,000	104.92	84.28	84.43	28.0%	28.0%
	\$60,000	60.32	54.55	53.59	38.4%	55.7%
	\$80,000	41.67	32.78	30.82	37.5%	72.3%
	\$100,000	25.32	25.36	25.36	27.8%	80.0%
8	\$20,000	133.24	122.11	119.49	—	—
	\$40,000	96.02	81.95	81.02	30.9%	30.9%
	\$60,000	63.10	52.08	50.20	36.1%	55.9%
	\$80,000	36.17	29.94	29.18	42.4%	74.6%
	\$100,000	24.59	24.59	24.60	22.6%	80.3%
10	\$20,000	123.67	119.11	117.85*	—	—
	\$40,000	89.36	79.05	78.73*	31.5%	31.5%
	\$60,000	56.48	49.55	49.21*	37.2%	57.0%
	\$80,000	32.03	29.03	28.37*	42.4%	75.2%
	\$100,000	23.85	23.83	23.81*	20.1%	80.2%

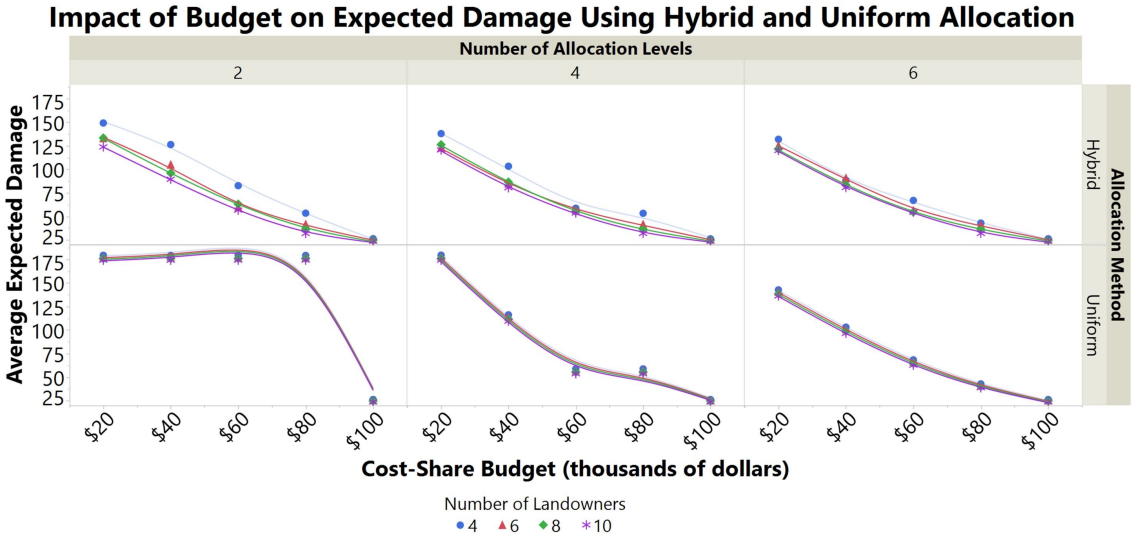


Figure A.1: Impact of budget on expected damage using hybrid and uniform cost-share allocation strategies.

Table A.7: Average percent improvement in expected damage reduction due to budget B for $|\mathcal{J}|$ landowners across $|\mathcal{K}|$ allocation levels using uniform allocation.

$\|\mathcal{J}\|$	B	$\|\mathcal{K}\|$			Improvement from Previous	Improvement over \$20,000
		2	4	6		
4	\$20,000	179.35	179.49	142.65	—	—
	\$40,000	179.14	116.14	103.09	20.6%	20.6%
	\$60,000	179.23	58.80	68.36	23.1%	38.9%
	\$80,000	179.14	58.78	42.84	8.4%	44.0%
	\$100,000	26.31	26.25	26.27	71.9%	84.3%
6	\$20,000	177.41	177.70	140.43	—	—
	\$40,000	177.30	113.72	100.78	20.9%	20.9%
	\$60,000	177.28	56.96	66.31	23.3%	39.3%
	\$80,000	177.25	56.88	41.34	8.3%	44.4%
	\$100,000	25.45	25.32	25.28	72.4%	84.7%
8	\$20,000	175.77	175.90	138.36	—	—
	\$40,000	175.80	111.63	98.69	21.2%	21.2%
	\$60,000	175.85	55.27	64.56	23.4%	39.7%
	\$80,000	175.85	55.29	40.08	8.3%	44.7%
	\$100,000	24.60	24.61	24.61	72.8%	84.9%
10	\$20,000	174.15	174.20	136.25	—	—
	\$40,000	174.17	109.43	96.49	21.6%	21.6%
	\$60,000	174.17	53.65	62.74	23.6%	40.0%
	\$80,000	174.22	53.65	38.78	8.2%	45.0%
	\$100,000	23.81	23.81	23.83	73.2%	85.3%

Table A.8: Average percent improvement in expected damage reduction due to budget B for $|\mathcal{J}|$ landowners across $|\mathcal{K}|$ allocation levels using hybrid allocation.

$\|\mathcal{J}\|$	B	$\|\mathcal{K}\|$			Improvement from Previous	Improvement over \$20,000
		2	4	6		
4	\$20,000	149.11	137.88	131.82	—	—
	\$40,000	126.24	103.20	88.38	24.1%	24.1%
	\$60,000	82.61	58.65	66.96	34.5%	50.3%
	\$80,000	53.42	53.26	42.89	28.2%	64.3%
	\$100,000	26.33	26.21	26.28	47.3%	81.2%
6	\$20,000	132.92	123.12	125.25	—	—
	\$40,000	104.97	84.74	90.72	26.5%	26.5%
	\$60,000	60.00	56.89	56.21	38.3%	54.6%
	\$80,000	41.69	41.61	41.25	28.0%	67.3%
	\$100,000	25.40	25.35	25.36	38.9%	80.0%
8	\$20,000	133.32	126.15	121.36	—	—
	\$40,000	95.93	86.49	83.26	30.2%	30.2%
	\$60,000	63.05	55.28	54.61	34.9%	54.6%
	\$80,000	36.34	36.30	36.38	37.0%	71.4%
	\$100,000	24.62	24.57	24.59	32.3%	80.6%
10	\$20,000	123.77	120.54	120.01	—	—
	\$40,000	89.42	80.41	80.71	31.2%	31.2%
	\$60,000	56.62	53.61	54.58	34.2%	54.8%
	\$80,000	31.91	31.95	31.97	41.9%	73.7%
	\$100,000	23.83	23.83	23.83	25.4%	80.4%

A.4 Cost-Share Allocation Levels in Each Allocation Strategy

Table A.9 demonstrates the cost-share assistance levels allocated to each landowner for each parameter combination in each allocation strategy. Cells marked with asterisk (*) represent the combinations that cannot reach optimality within 6 hours (21600 seconds) and yield the best found sub-optimal solution at termination.

A.5 Results of the Logistic Regression Model

The logistic regression model is used to predict the probability of a landowner to accept a cost-share program for a given amount or level of financial assistance. Let's consider that the minimum and the maximum cost-share amount that an agency can offer to a landowner are \$0 and \$39.82, respectively. Also, assume that the number of allocation levels $|\mathcal{K}| = 4$. The discretized cost-share allocation levels with the corresponding cost-share amounts (U.S. dollar) are shown in Table A.10.

The logistic regression model is trained using the synthetic data containing the cost-share amounts (U.S. dollar) and the landowner's accept/reject decisions. Now the fitted model is used to predict the probability of a landowner's acceptance or rejection for each of the cost-share amounts in Table A.10. The estimated probabilities of accepting or rejecting the cost-share offer for a given dollar amount or the corresponding allocation level are shown in Table A.11.

The probability of a landowner accepting a cost-share offer is used in the stochastic programming model.

Table A.9: Cost-share assistance levels allocated to each landowner in uniform, hybrid, and risk-based allocation strategies, where B is budget, $|\mathcal{K}| = 6$ is the number of allocation levels, $|\mathcal{J}|$ is the number of landowners.

B	$\lvert\mathcal{J}\rvert$	Uniform Allocation	Hybrid Allocation	Risk-Based Allocation
\$20,000	4	[1, 1, 1, 1]	[2, 2, 0, 0]	[1, 3, 0, 0]
	6	[1, 1, 1, 1, 1, 1]	[0, 3, 3, 0, 0, 0]	[0, 4, 2, 0, 0, 0]
	8	[1, 1, 1, 1, 1, 1, 1, 1]	[3, 3, 3, 0, 0, 0, 0, 0]	[2, 4, 3, 0, 0, 0, 0, 0]
	10	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1]	[3, 3, 3, 0, 0, 0, 0, 3, 0, 0]	[2, 3, 3, 3, 0, 0, 0, 0, 0, 0]*
\$40,000	4	[2, 2, 2, 2]	[3, 3, 0, 3]	[4, 4, 1, 0]
	6	[2, 2, 2, 2, 2, 2]	[4, 4, 4, 0, 0, 0]	[4, 4, 2, 3, 1, 0]
	8	[2, 2, 2, 2, 2, 2, 2, 2]	[3, 3, 3, 3, 0, 0, 3, 3]	[3, 4, 4, 2, 3, 0, 3, 0]
	10	[2, 2, 2, 2, 2, 2, 2, 2, 2, 2]	[3, 3, 3, 3, 3, 3, 0, 3, 3, 0]	[3, 4, 4, 4, 1, 3, 0, 0, 0, 3]*
\$60,000	4	[3, 3, 3, 3]	[4, 4, 0, 4]	[3, 4, 4, 3]
	6	[3, 3, 3, 3, 3, 3]	[4, 4, 4, 0, 4, 4]	[5, 4, 3, 4, 2, 3]
	8	[3, 3, 3, 3, 3, 3, 3, 3]	[4, 4, 4, 4, 4, 4, 4, 0]	[4, 4, 5, 4, 3, 2, 3, 3]
	10	[3, 3, 3, 3, 3, 3, 3, 3, 3, 3]	[4, 4, 4, 4, 4, 4, 4, 0, 0, 4]	[4, 5, 4, 4, 3, 4, 2, 4, 3, 2]*
\$80,000	4	[4, 4, 4, 4]	[4, 4, 4, 4]	[5, 5, 4, 4]
	6	[4, 4, 4, 4, 4, 4]	[4, 4, 4, 4, 4, 4]	[5, 5, 5, 5, 5, 3]
	8	[4, 4, 4, 4, 4, 4, 4, 4]	[5, 5, 5, 5, 5, 0, 5, 5]	[5, 5, 5, 5, 4, 4, 5, 4]
	10	[4, 4, 4, 4, 4, 4, 4, 4, 4, 4]	[5, 5, 5, 5, 5, 5, 5, 5, 0, 5]	[5, 5, 5, 4, 5, 5, 4, 4, 4, 5]*

Table A.10: Cost-share amount and the corresponding allocation levels

Allocation level (k)	Cost-share amount (c_k)
0	\$0.00
1	\$13.28
2	\$26.55
3	\$39.83

Table A.11: Estimated probabilities of accepting the cost-share program

Cost-share amount (c_k)	Allocation level (k)	Probability of Acceptance	Probability of Rejection
0	0	0.273	0.727
13.28	1	0.524	0.476
26.55	2	0.765	0.235
39.83	3	0.905	0.095

B Mean Value Model of Chapter 3

The mean value model is presented as follows:

$$min \quad \sum_{(i,j)\in\mathcal{A}} t_{ij} f_{ij} \tag{1a}$$

$$s.t. \quad \sum_{k\in\mathcal{K}} y_{zk} = 1 \qquad \forall z \in \mathcal{N}_{\mathcal{F}} \tag{1b}$$

$$\sum_{\ell\in\mathcal{L}} C_\ell x_\ell + \sum_{z\in\mathcal{N}_{\mathcal{F}}} \sum_{k\in\mathcal{K}} C_k y_{zk} \leq B \tag{1c}$$

$$f_{ij} \leq U_{ij} x_{\ell(i,j)} \qquad \forall (i,j) \in \mathcal{A} \tag{1d}$$

$$\sum_{(i,j)\in\mathcal{RS}(n)} f_{ji} - \sum_{(i,j)\in\mathcal{FS}(n)} f_{ij} = d_n \qquad \forall n \in \mathcal{N}_{\mathcal{D}} \tag{1e}$$

$$\sum_{(i,j)\in\mathcal{FS}(z)} f_{ij} - \sum_{(i,j)\in\mathcal{RS}(z)} f_{ji} \leq \sum_{k\in\mathcal{K}} \left(\mathbb{E}\left[a_{zs} | y_{zk} = 1\right] y_{zk}\right) \qquad \forall z \in \mathcal{N}_{\mathcal{F}} \tag{1f}$$

$$\sum_{(i,j)\in\mathcal{FS}(D)} f_{ij} \leq C_D \tag{1g}$$

$$y_{zk} \in \{0,1\} \qquad \forall z \in \mathcal{N}_{\mathcal{F}}, k \in \mathcal{K} \tag{1h}$$

$$x_\ell \in \{0,1\} \qquad \forall \ell \in \mathcal{L} \tag{1i}$$

$$f_{ij} \geq 0 \qquad \forall (i,j) \in \mathcal{A} \tag{1j}$$

$$\mathbb{E}\left[a_{zs} | y_{zk} = 1\right] = \textstyle\sum_{\omega\in\Omega} \mathbb{P}_h^\omega \mathbb{P}^\omega_{zs(z,\omega)m(z,\omega)k} a_{zs}^\omega$$

C Numerical Example of the Probability-Chain Reformulation

We consider the small example shown in Section 3.2 as well as the probabilities demonstrated in Table 3.1.

We assume that 1 unit of protection is allocated to the first facility and no protection is allocated to the second facility, i.e., $y_{11} = 1$, $y_{20} = 1$. We also consider that the PDTC in a given scenario ω is 4000 units where both facilities have a PDCS of 0. According to Eqs. (3.7), we have:

first facility:

$$q^{\omega}_{11} \leq 4000 y_{11}$$

second facility:

$$q^{\omega}_{20} \leq 4000 y_{20}$$

The PPWPDTC corresponding to the first facility in scenario ω is computed using the recursive Eq. (3.5) as follows:

$$q^{\omega}_{11} = \mathbb{P}^{\omega}_{1,0,0,1} y_{11} \times 4000 \quad = 0.15 \times 4000 = 600$$

Using the value of q^{ω}_{11} in the recursive Eqs. (3.6), we compute the PPWPDTC upto the second facility in scenario ω as follows:

$$q^{\omega}_{11} = \frac{1}{\mathbb{P}^{\omega}_{2,0,0,0}} \times q^{\omega}_{20} \quad \Rightarrow q^{\omega}_{20} = 0.90 \times 600 = 540$$

Therefore, the probability-weighted transportation cost in scenario ω is 540.

D Test Networks of Chapter 3

D.1 16-Node Networks

This section demonstrates the changes in the optimal solution as the decision maker's budget varies for a 16-node networks with 4 facilities. This section also presents the variation of the expected fraction of total demand as budget changes. Figure D.2 demonstrates the optimal link construction and protection investment decisions for the 16-node network with \$10,000 budget, number of protection levels, $|\mathcal{K}| = 3$ and number of capacity states, $|\mathcal{S}| = 3$. The number inside each node is the index of that node. The name of the city represented by each node and the corresponding demand of that node (inside parenthesis) are given adjacent to each node. The numbers above each link inside the parenthesis represent the index and transportation cost of per unit flow through that link, respectively. The numbers inside the parenthesis next to each facility represent the capacity of that facility prior to its exposure to a disruption and the protection level (k) invested to that facility, respectively. The solid and dashed lines between nodes represent the constructed and unconstructed links in the optimal solution, respectively.

Figure D.2 shows that given this small amount of budget, the decision maker can construct few links in the network and is unable to invest any protection to the facilities. It is to be noted that, given a small amount of budget, the primary concern is to construct links in the network, allowing transportation of commodities. The total demand of this 16-node network is 570.59 units. It is found that the optimal solution with this \$10,000 budget satisfies an expected demand of 308.80 units, which is 54.12% of the total demand.

Figure D.3 demonstrates the optimal link construction and protection investment decisions for the 16-node network with \$40,000 budget, number of protection levels, $|\mathcal{K}| = 3$ and number of capacity states, $|\mathcal{S}| = 3$. Unlike the solution with \$10,000 budget, after opening some links in the network, the optimal solution with \$40,000 budget invests some protection resources to the facilities that are exposed to high level of disruption intensity. The expected fraction of the total demand satisfied by the optimal solution with \$40,000 budget is found to be 95.4%.

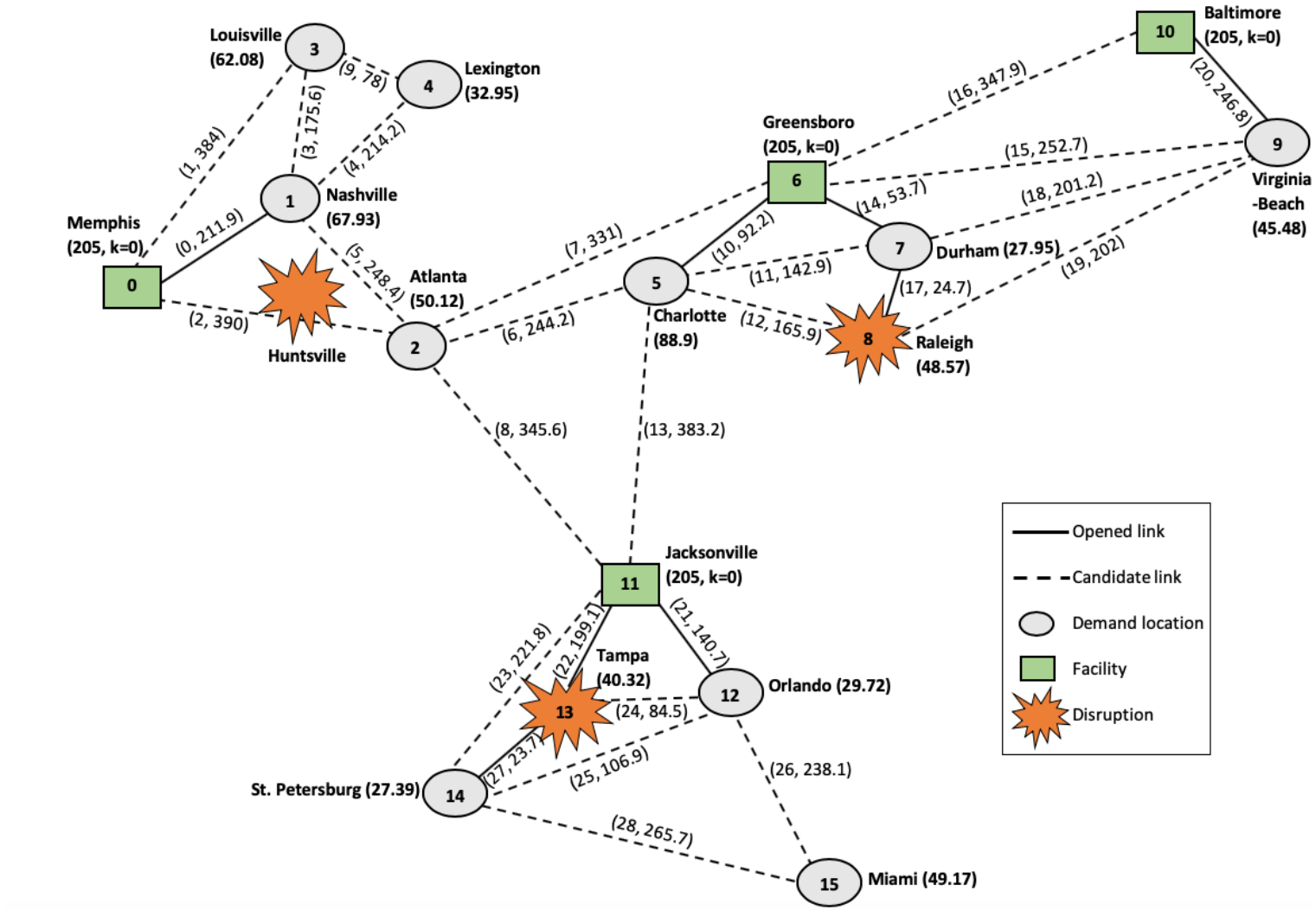


Figure D.2: 16-node network with $10,000 budget

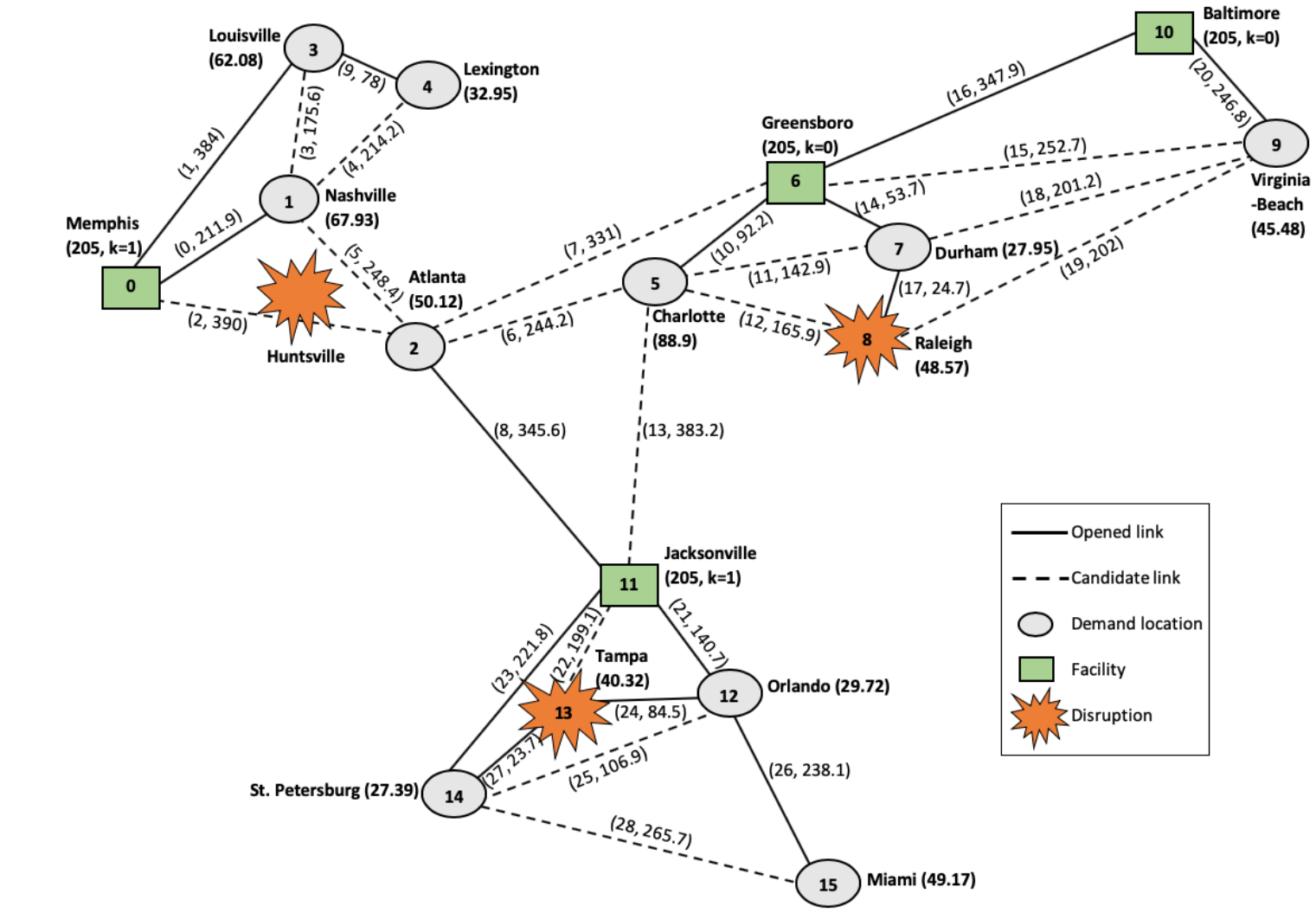


Figure D.3: 16-node network with $40,000 budget

Figure D.4 demonstrates the optimal solution for the 16-node network with $110,000 budget, number of protection levels, $|\mathcal{K}| = 3$ and number of capacity states, $|\mathcal{S}| = 3$. With this large amount of budget, the decision maker can open enough links to transport commodities to all the demand locations as well as invest highest level of protection resources to all the facilities. Therefore, the expected fraction of the total demand satisfied by the optimal solution with $110,000 budget is found to be 100%.

D.2 25-Node Network

Figure D.5 demonstrates a 25-node network with 4 facilities used in the computational experiments. The optimal solution for this 25-node network with $40,000 budget, number of protection levels, $|\mathcal{K}| = 3$ and number of capacity states, $|\mathcal{S}| = 3$ is also demonstrated in Figure D.5.

D.3 32-Node Network

Figure D.6 demonstrates a 32-node network with 4 facilities used in the computational experiments. Figure D.6 also shows the optimal solution with $120,000 budget, number of protection levels, $|\mathcal{K}| = 3$ and number of capacity states, $|\mathcal{S}| = 3$.

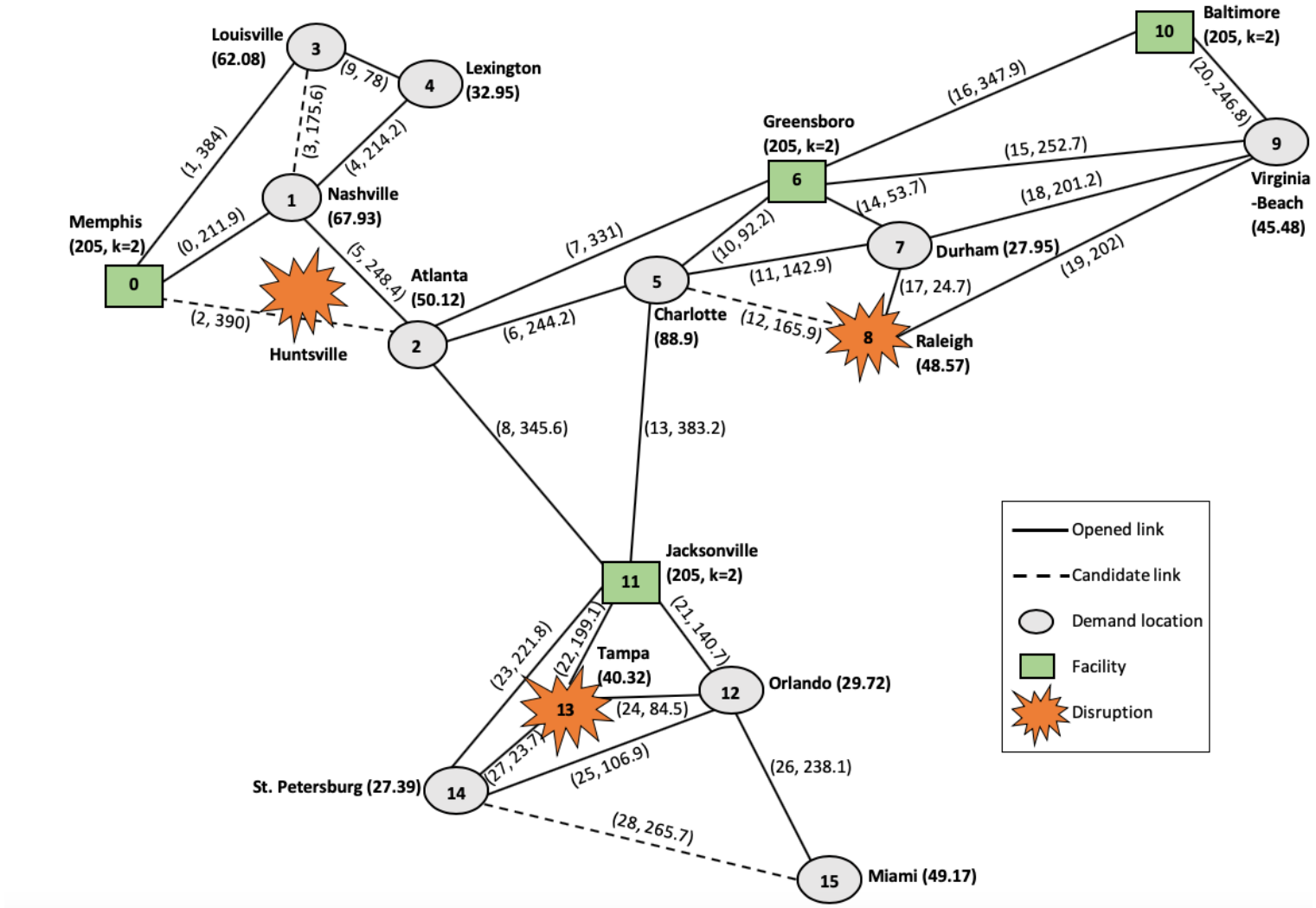


Figure D.4: 16-node network with $110,000 budget

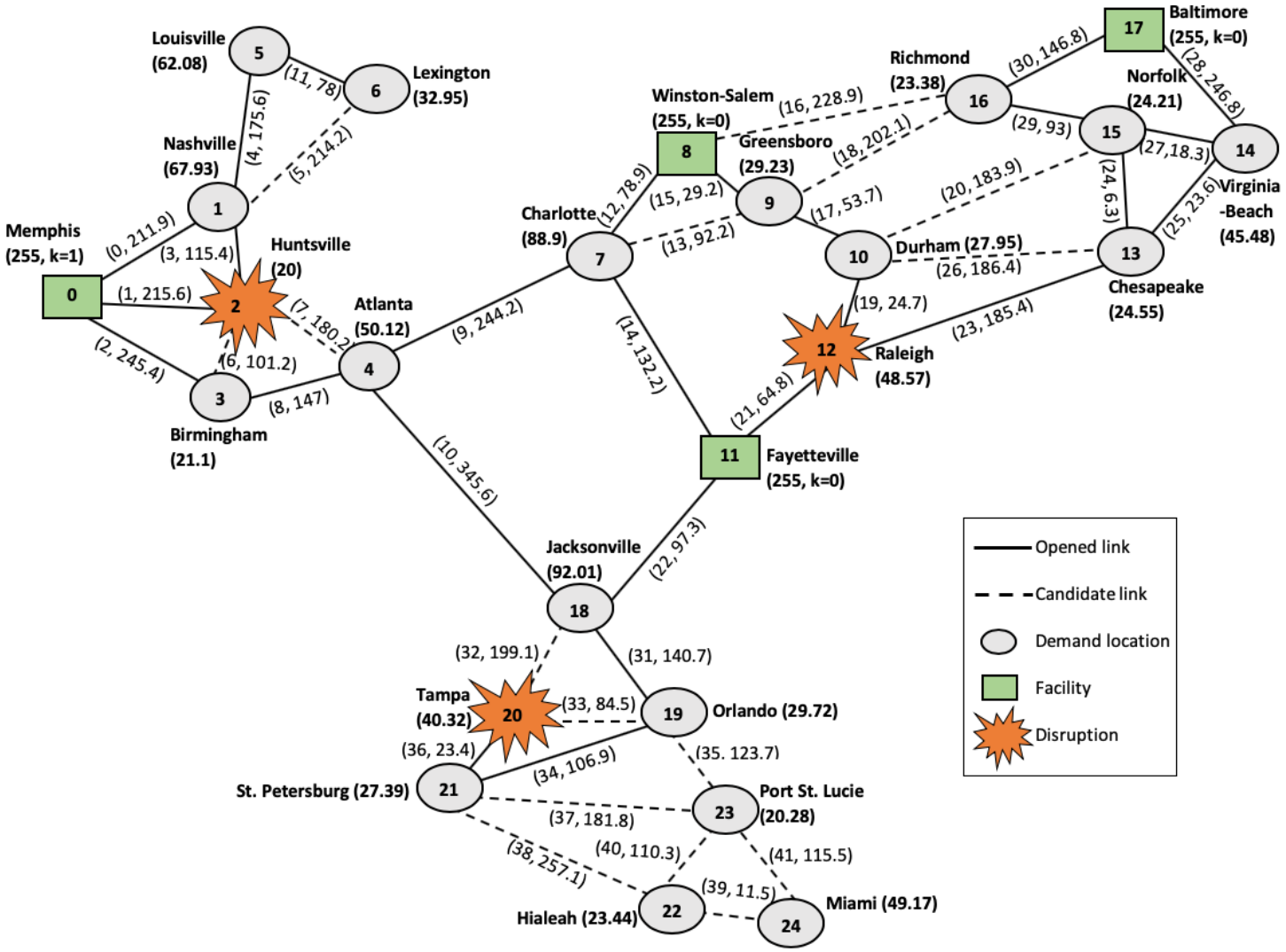


Figure D.5: A 25-node network

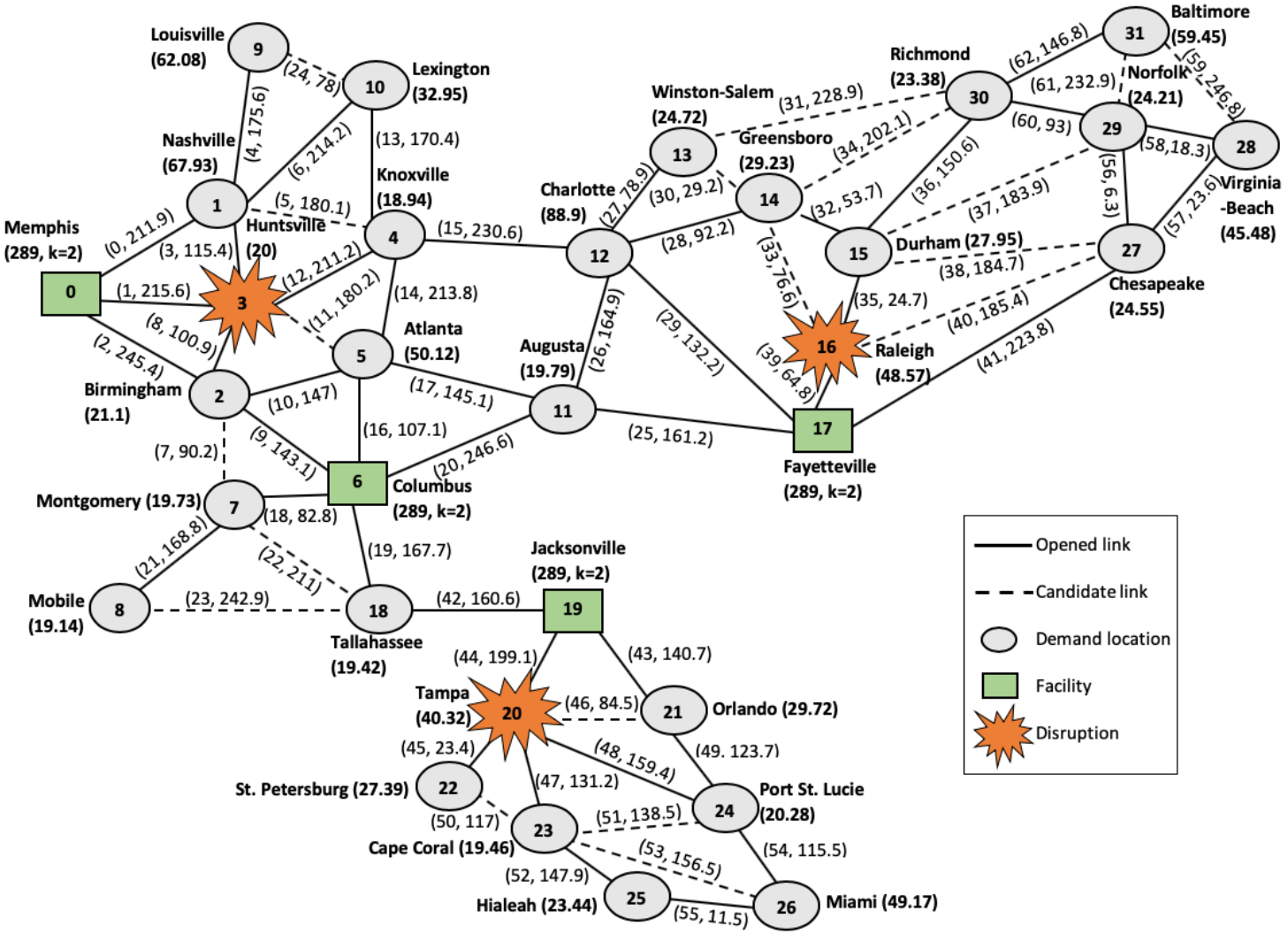


Figure D.6: A 32-node network.

E Runtime of the Accelerated L-shaped Decomposition Algorithm

Table E.12 demonstrates some of the runtime results for 25- and 32- node networks with 4 and 6 facilities and different number of protection levels and capacity states. The "Runtime" and "Gap" columns represent the runtime and the optimality gap of the algorithm at termination after 2 hours, respectively.

Table E.12: Runtime of the accelerated L-shaped algorithm (*MCut* + *TR* + *nLimit* + *VI* + *PCut*)

Network Size, $\lvert\mathcal{N}\rvert$	$\lvert\mathcal{N}_{\mathcal{F}}\rvert$	B	$\lvert\mathcal{S}\rvert$	$\lvert\mathcal{K}\rvert$	$\lvert\Omega\rvert$	**Runtime**	**Gap(%)**
25	4	30000	3	3	324	351.73	0
25	4	30000	3	4	324	490.41	0
25	4	30000	4	3	1024	1262.08	0
25	4	30000	4	4	1024	1582.73	0
25	4	70000	3	3	324	425.66	0
25	4	70000	3	4	324	651.62	0
25	4	70000	4	3	1024	1498.31	0
25	4	70000	4	4	1024	2034.98	0
25	4	90000	3	3	324	318.48	0
25	4	90000	3	4	324	516.53	0
25	4	90000	4	3	1024	1260.24	0
25	4	90000	4	4	1024	1681.64	0
25	6	30000	3	3	2916	3687.68	0
25	6	30000	3	4	2916	5982.02	0
25	6	30000	4	3	16384	*	100
25	6	30000	4	4	16384	*	100
25	6	70000	3	3	2916	*	44.98
25	6	70000	3	4	2916	*	88.48
25	6	90000	3	3	2916	4021.31	0
25	6	90000	3	4	2916	6402.82	0
25	6	90000	4	3	16384	*	100
25	6	90000	4	4	16384	*	100
32	4	30000	3	3	324	652.34	0
32	4	30000	3	4	324	773.85	0
32	4	30000	4	3	1024	1872.57	0
32	4	30000	4	4	1024	2459.05	0
32	4	70000	3	3	324	924.47	0
32	4	70000	3	4	324	1383.31	0
32	4	70000	4	3	1024	2261.78	0
32	4	70000	4	4	1024	2946.77	0
32	4	90000	3	3	324	680.03	0
32	4	90000	3	4	324	831.49	0
32	4	90000	4	3	1024	1958.67	0
32	4	90000	4	4	1024	2513.24	0
32	6	30000	3	3	2916	*	16.51
32	6	30000	3	4	2916	*	28.87
32	6	30000	4	3	16384	*	100

F A 32-Node Test Network of Chapter 4

Figure F.7 shows a 32-node water supply network with eight sensors. Figure F.7 also demonstrates the solution provided by the continuous greedy algorithm for this 32-node network with 25 units budget, 250 miles coverage radius, 80 iterations, and 60% minimum required demand coverage.

The number inside each node is the index of that node. The name of the city represented by each node and the corresponding demand of that node (inside parenthesis) are given adjacent to each node. The number above each link represents the great-circle distance between the two end nodes of that link. The number inside the parenthesis next to each sensor shows the failure rate (λ) determined by the algorithm.

We know that the goal of our model is to optimally invest resources to the sensors or equivalently to reduce the failure rate of the sensors to enhance their lifetime. As more resources are invested in the sensors, and their failure rate decreases, their survival probability increases. The more sensors survive the time horizon T, the more demand can be covered. This eventually increases the reliability—probability that the minimum required demand is covered. When the budget is not large enough to invest a fair amount of resources to all sensors or reduce all the sensors' failure rate, the model tries to minimize the failure rate of the critical sensors that cover large demand. We see from Figure F.7 that with a 25 unit budget, the sensors' have different failure rates in the final solution. As the 25 unit budget is not large enough to reduce the failure rate of all the sensors equally, the model prioritizes enhancing the lifetime of the critical sensors located in Huntsville ($\lambda = 0.16$), St. Petersburg ($\lambda = 0.70$), and Virginia Beach ($\lambda = 0.20$). Because these sensors cover a larger amount of demand than the other sensors, their survival during the time horizon is crucial to maximize reliability. On the other hand, the sensor located in Jacksonville has the highest failure rate ($\lambda = 73.18$). Because this sensor covers the least amount of demand and therefore gets the least priority in enhancing its lifetime.

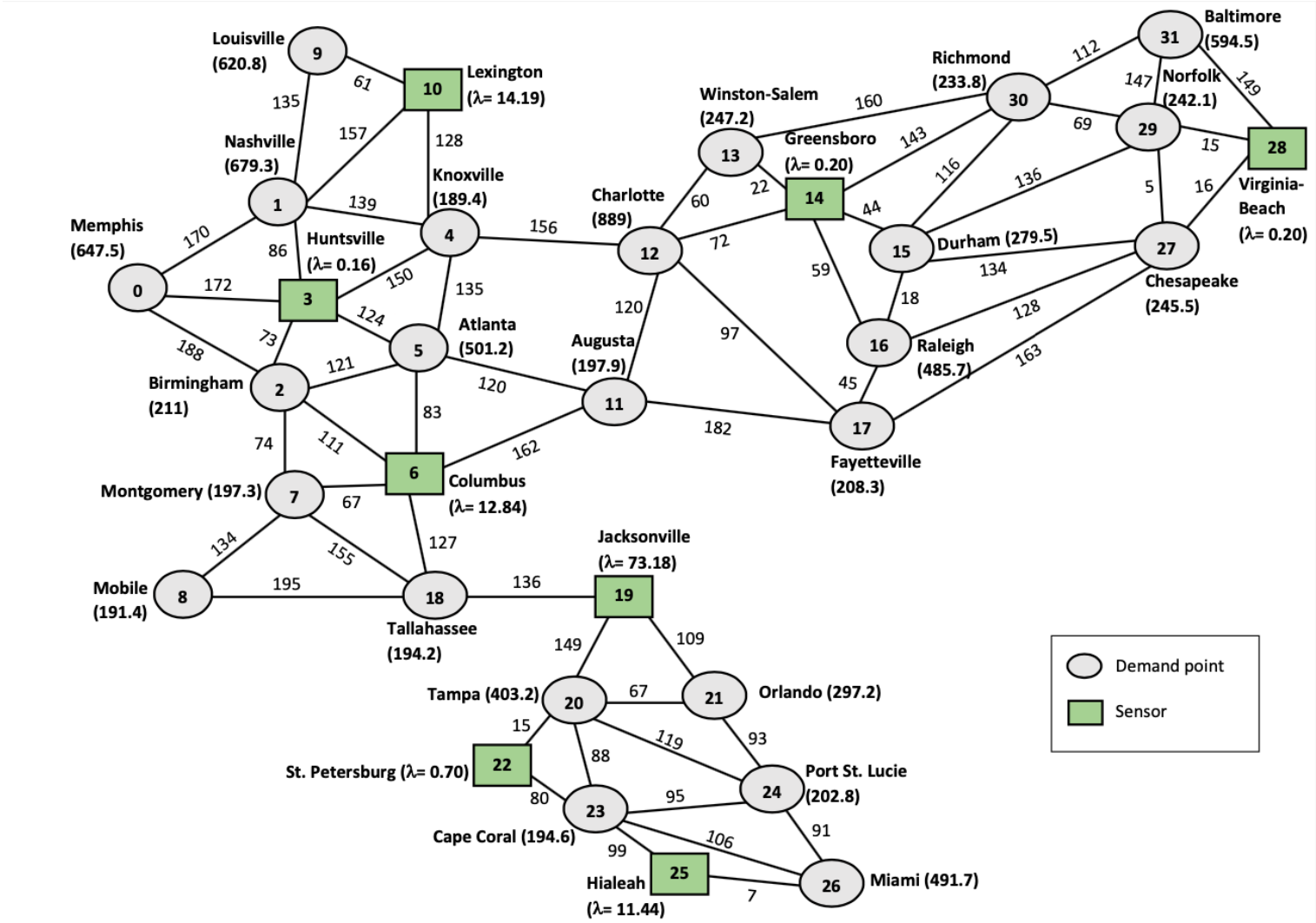


Figure F.7: Solution for a 32-node network

Vita

Tanveer Hossain Bhuiyan is originally from Dhaka, Bangladesh. He graduated from Bangladesh University of Engineering and Technology in Spring 2012 with a B.S. degree in Industrial Engineering. He received M.S. degree in Industrial Engineering from the same institution in June 2014. Later, Tanveer worked as a Lecturer and Assistant Professor in the same institution until Fall 2015, teaching *Introduction to Operations Research* and *Industrial Management* courses. Tanveer moved to the U.S. in August 2015 and received a M.S. degree in Industrial Engineering with a Minor in Statistics from Mississippi State University in Fall 2018. While studying at Mississippi State University, Tanveer also worked as an intern at the Pacific Northwest National Laboratory in the Summer of 2017 and 2018. Tanveer started as a Ph.D. student in the Industrial Engineering Department at the University of Tennessee, Knoxville in Fall 2018. He worked as a Graduate Research Assistant and Graduate Teaching Associate, teaching *Supply Chain Engineering* Course, in the Industrial Engineering Department. Tanveer worked as an intern in the Energy Systems Division at Argonne National Laboratory in Summer 2020.

Tanveer's Ph.D. research is on developing methodologies for data-driven optimization under endogenous uncertainty. Tanveer's research methodology includes data-driven stochastic programming, integer programming, statistical-learning-based optimization, and machine learning. His research has been applied to the areas of critical infrastructure protection, transportation network design, disaster mitigation, supply chain risk mitigation, cyber and information security, and power system expansion and operations planning with renewable energy integration. Tanveer's research has been published in the *European Journal of Operational Research*, *International Journal of Critical Infrastructure Protection*, and several *IEEE conference proceedings*.