Chance-Constrained Stochastic Programming Models for Humanitarian Relief Network Design

Özgün Elçi

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of the requirements for the degree of
Master of Science

Sabancı University

August, 2016

CHANCE-CONSTRAINED STOCHASTIC PROGRAMMING MODELS FOR HUMANITARIAN RELIEF NETWORK DESIGN

APPROVED BY:

Assoc. Prof. Dr. Nilay Noyan Bülbül .....
(Thesis Supervisor)

Assoc. Prof. Dr. Kerem Bülbül .....
(Thesis Supervisor)

Prof. Dr. Bülent Çatay .....

Assoc. Prof. Dr. Güvenç Şahin .....

Assoc. Prof. Dr. Z. Caner Taşkın .....

Date of Approval: 10.08.2016

Acknowledgements

I believe that a successful academic work rarely occurs without the support of so many people. Therefore, I would like to mention a few names in this page who supported me during my master's studies.

I do not remember when my mother taught me how to count, but I do recall my father taking to me to the extra math lectures when I was 10. Both of them have sacrificed a lot for me and I will be forever in their debt. I thank to my parents and my brother for their unending love and support.

I consider myself very privileged to have the opportunity to work with Nilay Noyan and Kerem Bülbül. They were so generous in teaching and advising me. This thesis would have never existed without their guidance and help. I will certainly miss their professional and personal support when I leave this university.

My flatmates; especially Umut, Paolo, Jackson, and Kıvanç have provided me with a perfect study environment and supported me a lot while I was working at home. Umut, Baran and Metin, I started this journey with you, and your support means a lot to me. Gizem, Özenç, Can, Baha, Betül and Ege, the years we spent together were certainly not enough. Thank you for the amazing four years we spent together, I always felt your support even when I was far away. My fellow graduate students were one of the biggest supporters of my master's studies. Murat, Gizem, Can, İhsan and Yağmur, I missed the duo and the lake already. Sinem and Zeynep, thank you for the cute notes and always making me smile. I thank all of my friends who helped me during my most stressful times while writing this thesis; Merve, Sonya, Ece, Amir, Hazal, Umut, Neda, Soner and many others. And especially Bahar, I will miss your help and kindness.

I feel very happy to be part of the IE community in Sabanci University and I would also like to extend my gratitude to all professors whom I had a chance to attend their courses; Ami Arbel, Aydın Aytuna, Barış Balcıoğlu, İlker Birbil, Nilay Noyan and Tonguç Ünlüyurt. I am also grateful to the jury members Bülent Çatay, Güvenç Şahin and Caner Taşkın, for their valuable comments and feedbacks. I am sincerely grateful to my advisors back in Bahçeşehir University, Semra Ağralı and Ethem Çanakoğlu, for introducing me to this field and encouraging me to follow an academic career.

I would like to thank The Scientific and Technological Research Council of Turkey (TUBITAK) BIDEB program for supporting me during master's studies. I also acknowledge the support from TUBITAK under grants # 112M864 and # 115M560. And lastly, I am sincerely grateful to J. Luedtke for providing the data sets for the probabilistic transportation problem.

İnsani Yardım Müdahale Ağı Tasarımı İçin Olasılıksal Kısıt İçeren Rassal Programlama Modelleri

Özgün Elçi

Endüstri Mühendisliği, Yüksek Lisans Tezi, 2016

Tez Danışmanları: Nilay Noyan Bülbül ve Kerem Bülbül

Anahtar Kelimeler: rassal programlama; riskten kaçınma; olasılıksal kısıtlar; değişken güvenilirlik; riske maruz değer; koşullu riske maruz değer; ağ tasarımı; insani yardım lojistiği; erişilebilirlik; eşitlik; Benders ayrıştırması; dal-ve-kesi

Özet

Taleplerin karşılanma olasılığı veya bir sistemin güvenilirliği gibi hususları dikkate alan çok sayıdaki mühendislik uygulamaları olasılıksal kısıt içeren matematiksel programlama modellerinin ortaya çıkmasına sebep olmuştur. Tezin ilk bölümünde, sadece sağ taraf vektörünün rassallık içerdiği ve bu rassal vektörün sonlu bir dağılıma sahip olduğu durumlarda ayrık olasılıksal kısıtlar içeren eniyileme modelleri üzerinde durulmaktadır. Yakın zamanda ortaya konulan, olasılıksal kısıtlara ilişkin güvenilirlik seviyelerini / risk toleranslarını karar değişkeni olarak ele alan ve gerçek maliyet / gelir ile belirlenmiş güvenilirliğe ilişkin maliyet arasındaki ödünleşimi amaç fonksiyonunda gözeten bir sınıf model için güçlü karışık tamsayılı programlama formulasyonları geliştirdik. Ayrıca, risk toleranslarına ilişkin olarak alternatif bir maliyet fonksiyonu tanımladık; bu tip bir fonksiyon değişken bir güvenilirlik seviyesine ilişkin riske maruz değeri (RMD) ifade edebilmeyi gerektirir. Bu işi RMD'nin yeni bir karmaşık sayılı doğrusal programlama gösterimini kullanarak başardık. Yapılan sayısal çalışma geliştirilen matematiksel programlama formülasyonlarımızın etkinliğini göstermektedir. Ayrıca, önerilen modelleme yaklaşımı yeni bir rassal afet sonrası müdahale ağı tasarımı problemi için uygulanmış ve bir vaka çalışması için sayısal sonuçlar sunulmuştur. Tezin ikinci bölümünde ise afet sonrası talep miktarlarındaki ve ulaşım ağı koşullarındaki belirsizlikler altında olan rassal

afet öncesi müdahale ağı tasarımı problemi üzerinde durulmaktadır. Bu problem için riskten kaçınan olasılıksal kısıt içeren bir iki-aşamalı rassal programlama modeli geliştirdik; bu model ortalama değer ve bir risk ölçütüne dayalı bir amaç fonksiyonu ve ikinci aşama probleminin olurluluğuna ilişkin olarak bir ortak olasılıksal kısıt içermektedir. Bu rassal eniyileme modeli için Benders ayrıştırmasına dayalı pekin bir dal-ve-kesi çözüm algoritması geliştirilmiştir ve ayrıntılı sayısal analizimiz çözüm algoritmasının bilgisayımsal etkinliğini göstermektedir.

Chance-Constrained Stochastic Programming Models for Humanitarian Relief Network Design

Özgün Elçi

Industrial Engineering, Master's Thesis, 2016

Thesis Supervisors: Nilay Noyan Bülbül and Kerem Bülbül

Keywords: stochastic programming; risk-aversion; probabilistic constraints; variable reliability; value-at-risk; conditional-value-risk; network design; humanitarian logistics; accessibility; equity; Benders decomposition; branch-and-cut

Abstract

Many engineering applications concerned with issues such as the probability of meeting demand or the reliability of a system give rise to mathematical programming models that involve chance (or probabilistic) constraints. First, we focus on optimization models involving individual chance constraints, in which only the right-hand side vector is random with a finite distribution. We develop strong mixed-integer programming formulations for a recently introduced class of chance-constrained models which treats the reliability levels/ risk tolerances associated with the chance constraints as decision variables and trades off the actual cost / return against the cost of the selected reliability levels in the objective function. In addition, we introduce an alternate cost function type associated with the risk tolerances which requires capturing the value-at-risk (VaR) associated with a variable reliability level. We accomplish this task via a new integer linear programming representation of VaR. Our computational study illustrates the effectiveness of our mathematical programming formulations. We also apply the proposed modeling approach to a new stochastic post-disaster relief network design problem and provide numerical results for a case study. Second, we consider a stochastic pre-disaster relief network design problem in which there is uncertainty in post-disaster demands and transportation network conditions. We develop a risk-averse two-stage chance-constrained stochastic programming model which features a mean-risk objective, and a joint probabilistic constraints enforced on the feasibility of the second-stage problem. We employ an exact Benders decomposition-based branch-and-cut algorithm and our extensive numerical analysis demonstrates the computational effectiveness of the solution algorithm.

Contents

1 Introduction **1**

1.1 Chance-Constrained Single-Stage Stochastic Programming Models 2

1.2 Chance-Constrained Two-Stage Stochastic Programming Models 5

1.3 Outline 7

2 Chance-Constrained Single-Stage Stochastic Programming Models **8**

2.1 Optimization Models with Individual Chance Constraints 9

2.2 Optimization Models with Individual Chance Constraints Under Variable Reliability Levels 16

2.2.1 Balancing the Actual Return/Cost and Risk 16

2.2.2 A New Approach to Quantifying the Cost of Reliability 18

2.3 A Stochastic Optimization Model for Designing Post-Disaster Relief Networks 21

2.3.1 Literature Review 21

2.3.2 Problem Description 23

2.3.3 Stochastic Optimization Model 25

2.4 Computational Study 29

2.4.1 Generation of the Problem Instances 29

2.4.2 Computational Performance 30

2.4.3 Model Analysis 36

3 Chance-Constrained Two-Stage Stochastic Programming Models **41**

3.1 Literature Review 41

3.2 A Stochastic Optimization Model for Designing Pre-Disaster Relief Networks 43

3.2.1 Mathematical Model and Formulations 45

3.3 Benders Decomposition-Based Branch-and-Cut Algorithm 52

3.3.1 Optimality Cuts 54

3.3.2 Feasibility Cuts 58

3.3.3 Computational Enhancements and Implementation Details 61

3.4 Computational Study 63

3.4.1 Generation of the Problem Instances 63
3.4.2 Computational Performance of the Solution Method 64

4 Conclusion and Future Work **69**

List of Figures

2.1 Empirical cumulative distribution functions of MPUD and APUD ($\tau = 0.02, c = 2.0$). 38
2.2 Illustrative Example. 39

List of Tables

2.1 Summary of the MIP formulations of the chance-constrained optimization models. . . . 22
2.2 Computational performance of the alternate MIPs for $(\mathbf{MICC - VRT})$ on the TR instances. . . . 31
2.3 Computational performance of the alternate MIPs for $(\mathbf{MICC - VRT})$ on the RND instances. . . . 32
2.4 Computational performance of the alternate VaR representations on the TR instances. . . . 32
2.5 Computational performance of the alternate VaR representations on the RND instances. . . . 33
2.6 Computational performance of the alternate MIPs for $(\mathbf{MICC - VRTQ})$ on the TR instances. . . . 34
2.7 Computational performance of the alternate MIPs for $(\mathbf{MICC - VRTQ})$ with (2.20) on the RND instances. . . . 35
2.8 Values of the objective function terms at optimality for varying parameter settings ($\bar{\epsilon} = 0.3$). . . . 36
2.9 Summary of the attained risk tolerances (τ = 0.01 and c = 2). . . . 37
2.10 Optimal solutions of the illustrative example. . . . 40

3.1 Computational performances of the $(\mathbf{DEF})$ and the branch-and-cut algorithm for varying ϵ, λ and α where $|J| = 10$. . . . 65
3.2 Computational performances of the $(\mathbf{DEF})$ and the branch-and-cut algorithm for varying ϵ, λ and α where $|J| = 30$. . . . 66
3.3 Impact of Preprocessing on Chance-Constrained Risk-Neutral Instances ($\lambda = 0$). . . . 67
3.4 Impact of Preprocessing on Risk-Averse Instances ($\lambda = 0.5$, $\alpha = 0.7$). . . . 68

Chapter 1

Introduction

In many decision making problems under uncertainty it is desired to account for the probability of certain unfavorable events. This is very much in agreement with the concept of reliability often used to characterize the quality of service. In a wide range of domains, applications concerned with such issues (the probability of meeting demand or the reliability of a system) give rise to mathematical programming models that involve chance (or probabilistic) constraints. Areas of application of chance-constrained optimization models include but are not restricted to network design, supply chain management, and disaster management. For a thorough overview of the applications of chance-constrained optimization models and the corresponding theory and numerical methods, we refer the reader to Kall and Wallace (1994), Prékopa (1995), Prékopa (2003), Dentcheva (2006), Shapiro et al. (2009), and the references therein.

There are two main types of chance constraints: joint and individual (separate) chance constraints. A joint chance constraint imposes that a set of goal constraints (inequalities) hold together with a high probability. In contrast, an individual chance constraint is introduced to account for the probability of a single goal constraint. From a modeling point of view, the problem of interest determines the appropriate type of chance constraint. As discussed in Haneveld and van der Vlerk (2015), a joint chance constraint is more fitting when the individual goal constraints collectively describe one single goal. Otherwise, if the individual goal constraints describe different goals, it makes more sense to consider them separately. In this case, the ability to vary the reliability levels associated with the separate chance constraints provides us with a flexible modeling framework, which can prioritize the set of goals at hand. In practice, another important criterion is the computational tractability of the resulting mathematical programming formulations. Optimization with a joint chance constraint is generally significantly more challenging than optimization with individual chance constraints (see, e.g., Luedtke et al., 2010; Kucukyavuz, 2012; Lejeune, 2012). Enforcing a joint chance constraint with a high probability level on a large set of goal constraints typically leads to individual probability levels close to one,

and consequently, may result in very conservative solutions. As a partial remedy, the decision maker may opt for decreasing the probability level of the joint chance constraint, but this exacerbates the computational difficulties. Therefore, one may prefer to employ individual chance constraints even if a joint chance constraint is more appropriate from a modeling perspective. In particular, it is natural to use individual chance constraints to develop a computationally tractable approximation of a joint chance-constrained optimization model (see, e.g., Prékopa, 2003; Haneveld and van der Vlerk, 2015).

In the first part of this thesis, we focus on single-stage stochastic programming models with individual chance constraints under variable reliability (probability) levels, and apply the proposed single-stage modeling approach to a new stochastic post-disaster relief network design problem. On the other hand, in the second part of this thesis, we focus on two-stage stochastic programming models with a joint chance constraint under a fixed reliability level, and develop a risk-averse two-stage stochastic model for a new stochastic pre-disaster relief network design problem.

1.1 Chance-Constrained Single-Stage Stochastic Programming Models

The first part of this thesis is dedicated to individual chance-constrained linear programs (LP), where the uncertainty is restricted to the right hand sides of the probabilistic constraints, and the random parameters have discrete distributions. The convexity of the feasible region is still preserved in this structure given the fixed probability levels. A general form of the classical linear program with multiple individual chance constraints is then given by:

$$
\begin{aligned}
(\mathbf{MICC}) \qquad & \text{minimize} && \mathbf{c}^T\mathbf{x} && && (1.1)\\
& \text{subject to} && \mathbb{P}(T_k\mathbf{x} \geq \xi_k) \geq 1-\epsilon_k, && \forall\, k = 1,\ldots,m && (1.2)\\
& && \mathbf{x} \in \mathcal{X}. && && (1.3)
\end{aligned}
$$

Here, the feasible set associated with the deterministic constraints is represented by the polyhedron $\mathcal{X} \subseteq \mathbb{R}^n_+$. We assume that T is a deterministic $m \times n$ matrix with its kth row denoted by T_k, and $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_m)^T$ is an m-dimensional random vector with a finite support. The realizations of the random vector $\boldsymbol{\xi}$ are denoted by $\boldsymbol{\xi}^i$ with corresponding probabilities $p^i > 0$ for $i \in \{1, \ldots, N\}$. The individual chance constraints (1.2) ensure that the stochastic (goal) constraint $T_k\mathbf{x} \geq \xi_k$ holds with a probability at least equal to $1-\epsilon_k$ for all $k \in \{1, \ldots, m\}$, where ϵ_k is the risk tolerance corresponding to the reliability level $1-\epsilon_k$. Note that integrality requirements can also be incorporated into the definition of $\mathcal{X}$, and our entire modeling and solution framework directly carries over to individual

chance-constrained *integer* linear programs as well.

It is well-known that $(\mathbf{MICC})$ can be reformulated as a linear program when the risk tolerances $\epsilon_k,\ k = 1, \ldots, m$, are input parameters specified by the decision maker. Alternatively, we can consider the risk tolerances / reliability levels as decision variables, and the resulting mathematical models can be put into good use in several ways at the expense of additional complexity. For instance, $(\mathbf{MICC})$ with a set of additional constraints on the variable/adjustable reliability levels is proposed by Prékopa (2003) – see formulation (5.10) – as an approximation of a joint chance-constrained optimization model. Another motivation for varying the values of the reliability levels is to perform a Pareto analysis in order to extract insights about the trade-off between the actual cost/return factors and the cost associated with the probabilities of the undesirable events of interest (Rengarajan and Morton, 2009; Rengarajan et al., 2013). In a similar vein, Shen (2014) proposes a new class of optimization models with adjustable reliability levels, where the author incorporates a linear cost function of the individual risk tolerances into the objective function (1.1). Lejeune and Shen (2016) follow this line of research in the context of joint chance-constrained optimization. They consider two types of a joint constraint – with a deterministic technology matrix T on the left-hand side (randomness exists only on the right-hand side) and with a random technology matrix T – and develop effective mathematical programming formulations based on a Boolean modeling framework. We also refer to Lejeune and Shen (2016) for a detailed review on studies which consider a trade-off between the conflicting cost/return criteria and reliability objectives.

Our study is directly related to and motivated by Shen (2014). The author presents a mixed-integer linear programming (MIP) reformulation for $(\mathbf{MICC})$ with adjustable risk tolerances. However, the reformulation of the chance constraints relies on the classical big-M paradigm and solving large instances presents a formidable challenge due to the well-known weakness of the LP relaxations of formulations featuring big-Ms. Our primary objective in this study is therefore to offer a computationally effective MIP reformulation of $(\mathbf{MICC})$ when the reliability levels are treated as decision variables. To this end, we exploit recent methodological advances for modeling and solving chance-constrained linear programs with fixed reliability levels. In particular, we use a modeling approach similar to that presented in Luedtke et al. (2010). The fundamental idea is to rewrite the chance constraints (1.2) in the form of $T_k\mathbf{x} \geq z_k$, where z_k corresponds to the $(1-\epsilon_k)$th quantile of the random component ξ_k for $k \in \{1, \ldots, m\}$. Variable reliability levels render the quantile values denoted by $z_k,\ k \in \{1, \ldots, m\}$, variable as well, and reformulating the chance constraints requires being able to express the quantile values as functions of the reliability level variables. To this end, we develop two alternate approaches to express the decision variables $z_k,\ k \in \{1, \ldots, m\}$. The first representation is based on the mixing inequalities proposed by Luedtke et al. (2010). The authors study the mixing set with a knapsack constraint arising in the deterministic equivalent formula-

tion of a joint chance-constrained optimization model with a finitely distributed random right-hand side and a fixed reliability level. It turns out that the results of this work can be applied to individual chance-constrained optimization models with adjustable risk tolerances as well. An alternate second representation arises from using a different set of binary variables to identify the scenarios under which the goal constraints are violated. The resulting MIP formulations outperform the current state-of-the-art based on the big-M type of constraints by a significant margin.

A large number of application-based studies in a wide range of areas depend on the big-M paradigm to linearize the chance constraints. The computational challenges faced in solving the resulting deterministic equivalent formulations with a large number of scenarios, however, severely restrict our ability in capturing the inherent uncertainty more accurately by increasing the number of scenarios. Therefore, one noteworthy contribution of our work is to highlight the existence and efficacy of alternate formulations for individual chance-constrained (integer) linear programs with and without variable risk tolerances and make recent methodological progress in modeling and solving chance-constrained optimization models more accessible to practitioners. In addition, we elaborate on how to construct a cost function of the variable reliability levels and extend/modify Shen (2014)'s model by quantifying the cost of reliability with a different perspective. Ultimately, we intend to reach a wide audience by applying the proposed powerful and easy-to-implement modeling approaches to a stochastic network design problem in a post-disaster humanitarian logistics context. Our computational results, which clearly illustrate that large instances of chance-constrained models with variable reliability levels are well-solved with the proposed MIP formulations, attest to the effectiveness of the concepts laid out in this chapter.

Optimization capturing the trade-off between the actual cost factors and the cost of the risk tolerances associated with the chance constraints is a fairly recent research area, and such a hybrid modeling approach has promise to be applied in different fields. In this study, we make our point by applying the proposed modeling approach to humanitarian relief logistics, where it may be essential to consider multiple and possibly conflicting performance criteria – such as accessibility and equity, see, e.g., Noyan et al. (2016). In particular, we focus on balancing the trade-off between accessibility and the level of demand satisfaction in the context of post-disaster relief network design. We introduce a new stochastic last mile distribution network design problem, which determines the locations of the Points of Distribution (PODs), the assignments of the demand nodes to PODs, and the delivery amounts to the demand nodes while considering the equity and accessibility issues and incorporating the inherent uncertainties. The studies that consider decisions related to the locations of the last mile facilities are scarce, and as emphasized in Noyan et al. (2016), the majority of these studies either assume a deterministic setting and/or do not incorporate the concepts of accessibility and equity. Our study contributes

to the humanitarian relief literature by introducing a new hybrid supply allocation policy and developing a new risk-averse optimization model. We delay the relevant literature review on relief network design until Section 2.3.

1.2 Chance-Constrained Two-Stage Stochastic Programming Models

In the second part of this thesis, we focus on the long-term pre-disaster planning and study the problem of designing a pre-disaster relief network to respond to sudden-onset natural disasters. This problem determines the response facility location and stock pre-positioning decisions in order to improve the effectiveness of the post-disaster relief operations. The importance of long-term pre-disaster planning and taking into consideration the potentially high level of inherent uncertainty have been emphasized in the recent literature (see e.g., Balcik and Beamon, 2008; Salmerón and Apte, 2010; Hong et al., 2015). Since the design decisions must be made well before a disaster strikes (for instance, without knowing the epicenter and the intensity of an earthquake), it is crucial to develop sound decision-making models in the presence of uncertain parameters. In particular, we consider the uncertainty in demand and transportation infrastructure and develop a two-stage stochastic programming model. Several two-stage stochastic programming problems are proposed for pre-disaster management (see, e.g., Balcik and Beamon, 2008; Rawls and Turnquist, 2010; Salmerón and Apte, 2010; Mete and Zabinsky, 2010; Döyen et al., 2012). However, most of the existing studies with a few exception (e.g., Rawls and Turnquist, 2011; Noyan, 2012; Hong et al., 2015) propose risk-neutral two-stage stochastic programming models. However, relying on expected values and disregarding the variability in random parameters may not be good enough for rarely occurring disaster events; risk-neutral models may provide solutions that perform poorly under certain realizations of the random data. These concerns call for risk-averse optimization models that incorporate the random variability inherent in chaotic disaster relief systems.

We introduce a new risk-averse two-stage stochastic optimization model for pre-disaster relief network design under random demand and transportation network conditions. It features a mean-risk objective based on the popular risk measure conditional-value-at-risk (CVaR), and enforces a joint probabilistic constraint on the feasibility of the second-stage problem concerned with distributing the relief supplies to the affected areas in case of a disaster. In the context of humanitarian logistics, Rawls and Turnquist (2010), Noyan (2012), and Hong et al. (2015) are the studies most closely related to ours. Rawls and Turnquist (2010) introduce a pre-disaster relief network design problem which determines the locations and size types (capacities) of the response facilities and their inventory levels of multiple types of relief supplies in the presence of uncertainty in demand, trans-

portation (link) capacities, and the damage level of pre-stocked supplies. They develop a risk-neutral two-stage stochastic programming model, which is later extended in Noyan (2012) by incorporating CVaR as the risk measure on the total cost in addition to its expectation. In the common underlying model, the stochastic demand satisfaction constraints are relaxed and the corresponding violations are penalized in the objective function. As an alternative to this quantitative approach, Hong et al. (2015) propose to use a qualitative approach which controls the violation of the relaxed demand satisfaction constraints via a joint chance constraint. In fact, they only use a qualitative mean to control the feasibility of the second-stage problem and do not capture an objective function for the second-stage problem. On the other hand, we follow the study of Bülbül et al. (2016) and propose to benefit from both types of modeling approaches by developing a hybrid risk-averse two-stage stochastic model, which takes into account both the quantitative and qualitative aspects of risk.

The proposed risk-averse two-stage modeling approach subsumes the well-known risk-neutral counterpart (Rawls and Turnquist, 2010) and the existing risk-averse counterparts (Noyan, 2012; Hong et al., 2015) as special cases. Moreover, it provides a flexible way of modeling decision makers' preferences based on multiple stochastic criteria. One way of impacting the trade-off between the multiple conflicting criteria (such as acquisition costs and level of demand satisfaction) is through changing the probability level of the joint chance constraint. To the best of our knowledge, developing such a risk-averse two-stage stochastic model is novel in the disaster management literature. In our study, we also use a different approach in modeling the relief distribution problem at the second-stage. The relevant existing studies follow the modeling approach of Rawls and Turnquist (2010) and consider a particular type of network design model, where the second-stage problem is formulated as a classical transportation network flow model. Alternatively, we introduce an assignment-based second-stage problem to guarantee a more structured distribution service considering the common practices in relief operations. In particular, the main focus is on identifying the assignments of the demand points to the open facilities and determining the delivery amounts to the demand points from the corresponding facilities.

We consider a finite probability space, where the possible outcomes (joint realizations of the random parameters) are considered as scenarios. It is well-known that the scenario-based stochastic programming models are generally computationally challenging and introducing a joint chance-constraint further complicates the solution of these non-convex models. Several decomposition methods have been developed for solving risk-neutral two-stage stochastic programs. Recently, their variants have also been developed for the two-stage models involving a risk measure (see, e.g., Ahmed, 2006; Schultz and Tiedemann, 2006; Noyan, 2012) and a joint chance constraint (Luedtke, 2014; Liu et al., 2016). Following these recent developments, Bülbül et al. (2016) employ a Ben-

ders decomposition-based branch-and-cut algorithm for the hybrid model of interest in the context of single-machine scheduling. In particular, the algorithm adapts the feasibility and optimality cuts presented in Luedtke (2014) and Liu et al. (2016), respectively, to guarantee the satisfaction of the joint chance-constraint and the exact calculation of the optimal second-stage objective function (recourse function) values. In addition, it relies on the basic linear representation of CVaR to approximate the CVaR measure in the objective function (as in Noyan, 2012). We adapt their methods to devise an exact Benders decomposition-based branch-and-cut algorithm for solving the proposed risk-averse relief network design model.

1.3 Outline

The rest of the thesis is organized as follows. In Chapter 2, we focus on single-stage stochastic programming models with individual chance constraints under variable reliability levels. We present effective alternate MIP formulations of the models of interest, and apply the proposed modeling approach to a new stochastic last mile distribution network design problem. Chapter 3 is dedicated to a chance-constrained two-stage stochastic programming approach in the context of pre-disaster relief network design. We conclude in Chapter 4 with further research directions.

Chapter 2

Chance-Constrained Single-Stage Stochastic Programming Models

In this chapter, we focus on optimization models involving individual chance constraints, where the uncertainty is restricted to the right hand sides of the probabilistic constraints, and the random parameters have discrete distributions. A recently introduced class of such models treats the reliability levels associated with the chance constraints as decision variables and trades off the actual cost against the cost of the selected reliability levels in the objective function. Leveraging recent methodological advances for modeling and solving chance-constrained linear programs with fixed reliability levels, we develop strong mixed-integer programming formulations for this new variant with variable reliability levels. In addition, we elaborate on how to construct a cost function of the variable reliability levels and introduce an alternate cost function type which requires capturing the VaR associated with a variable reliability level. We accomplish this task via a new integer linear programming representation of VaR. Our computational study illustrates the effectiveness of our mathematical programming formulations. We also apply the proposed modeling approach to a stochastic network design problem in a post-disaster humanitarian logistics context. Our numerical results – for a case study based on the real-world data from the 2011 Van earthquake in Turkey – clearly attest to the effectiveness of the concepts laid out in this chapter. For a more detailed introduction we refer the reader to Section 1.1.

The rest of the chapter is organized as follows. In Section 2.1, we present two effective alternate MIP formulations of $(\mathbf{MICC})$ with adjustable/variable reliability levels. In Section 2.2, we extend these formulations to a related class of models that focus on balancing the trade-off between the cost/return and the reliability levels. Section 2.3 is dedicated to the new stochastic last mile distribution network design problem discussed above. This is followed in Section 2.4 by the computational study.

2.1 Optimization Models with Individual Chance Constraints

In this section, we first present the classical mathematical programming reformulation of $(\mathbf{MICC})$ with fixed reliability levels, and then delve into the extensions of $(\mathbf{MICC})$ with variable risk tolerances. Before we proceed, some of the conventions used throughout the chapter are due here. F_X designates the cumulative distribution function of a random variable X. The set of the first n positive integers is denoted by $[n] = \{1, \ldots, n\}$, while $[a]_+ = \max(a, 0)$ indicates the positive part of a number $a \in \mathbb{R}$.

Quantiles play a major part in the reformulations of chance-constrained optimization models, where the first quantile function $F_X^{(-1)} \ : \ (0, 1] \to \mathbb{R}$ of a random variable X is the *generalized inverse* of the cumulative distribution F_X given by $F_X^{(-1)}(\alpha) = \inf\{\eta \in \mathbb{R} \ : \ F_X(\eta) \geq \alpha\}$. This quantity is also known as the value-at-risk (VaR) of the random variable X at confidence level $\alpha \in (0, 1]$ and denoted by $\mathrm{VaR}_\alpha(X)$. Observing that the equivalence relation

$$\mathbb{P}(T_k\mathbf{x} \geq \xi_k) \geq 1 - \epsilon_k \quad \Leftrightarrow \quad T_k\mathbf{x} \geq F_{\xi_k}^{(-1)}(1 - \epsilon_k), \tag{2.1}$$

holds for any $k \in [m]$ yields the following LP as the deterministic equivalent formulation of $(\mathbf{MICC})$ with fixed risk tolerances $\epsilon_k, \ k \in [m]$:

$$\begin{aligned}
(\mathbf{MICC-DEF}) \qquad \text{minimize} \quad & \mathbf{c}^T\mathbf{x} && && (2.2)\\
\text{subject to} \quad & T_k\mathbf{x} \geq F_{\xi_k}^{(-1)}(1 - \epsilon_k), && \forall\, k \in [m] && (2.3)\\
& \mathbf{x} \in \mathcal{X}. && && (2.4)
\end{aligned}$$

The inequalities in (2.3) are linear because the quantile values $F_{\xi_k}^{(-1)}(1 - \epsilon_k), \ k \in [m]$, are input parameters calculated a priori for the given set of scenarios representing the distribution of the random vector $\boldsymbol{\xi}$. Since we enforce individual chance constraints, without loss of generality we can assume that the realizations of ξ_k are relabeled so that $\xi_k^1 \geq \cdots \geq \xi_k^N$ holds with the corresponding probabilities $p_k^i, \ i = 1, \ldots, N$, for all $k \in [m]$. Then, it is easy to see that we have $F_{\xi_k}^{(-1)}(\alpha) = \xi_k^{i'_k}$ for all $k \in [m]$, where

$$i'_k := \max\{i \in [N] \ : \ \sum_{l=1}^{i} p_k^l \leq 1 - \alpha\} + 1 = \max\{i \in [N] \ : \ \sum_{l=i}^{N} p_k^l \geq \alpha\}. \tag{2.5}$$

As discussed in Section 1.1, one may prefer to allow the reliability levels to be decision variables in some decision making problems. Consequently, the quantile values are also incorporated as decision variables into this version of $(\mathbf{MICC})$ – referred to as $(\mathbf{MICC-VR-DEF})$ in this chapter. In this case, we cannot preserve an LP structure

and have to resort to the binary variables $\beta_{ki},\ k \in [m],\ i \in [N]$, so that β_{ki} is set to one if $T_k\mathbf{x} < \xi_k^i$ – as ensured by the constraints (2.6b) in the formulation below:

$$(\mathbf{MICC-VR-DEF})$$

$$\begin{aligned}
\text{minimize} \quad & \mathbf{c}^T\mathbf{x} && && (2.6a)\\
\text{subject to} \quad & T_k\mathbf{x} \geq \xi_k^i - M\beta_{ki}, && \forall\, k \in [m],\ i \in [N] && (2.6b)\\
& \sum_{i\in[N]} p_k^i \beta_{ki} \leq \epsilon_k, && \forall\, k \in [m] && (2.6c)\\
& A\boldsymbol{\epsilon} \leq \mathbf{b}, && && (2.6d)\\
& 0 \leq \epsilon_k \leq \bar{\epsilon}_k, && \forall\, k \in [m] && (2.6e)\\
& \mathbf{x} \in \mathcal{X}, && && (2.6f)\\
& \boldsymbol{\beta} \in \{0,1\}^{m\times N}. && && (2.6g)
\end{aligned}$$

The constraints (2.6b)-(2.6c), where M is a large number, collectively mandate that $\mathbb{P}(T_k\mathbf{x} \geq \xi_k) \geq 1 - \epsilon_k$. Given an optimal solution $\left(\hat{\mathbf{x}},\ \hat{\boldsymbol{\beta}},\ \hat{\boldsymbol{\epsilon}}\right)$, we have $T_k\hat{\mathbf{x}} \geq \max\{\xi_k^i\ :\ i \in [N],\ \hat{\beta}_{ki} = 0\}$ for $k \in [m]$. It is easy to see that the right hand side of this expression is the quantile value in (2.3) for the risk tolerance $\epsilon_k = \sum_{i\in[N]:\hat{\beta}_{ki}=1} p_k^i$. Two types of constraints are imposed on the variable risk tolerances $\epsilon_k,\ k \in [m]$. The reliability level of an individual chance constraint $k \in [m]$ can be no less than $1 - \bar{\epsilon}_k$ as ensured by the simple upper bound constraints (2.6e). In addition, we include a general set of constraints (2.6d) with $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_m)^T$ on the variable risk tolerances. If $(\mathbf{MICC-VR-DEF})$ is employed as an approximation of a corresponding optimization model with an embedded joint chance constraint of the form $\mathbb{P}(T_k\mathbf{x} \geq \xi_k,\ k \in [m]) \geq \alpha$ as discussed in the introduction, then (2.6d) consists of a single constraint $\sum_{k\in[m]} \epsilon_k \leq 1 - \alpha$ and also implies that $\bar{\epsilon}_k = 1 - \alpha$ for all $k \in [m]$ in (2.6e). Here, allowing variable reliability levels is of essence to improve the effectiveness of the approximation. In a different application – say in a production planning problem, constraints (2.6d) may reflect the relative importance of an individual chance constraint with respect to the others. To illustrate, if customer 1 requires a higher service level than customer 2 in a demand satisfaction context, then the constraint $a_{12}\epsilon_1 \leq \epsilon_2$ with $a_{12} > 1$ would be included in the formulation.

A MIP formulation of the form $(\mathbf{MICC-VR-DEF})$ quickly becomes computationally intractable as the number of scenarios increases due to the presence of the big-M type of constraints – such as (2.6b). As underlined in Section 1.1, one of the promises of this study is to empower practitioners to solve large-scale individual chance-constrained optimization problems by demonstrating good modeling practices put forward by recent methodological progress. The key result, which enables stronger and computationally effective formulations, is that for a given $k \in [m]$, the set of constraints

$T_k\mathbf{x} \geq \xi_k^i - M\beta_{ki},\ i \in [N]$, can be substituted by a single constraint

$$T_k\mathbf{x} \geq z_k, \tag{2.7}$$

where z_k is a quantile expressed purely through the realizations $\xi_k^i,\ i = 1, \ldots, N$, and a set of binary variables. This result rests on the equivalence relation (2.1) and the fact that the realizations of $\boldsymbol{\xi}$ can be sorted independently for each component $k \in [m]$ because we deal with individual chance constraints. These two observations together imply the existence of a threshold index i'_k for any fixed set of decisions $\bar{\mathbf{x}}$ such that $T_k\bar{\mathbf{x}} < \xi_k^i$ for $i = 1, \ldots i'_k - 1$, and $T_k\bar{\mathbf{x}} \geq \xi_k^i$ for $i = i'_k, \ldots N$. Correspondingly, the values of the associated β-variables are set as $\beta_{ki} = 1$ for $i = 1, \ldots i'_k - 1$, and $\beta_{ki} = 0$ for $i = i'_k, \ldots N$. Exploiting this structure and detecting the correct value of i'_k is the central theme in the development of the valid inequalities and variable reduction techniques summarized in the next lemma. The proofs are omitted either because they are simple to derive or exist in the references provided.

Lemma 1

i. ***Preprocessing – variable reduction*** *(see, e.g., Luedtke et al., 2010; Lejeune and Noyan, 2010). Let* $i_k^* := \max\{i \in [N] \ :\ \sum_{l=1}^{i} p_k^l \leq \bar{\epsilon}_k\} + 1$ *for* $k \in [m]$*. Then, the necessary condition for any* $\mathbf{x} \in \mathcal{X}$ *to satisfy* (1.2) *with* $\epsilon_k \leq \bar{\epsilon}_k,\ k \in [m]$*, is that the following quantile-based inequalities hold:*

$$T_k\mathbf{x} \geq \xi_k^{i_k^*} \quad \forall\, k \in [m].$$

Consequently, $\beta_{ki} = 0$ *for all* $k \in [m],\ i \in \{i_k^*, \ldots, N\}$*, in any feasible solution of* $(\mathbf{MICC - VR - DEF})$*, and these variables can be omitted from the formulation. Note that* $\bar{\epsilon}_k$ *is an input parameter while* ϵ_k *is a decision variable, and by the definition of* i_k^* *we have* $F_k^{(-1)}(1 - \bar{\epsilon}_k) = \xi_k^{i_k^*} \geq \xi_k^i$ *for all* $i \in \{i_k^*, \ldots, N\}$*. Then, by the monotonicity of the first quantile function,* $F_k^{(-1)}(1 - \epsilon_k) \geq F_k^{(-1)}(1 - \bar{\epsilon}_k)$ *holds, and the assertion follows from* (2.1)*.*

ii. ***Valid inequalities*** *(see, e.g., Ruszczyński, 2002; Luedtke et al., 2010). The following inequalities are valid for* $(\mathbf{MICC - VR - DEF})$*:*

$$\beta_{ki} \geq \beta_{k,i+1} \quad \forall\, k \in [m],\ i \in [N-1].$$

iii. For a discrete random variable ξ_k *with realizations* $\xi_k^1 \geq \cdots \geq \xi_k^N$ *and corresponding probabilities* $p_k^i,\ i \in [N]$, $\mathrm{VaR}_\gamma(\xi_k) = \mathrm{VaR}_\alpha(\xi_k)$ *holds for any* $\gamma \in (\alpha_-, \alpha_+]$*, where* $\alpha_- = \mathbb{P}(\xi_k < \mathrm{VaR}_\alpha(\xi_k))$ *and* $\alpha_+ = \mathbb{P}(\xi_k \leq \mathrm{VaR}_\alpha(\xi_k))$*. In other words, the quantile*

function $F_k^{(-1)}$ *in* (2.1) *is constant on the following intervals:*

$$F_k^{(-1)}(\alpha) = \xi_k^i \qquad \forall\, \alpha \in \left(\sum_{l=i+1}^{N} p_k^l, \sum_{l=i}^{N} p_k^l \right], \; k \in [m]. \tag{2.8}$$

iv. ***Valid equalities****. Based on item* (iii)*, there exists an optimal solution to* $(\mathbf{MICC - VR - DEF})$ *for finite probability spaces such that the variable* ϵ_k *is set to one of the values* $\left\{\tilde{p}_k^{i_k^*-1}, \tilde{p}_k^{i_k^*-2}, \cdots, \tilde{p}_k^1, \tilde{p}_k^0\right\}$*, where* $\tilde{p}_k^i = \mathbb{P}(\xi_k \geq \xi_k^i) = \sum_{l\in[i]} p_k^l$ *for* $i \in [i_k^* - 1]$ *and* $\tilde{p}_k^0 = 0$*. Then, the following constraints are valid for* $(\mathbf{MICC - VR - DEF})$*:*

$$\epsilon_k = \sum_{i\in[i_k^*-1]} \tilde{p}_k^i(\beta_{ki} - \beta_{k,i+1}) \text{ with } \beta_{k,i_k^*} = 0 \quad \forall\, k \in [m].$$

A similar set of constraints featuring a different type of binary variables is used in Shen (2014).

v. ***Strengthened star (mixing) inequalities*** *(Luedtke et al., 2010). The big-*M *constraints* (2.6b) *in* $(\mathbf{MICC - VR - DEF})$ *can be substituted by*

$$T_k\mathbf{x} \geq \xi_k^1 - \sum_{i\in[i_k^*-1]} (\xi_k^i - \xi_k^{i+1})\beta_{ki}, \quad \forall\, k \in [m]. \tag{2.9}$$

Starting from the structure of the $\beta-$variables discussed preceding Lemma 1, it is straightforward to attach an intuitive meaning to the inequalities (2.9). In order to explain the theoretical basis of these valid inequalities, observe that the random vector $\boldsymbol{\xi} \in \mathbb{R}^m$ can be assumed to be non-negative without loss of generality. This trivially implies that $T_k\mathbf{x} \geq 0$ for every feasible solution given any set of reasonable reliability values (i.e, $\epsilon_k < 1,\ k \in [m]$), and we can replace (2.6b) by $T_k\mathbf{x} \geq \xi_k^i(1 - \beta_{ki}),\ \forall\, k \in [m],\ i \in [N]$. This new form of the big-M constraints exposes the connection with the mixing set – a well-known concept in integer programming. This connection is exploited by Luedtke et al. (2010), who focus on a single chance constraint indexed by $k \in [m]$ with a *fixed risk tolerance level* $\bar{\epsilon}_k$ as a means of developing strong MIP formulations for joint chance-constrained LPs. The authors first leverage the knapsack inequality (2.6c) to obtain a strengthening of the mixing set $P_k = \{(y, \boldsymbol{\beta}_k) \in \mathbb{R}_+ \times \{0,1\}^N :\ y + \xi_k^i\beta_{ki} \geq \xi_k^i,\ i \in [N]\}$ associated with the inequalities $T_k\mathbf{x} \geq \xi_k^i(1-\beta_{ki}),\ \forall\, i \in [N]$. Then, they apply the mixing inequalities of Günlük and Pochet (2001) to establish that (2.9) are facet-defining for P_k. The validity of these inequalities for $(\mathbf{MICC - VR - DEF})$ can be argued from the necessary condition (2.3) and the relation $F_{\xi_k}^{(-1)}(1 - \epsilon_k) \geq F_{\xi_k}^{(-1)}(1 - \bar{\epsilon}_k)$ implied by the upper bounding constraints (2.6e).

Recall that the end goal in our endeavor to arrive at a stronger alternative to

($\mathbf{MICC-VR-DEF}$) is to replace the set of constraints $T_k\mathbf{x} \geq \xi_k^i - M\beta_{ki},\ i \in [N]$, by a single constraint (2.7). The main challenge we face in this task is to express $z_k = F_{\xi_k}^{(-1)}(1-\epsilon_k) = \mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)$ via linear constraints, where ϵ_k is a decision variable. To this end, we can build upon Lemma 1 and use one of the linear representations of VaR provided in the next lemma, which – to the best of our knowledge – have not appeared in the literature before. An alternate representation of VaR has been proposed in Küçükyavuz and Noyan (2016); however, that one is more general and is also valid when the realizations cannot be sorted in advance, e.g., when the outcomes depend on the decisions. Ultimately, formulations incorporating this more general representation of VaR are more complicated. In this study, we can enjoy the simpler representations of Lemma 2 for the special case when the realizations can be ordered a priori.

Lemma 2 *Suppose that V is a random variable with realizations $v^1 \geq \ldots \geq v^N$ and corresponding probabilities $p^i > 0,\ i \in [N]$. Let $\tilde{p}^i = \mathbb{P}(V \geq v^i) = \sum_{l\in[i]} p^l$ and $i^* := \max\{i \in [N]\ :\ \tilde{p}^i \leq \bar{\epsilon}\} + 1$, where $\bar{\epsilon}$ is a positive constant in $(0,1]$. Then, for a given risk tolerance ϵ' such that $0 \leq \epsilon' \leq \bar{\epsilon}$, the equality $\mathrm{VaR}_{(1-\epsilon')}(V) = z'$ holds if and only if*

1. *there exists a vector $(z, \epsilon, \beta_0, \boldsymbol{\beta})$ satisfying*

$$z = v^1 - \sum_{i\in[i^*-1]} (v^i - v^{i+1})\beta_i, \tag{2.10a}$$

$$\sum_{i\in[i^*-1]} p^i\beta_i \leq \epsilon, \tag{2.10b}$$

$$\sum_{i\in[i^*-1]} p^i\beta_i + \sum_{i\in[i^*]} (\beta_{i-1} - \beta_i)p^i \geq \epsilon + \delta, \tag{2.10c}$$

$$\beta_0 = 1, \tag{2.10d}$$

$$\beta_{i^*} = 0, \tag{2.10e}$$

$$\beta_i \geq \beta_{i+1}, \qquad \forall\, i \in [i^*-2] \tag{2.10f}$$

$$\beta_i \in \{0,1\}, \qquad \forall\, i \in [i^*], \tag{2.10g}$$

$$0 \leq \epsilon \leq \bar{\epsilon}, \tag{2.10h}$$

   where $\boldsymbol{\beta} \in \{0,1\}^{i^*}$, $z = z'$, and δ is a sufficiently small positive constant to ensure that the constraint (2.10c) *is equivalent to the strict inequality $\sum_{i\in[i^*-1]} p^i\beta_i + \sum_{i\in[i^*]}(\beta_{i-1} - \beta_i)p^i > \epsilon$.*

2. *there exists a vector $(z, \epsilon, \boldsymbol{\beta})$ satisfying*

$$\sum_{i\in[i^*-1]} (\beta_i - \beta_{i+1})\tilde{p}^i = \epsilon, \tag{2.11a}$$

$$(2.10a), (2.10e) - (2.10h), \tag{2.11b}$$

where $\boldsymbol{\beta} \in \{0,1\}^{i^*}$ *and* $z = z'$.

Proof. We distinguish between two cases. First, assume that $0 \leq \epsilon' < p^1$ which implies $\mathrm{VaR}_{(1-\epsilon')}(V) = v^1$. In this case, it is a simple matter to verify that $(z, \epsilon, \beta_0, \boldsymbol{\beta}) = \Big(v^1, \epsilon', 1, \Big(0, \ldots, 0\Big)\Big)$ and $(z, \epsilon, \boldsymbol{\beta}) = \Big(v^1, 0, \Big(0, \ldots, 0\Big)\Big)$ are feasible with respect to (2.10) and (2.11), respectively. Conversely, the existence of a feasible solution $(z, \epsilon, \beta_0, \boldsymbol{\beta}) = \Big(v^1, \epsilon', 1, \Big(0, \ldots, 0\Big)\Big)$ with $0 \leq \epsilon' < p_1$ for (2.10) correctly implies that $\mathrm{VaR}_{(1-\epsilon')}(V) = v^1$. On the other hand, a feasible solution $(z, \epsilon, \boldsymbol{\beta}) = \Big(v^1, 0, \Big(0, \ldots, 0\Big)\Big)$ of (2.11) maps back to $\mathrm{VaR}_{(1-\epsilon')}(V) = v^1$ for any $0 \leq \epsilon' < p^1$ by Lemma 1-(iii).

If $p^1 \leq \epsilon' \leq \bar{\epsilon}$, then $\mathrm{VaR}_{(1-\epsilon')}(V) = v^{\ell+1}$ for some $\ell \in [i^* - 1]$. For this case, we can construct two feasible solutions $(z, \epsilon, \beta_0, \boldsymbol{\beta}) = \Big(v^{\ell+1}, \epsilon', 1, \Big(1, 1, \ldots, 1, 0, \ldots, 0\Big)\Big)$ and $(z, \epsilon, \boldsymbol{\beta}) = \Big(v^{\ell+1}, \tilde{p}^{\ell}, \Big(1, 1, \ldots, 1, 0, \ldots, 0\Big)\Big)$ for (2.10) and (2.11), respectively. In both cases, $\boldsymbol{\beta}$ is composed of ones in the first ℓ positions, followed by all zeros for the remaining components. To check the feasibility of these solutions, note that $\boldsymbol{\beta}$ fulfills (2.10e)-(2.10g) and inserting it into (2.10a) yields $z = v^{\ell+1}$. In addition, the constraints (2.10b)-(2.10d) prescribe that $\sum_{i \in [\ell]} p^i \leq \epsilon' < \sum_{i \in [\ell+1]} p^i$. This must hold when $\mathrm{VaR}_{(1-\epsilon')}(V) = v^{\ell+1}$, and therefore, the first solution is feasible for (2.10). To establish the feasibility of the second solution for (2.11), observe that $\epsilon = \tilde{p}^{\ell}$ satisfies (2.11a), and (2.10h) is ensured by $\ell \leq i^* - 1$ and the definition of i^*. Starting from the same feasible solutions, we can show the converse of these statements and complete the proof through arguments analogous to those in the first part of the proof. ■

As is also evident from the proof of Lemma 2, (2.10) is developed starting from the first expression in (2.5). On the other hand, the underlying premise of (2.11) is Lemma 1-(iii), which allows us to substitute (2.10b)-(2.10d) with (2.11a) to obtain (2.11) from (2.10). This observation further asserts that (2.11) is a stronger representation of $\mathrm{VaR}_{(1-\epsilon)}(V)$ compared to (2.10) because all feasible solutions of (2.10) of the form $(z, \epsilon, \beta_0, \boldsymbol{\beta}) = \Big(v^{\ell+1}, \epsilon', 1, \Big(1, 1, \ldots, 1, 0, \ldots, 0\Big)\Big)$ with $\tilde{p}^{\ell} \leq \epsilon' < \tilde{p}^{\ell+1}$ correspond to a single feasible solution $(z, \epsilon, \boldsymbol{\beta}) = \Big(v^{\ell+1}, \tilde{p}^{\ell}, \Big(1, 1, \ldots, 1, 0, \ldots, 0\Big)\Big)$ of (2.11). Finally, employing the enhancements of Lemma 1 and the VaR representation (2.11), we arrive at a stronger MIP formulation of individual chance-constrained LPs with variable reliability levels given below. An alternate formulation based on the VaR representation (2.10) can be obtained in a very similar fashion.

$(\mathbf{SMICC - VR - \boldsymbol{\beta}})$

$$\begin{aligned}
\text{minimize} \quad & \mathbf{c}^T\mathbf{x} && \text{(2.12a)}\\
\text{subject to} \quad & T_k\mathbf{x} \geq z_k, \quad \forall\, k \in [m] && \text{(2.12b)}\\
& z_k = \xi_k^1 - \sum_{i \in [i_k^* - 1]} (\xi_k^i - \xi_k^{i+1})\beta_{ki}, \quad \forall\, k \in [m] && \text{(2.12c)}\\
& \beta_{ki} \geq \beta_{k,i+1}, \quad \forall\, k \in [m],\ i \in [i_k^* - 2] && \text{(2.12d)}
\end{aligned}$$

$$\beta_{k,i_k^*} = 0, \quad \forall\, k \in [m] \tag{2.12e}$$

$$\epsilon_k = \sum_{i \in [i_k^* - 1]} \tilde{p}_k^i (\beta_{ki} - \beta_{k,i+1}), \quad \forall\, k \in [m] \tag{2.12f}$$

$$\text{(2.6d)} - \text{(2.6f)}, \tag{2.12g}$$

$$\beta_{ki} \in \{0,1\}, \quad \forall\, k \in [m],\ i \in [i_k^*]. \tag{2.12h}$$

For risk-averse decision makers, the typical values for the risk tolerances are small values such as 0.05 and implies that the reasonable choices for the values of the corresponding upper bounds $\bar{\epsilon}_k$ would also be small. Therefore, the variable reduction-based preprocessing methods are very effective in reducing the number of binary variables and constraints. However, in general, even the reduced form of $(\mathbf{MICC - VR - DEF})$ may not scale well with increasing problem size. The proposed formulations featuring the quantile-based representation in the form of (2.7) have a huge positive impact on the solution times and the maximum instance size that can be handled effectively. These claims will be substantiated in Section 2.4.2.

We conclude this section by providing an alternate equivalent formulation of $(\mathbf{SMICC - VR - \boldsymbol{\beta}})$ based on a different set of binary variables. To this end, we define $y_{ki} \in \{0,1\},\ k \in [m],\ i \in [N]$, so that y_{ki} takes the value of 1 if $T_k\mathbf{x} < \xi_k^{i-1}$ and $T_k\mathbf{x} \geq \xi_k^i$ hold together. Clearly, we have $y_{ki} = \beta_{k,i-1} - \beta_{ki}$ with the understanding that $\beta_{k0} = 1$. In this notation, exactly one of the variables $y_{ki},\ i \in [i_k^*]$, – say $y_{ki'}$ – assumes the value 1 for each $k \in [m]$ as mandated by the constraints (2.13e) below. Accordingly, the variable risk tolerance ϵ_k and the corresponding quantile z_k are set to $\tilde{p}_k^{i'-1} = \sum_{l \in [i'-1]} p_k^l$ and $\xi_k^{i'}$ by the constraints (2.13d) and (2.13c), respectively. This VaR representation (2.13c)-(2.13f) expressed via the $y-$variables provides us with the formulation stated next:

$$(\mathbf{SMICC - VR - y}) \quad \text{minimize} \quad \mathbf{c}^T\mathbf{x} \tag{2.13a}$$

$$\text{subject to} \quad T_k\mathbf{x} \geq z_k, \qquad \forall\, k \in [m] \tag{2.13b}$$

$$z_k = \sum_{i \in [i_k^*]} \xi_k^i y_{ki}, \qquad \forall\, k \in [m] \tag{2.13c}$$

$$\epsilon_k = \sum_{i \in [i_k^*]} \tilde{p}_k^{i-1} y_{ki}, \qquad \forall\, k \in [m] \tag{2.13d}$$

$$\sum_{i \in [i_k^*]} y_{ki} = 1, \qquad \forall\, k \in [m] \tag{2.13e}$$

$$y_{ki} \in \{0,1\}, \qquad \forall\, k \in [m],\ i \in [i_k^*] \tag{2.13f}$$

$$\text{(2.6d)} - \text{(2.6f)}. \tag{2.13g}$$

Shen (2014) bases her formulations on these $y-$variables – as will be explained in more depth in the next section. Our motivation for providing this alternate formulation is to form a unified framework for the models introduced in the rest of the chapter and a

foundation for discussing Shen (2014)'s model in the next section. Finally, we point out that the LP relaxations of $(\mathbf{SMICC-VR-}\boldsymbol{\beta})$ and $(\mathbf{SMICC-VR-y})$ are equivalent; however, these two formulations do not necessarily perform similarly in our computational experiments because the different cuts generated by the solver based on the $\beta-$ and $y-$variables lead to different search trees.

2.2 Optimization Models with Individual Chance Constraints Under Variable Reliability Levels

In this section, we focus on a class of optimization models with variable reliability levels, where the objective function also features a cost term associated with the variable risk tolerances. The aim is to make decisions by taking into account the trade-off between the actual cost/return and risk. We first present such an existing model proposed by Shen (2014) and provide associated strong MIP formulations based on the modeling constructs of Section 2.1, which prove to be effective in solving large problem instances in Section 2.4. In the second part of this section, we take a different stance from Shen (2014) in capturing the trade-off between the variable reliability levels and the actual cost/return and introduce an alternate way of quantifying the cost of reliability in the objective. A corresponding new class of optimization models with effective MIP formulations are presented.

2.2.1 Balancing the Actual Return/Cost and Risk

A general form of the extended version of $(\mathbf{MICC-VR-DEF})$, which incorporates the trade-off between the actual cost/return and risk is given by

$$(\mathbf{MICC-VRT}) \qquad \text{minimize} \quad \mathbf{c}^T\mathbf{x} + \sum_{k\in[m]} h_k(\epsilon_k) \tag{2.14}$$

$$\text{subject to} \quad (1.2), (2.6\text{d}) - (2.6\text{f}). \tag{2.15}$$

In this modeling approach, the characterization of the cost function $h_k : [0, \bar{\epsilon}_k] \to \mathbb{R}$ plays a fundamental role. Shen (2014) considers a monotonically increasing linear function of the form $h_k(\epsilon_k) = a_k\epsilon_k$ with $a_k > 0$ for all $k \in [m]$. The author justifies her model by pointing out several problem settings with conflicting individual chance constraints, where it may make sense to enable the decision maker to adjust the individual reliability levels within their respective allowable intervals by solving $(\mathbf{MICC-VRT})$. In a way, this model is akin to multi-criteria methods, which form a single composite objective function by taking a weighted sum of the individual conflicting objectives, and presents an alternative to exploring the Pareto frontier by iteratively solving one of appro-

priate formulations from Section 2.1 with manually adjusted reliability levels. The issue whether a linear function of the risk tolerances incorporated directly into the objective – as in (2.14) – serves the intended purpose is taken up in the next section. We first present the deterministic equivalent formulation of $(\mathbf{MICC-VRT})$ proposed by Shen (2014), which employs the $y-$ variables introduced at the end of Section 2.1. Compared to Shen (2014)'s original formulation, the version below is slightly enhanced by incorporating the preprocessing technique of Lemma 1-(i).

$(\mathbf{MICC-VRT-DEF})$

$$\text{minimize} \quad \mathbf{c}^T\mathbf{x} + \sum_{k\in[m]} a_k\epsilon_k \tag{2.16a}$$

$$\text{subject to} \quad T_k\mathbf{x} \geq \xi_k^i y_{ki} - M(1-y_{ki}), \quad \forall\, k \in [m],\ i \in [i_k^*] \tag{2.16b}$$

$$\sum_{k\in[m]} \epsilon_k \leq \Gamma \tag{2.16c}$$

$$(2.6f), (2.13d) - (2.13f). \tag{2.16d}$$

Obviously, the underlying premise of this formulation is the well-known quantile-based inequality (2.1) and the observation formalized in Lemma 1-(iii). The simple upper bounds on the variable risk tolerances – see (2.6e) – are excluded and the generic restrictions (2.6d) are replaced by a single constraint (2.16c) in this formulation, where Γ is a given budget of risk shared by all chance constraints. The big-M constraints (2.16b) enforce that $T_k\mathbf{x} \geq \xi_k^i y_{ki}$ whenever $y_{ki} = 1$ and can be substituted by the enhanced form $T_k\mathbf{x} \geq \xi_k^i y_{ki}$ by relying on the non-negativity of the random right hand side vector $\boldsymbol{\xi} \in \mathbb{R}^m$ without loss of generality – see the discussion immediately following Lemma 1. This latter form is utilized in our computational study, but it turns out that even the pre-processed reduced formulation $(\mathbf{MICC-VRT-DEF})$ does not scale well as the problem size increases. We illustrate this claim in Section 2.4 by performing computational tests on two classes of problems: a transportation problem (Luedtke et al., 2010) and the stochastic network design problem described in Section 2.3.

The work in Shen (2014) is structured into two parts. In the first part, the attention is restricted to the special case of $(\mathbf{MICC-VRT})$ with just a *single* individual chance constraint by setting $m = 1$. In this context, the author also presents two methods alternative to solving $(\mathbf{MICC-VRT-DEF})$ directly. Both of these leverage Lemma 1-(iii). The first approach conducts a bisection search over the discrete set $\left\{\tilde{p}_k^{i_k^*-1}, \tilde{p}_k^{i_k^*-2}, \cdots, \tilde{p}_k^1, \tilde{p}_k^0\right\}$ for the correct value of ϵ_k but leads to an *inexact* algorithm. The latter approach executes an exhaustive enumeration over the same discrete set. In both cases, the author repeatedly solves $(\mathbf{MICC})$ with $m = 1$ and then adds the cost associated with the given fixed reliability level to the resulting objective function value. The related computational experiments are performed on a minimum cost network flow prob-

lem with a single demand node, and the results clearly demonstrate that the exhaustive enumeration has no merit over solving $(\mathbf{MICC-VRT-DEF})$ for large instances. In the second part of Shen (2014), the focus shifts to the general problem with multiple individual chance constraints, where solving $(\mathbf{MICC-VRT-DEF})$ is the sole generic tool proposed due to the combinatorial explosion of the exhaustive enumeration procedure. For this case, numerical results are reported on a multi-commodity flow network capacity design problem with a pretty limited number of – three or six – individual chance constraints. The number of scenarios is kept constant throughout, and the results indicate that the computational effort expended tends to grow sharply with increasing network size and larger values of Γ. In her concluding remarks, Shen (2014) points out that iterative methods relying on repeatedly solving $(\mathbf{MICC})$ with fixed reliability levels is so far not promising for tackling $(\mathbf{MICC-VRT})$ and emphasizes the need for good MIP formulations to this end. We take up this issue in this study and develop substantially more effective and scalable MIP formulations for $(\mathbf{MICC-VRT})$ by drawing upon the modeling tools of Section 2.1.

Compared to $(\mathbf{SMICC-VR-y})$ in the previous section, the drawback of the formulation $(\mathbf{MICC-VRT-DEF})$ is that it fails to insert sufficiently large quantile values into the right hand sides of the constraints (2.16b). The crucial factor here is calculating the quantile values correctly as a function of the variable risk levels $\epsilon_k,\ k \in [m]$, and representing the inequalities (2.16b) compactly. This issue is also of essential significance in Section 2.2.2, where we devise an alternate expression for capturing the cost of reliability in the objective function. Ultimately, we obtain a strong formulation of $(\mathbf{MICC-VRT})$ by appending the term $\sum_{k\in[m]} a_k\epsilon_k$ to the objective function of $(\mathbf{SMICC-VR-y})$. This formulation is referred to as $(\mathbf{SMICC-VRT-y})$ in the rest of the chapter. In a similar vein, incorporating the same term $\sum_{k\in[m]} a_k\epsilon_k$ into the objective function of $(\mathbf{SMICC-VR-}\boldsymbol{\beta})$ yields an alternate strong formulation $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ based on the $\beta-$variables. The numerical results in Section 2.4 attest to the significant computational edge of $(\mathbf{SMICC-VRT-y})$ and $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ over $(\mathbf{MICC-VRT-DEF})$ – with gains of up to two orders of magnitude in solution times for some instances. These experimental outcomes highlight the recent advances in the field of stochastic programming (see, e.g., Luedtke et al., 2010; Kucukyavuz, 2012) and confirm their potential applicability to practical problems, in which the uncertainty is captured with a large number of scenarios.

2.2.2 A New Approach to Quantifying the Cost of Reliability

Chance constraints are based on a qualitative risk concept and measure the probabilities of violating the stochastic constraints, irrespective of the magnitude of violation. Shen (2014)'s approach of incorporating the term $\sum_{k\in[m]} a_k\epsilon_k$ into the objective function may

be considered as an attempt at constructing a hybrid stochastic model with both qualitative and quantitative aspects. We concur with this perspective and also quote from Prékopa (1995), who notes that "the best model construction is the one which combines the use of a probabilistic constraint and penalties in the objective function," as a further motivating support from the literature. However, we find that a linear function of the risk tolerances directly appended to the objective function falls short of properly capturing the cost of reliability for two reasons. First, the two components of the objective function (2.16a) are not commensurate; i.e., they are not on the same scale. Second, and more importantly, the structure of (2.16a) implies that decreasing the value of ϵ_k from 0.10 to 0.09 and from 0.05 to 0.04 have the same impact of $0.01a_k$ on the reliability component of the objective function value. This is hardly justifiable as the first quantile function is not linear in the risk tolerance. To provide a concrete example, suppose that decreasing the risk tolerance from 0.10 to 0.09 requires supplying an additional 50 units of a particular product, while decreasing the risk tolerance from 0.05 to 0.04 may require an additional supply of 200 items. Clearly, the cost function associated with reliability should not treat these two cases identically. Ultimately, we contend that in order to account for the nonlinear structure of the first quantile function, the cost coefficient a_k in (2.16a) should be specified according to the level of the quantity-based service associated with a particular risk tolerance. Observe that according to the relation (2.1), changing the value of the risk tolerance ϵ_k is equivalent to changing the right-hand side of the corresponding constraint in the deterministic equivalent formulation – see, e.g., (2.13b)-(2.13d). From this viewpoint, it is more natural to define the cost function associated with reliability explicitly in terms of the associated VaR values. More specifically, we suggest to quantify the cost of reliability with a function of the form $h_k\left(\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right)$, which leads to the problem statement:

$$(\mathbf{MICC-VRTQ}) \qquad \text{minimize} \quad \mathbf{c}^T\mathbf{x} + \sum_{k\in[m]} h_k\left(\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right) \tag{2.17}$$
$$(1.2), (2.6\text{d}) - (2.6\text{f}).$$

In this context, larger VaR values – equivalently, smaller ϵ_k values – are preferred in terms of the reliability levels of the chance constraints. Consequently, we consider monotonically *decreasing* cost functions $h_k()$ in order to incorporate the cost of compromising from the reliability levels. The particular focus is on functions that can be represented by linear objective terms and constraints.

Due to the variability of the reliability levels, the main task in reformulating $(\mathbf{MICC-VRTQ})$ as a MIP is to express $F^{(-1)}_{\xi_k}(1-\epsilon_k) = \mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)$ via linear constraints and can be accomplished by drawing upon the modeling tools of Section 2.1. A MIP formulation of $(\mathbf{MICC-VRTQ})$ is obtained from $(\mathbf{SMICC-VRT-y})$ in

a straightforward way:

$$(\mathbf{SMICC-VRTQ-y}) \qquad \text{minimize} \quad \mathbf{c}^T\mathbf{x} + \sum_{k\in[m]} h_k(z_k) \tag{2.18a}$$

$$\text{subject to} \quad \text{(2.13b)} - \text{(2.13g)}. \tag{2.18b}$$

We can also develop alternate MIP formulations employing the $\beta-$ variables in the spirit of Sections 2.1 and 2.2.1. As underlined previously at the end of Section 2.1, equivalent formulations based on the $y-$ and $\beta-$variables are closely related, but they may still exhibit different computational behavior. From this perspective, it makes sense to extend $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ by integrating the new cost functions $h_k\left(\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right)$, $k \in [m]$. To this end, we can put either one of the VaR representations in Lemma 2 to use. The presentation (2.11) was previously demonstrated in $(\mathbf{SMICC-VR-}\boldsymbol{\beta})$. To illustrate the use of (2.10), here we only present the MIP formulation of $(\mathbf{MICC-VRTQ})$ obtained from $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ by incorporating (2.10):

$(\mathbf{SMICC-VRTQ-}\boldsymbol{\beta})$

$$\text{minimize} \quad \mathbf{c}^T\mathbf{x} + \sum_{k\in[m]} h_k(z_k) \tag{2.19a}$$

$$\text{subject to} \quad \text{(2.12b)} - \text{(2.12e)}, \text{(2.12g)} - \text{(2.12h)}, \tag{2.19b}$$

$$\sum_{i\in[i_k^*-1]} p_k^i \beta_{ki} \leq \epsilon, \qquad \forall\, k \in [m] \tag{2.19c}$$

$$\sum_{i\in[i_k^*-1]} p_k^i \beta_{ki} + \sum_{i\in[i_k^*]} (\beta_{k,i-1} - \beta_{ki}) p_k^i \geq \epsilon_k + \delta, \qquad \forall\, k \in [m] \tag{2.19d}$$

$$\beta_{k0} = 1, \qquad \forall\, k \in [m]. \tag{2.19e}$$

An important aspect of the VaR representations in Lemma 2 is that they capture the value of $\mathrm{VaR}_{(1-\epsilon)}$ precisely without relying on the structure of the objective function. This is essential given that the second component of the objective function of $(\mathbf{MICC-VRTQ})$ related to the cost of reliability favors larger values of VaR. Otherwise, if smaller quantile values would be preferred by the objective, then (2.10c) could safely be omitted from the representation (2.10).

We next elaborate on our particular choice of the cost function $h_k\left(\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right)$. One option is to set $h_k\left(\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right) = -a_k\,\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)$ with $a_k > 0$. Alternatively, we define it based on the random outcome $\left[\xi_k - \mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right]_+$, which allows us to incorporate a quantitative measure on the potential violation of the corresponding stochastic goal constraint. As noted at the start of this section, such hybrid approaches are promoted in the literature as good modeling practice (Prékopa, 1995). In this spirit, we

intend to determine the optimal risk tolerance ϵ_k by also taking into account the realizations of ξ_k in excess of the lower threshold value $\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)$. In particular, we focus on the expected violation and set $h_k\left(\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right) = a_k E([\xi_k - \mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)]_+)$. If scenario-dependent cost coefficients are required or preferred, then the cost function takes the form of

$$h_k\left(\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)\right) = \sum_{i\in[i_k^*-1]} a_{ki} p_k^i [\xi_k^i - \mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)]_+. \tag{2.20}$$

For this choice of quantifying the cost of reliability, we define w_{ki} to represent $[\xi_k^i - \mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)]_+$. Then, it is sufficient to substitute the term $\sum_{k\in[m]}\sum_{i\in[i_k^*-1]} a_{ki} p_k^i w_{ki}$ for $\sum_{k\in[m]} h_k(z_k)$ in the objective functions of $(\mathbf{SMICC-VRTQ-y})$ and $(\mathbf{SMICC-VRTQ-}\boldsymbol{\beta})$ and append the following two constraints to the formulation in either case:

$$w_{ki} \geq \xi_k^i - z_k, \qquad \forall\, k \in [m],\ i \in [i_k^* - 1] \tag{2.21}$$

$$w_{ki} \geq 0, \qquad \forall\, k \in [m],\ i \in [i_k^* - 1]. \tag{2.22}$$

The proposed approach is a flexible modeling tool to balance the trade-off between the actual cost/return and the cost of reliability and accounts for the violation of the stochastic goal constraints both qualitatively and quantitatively.

For ease of reference, we provide a summary of the compact MIP formulations of the presented chance-constrained optimization models in Table 2.1.

2.3 A Stochastic Optimization Model for Designing Post-Disaster Relief Networks

We introduce a new stochastic last mile distribution network design problem and develop an associated stochastic optimization model that incorporates the concepts of accessibility and equity while capturing the uncertainty in post-disaster demands and transportation network conditions. The proposed model showcases the generic chance-constrained stochastic programming formulations with variable reliability levels of the previous sections.

2.3.1 Literature Review

There is a growing body of literature devoted to the development of stochastic programming models for humanitarian relief logistics (see, e.g., Liberatore et al., 2013b). The majority of these studies involving facility location decisions are dedicated to the pre-disaster management context (see, e.g., Balcik and Beamon, 2008; Rawls and Turnquist,

Model	MIP Formulation
$(\mathbf{SMICC-VR-}\boldsymbol{\beta})$ Based on β-variables	$\min\{\mathbf{c}^T\mathbf{x}$: (2.12b)-(2.12h)$\}$, or simply, (2.12)
$(\mathbf{SMICC-VR-y})$ Based on y-variables	$\min\{\mathbf{c}^T\mathbf{x}$: (2.13b)-(2.13g)$\}$, or simply, (2.13)
$(\mathbf{MICC-VRT-DEF})$ Shen (2014)'s model	$\min\{\mathbf{c}^T\mathbf{x}+\sum_{k\in[m]} a_k\epsilon_k$: (2.16b)* $-$ (2.16d)$\}$, or simply, (2.16)
$(\mathbf{SMICC-VRT-y})$ $(\mathbf{SMICC-VR-y})$ with trade-off	$\min\{\mathbf{c}^T\mathbf{x}+\sum_{k\in[m]} a_k\epsilon_k$: (2.13b)-(2.13g)$\}$
$(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ $(\mathbf{SMICC-VR-}\boldsymbol{\beta})$ with trade-off	$\min\{\mathbf{c}^T\mathbf{x}+\sum_{k\in[m]} a_k\epsilon_k$: (2.12b)-(2.12h)$\}$
$(\mathbf{SMICC-VRTQ-y})$ $(\mathbf{SMICC-VR-y})$ with VaR-based trade-off	$\min\{\mathbf{c}^T\mathbf{x}+\sum_{k\in[m]} h_k(z_k)$: (2.13b)-(2.13g)$\}$ or simply, (2.18)
$(\mathbf{SMICC-VRTQ-}\boldsymbol{\beta})$ $(\mathbf{SMICC-VR-}\boldsymbol{\beta})$ with VaR-based trade-off	$\min\{\mathbf{c}^T\mathbf{x}+\sum_{k\in[m]} h_k(z_k)$: (2.19b)-(2.19e)$\}$, or simply, (2.19)
$(\mathbf{SMICC-VRTQA-y})$ $(\mathbf{SMICC-VRTQ-y})$ with (2.20)	$\min\{\mathbf{c}^T\mathbf{x}+\sum_{k\in[m],\ i\in[i_k^*-1]} a_{ki}p_k^i w_{ki}$: (2.13b)-(2.13g), (2.21)-(2.22)$\}$
$(\mathbf{SMICC-VRTQA-}\boldsymbol{\beta})$ $(\mathbf{SMICC-VRTQ-}\boldsymbol{\beta})$ with (2.20)	$\min\{\mathbf{c}^T\mathbf{x}+\sum_{k\in[m],\ i\in[i_k^*-1]} a_{ki}p_k^i w_{ki}$: (2.19b)-(2.19e), (2.21)-(2.22)$\}$

Table 2.1: Summary of the MIP formulations of the chance-constrained optimization models.

*: The big-M constraints (2.16b) are substituted by the enhanced form $T_k\mathbf{x} \geq \xi_k^i y_{ki}, \forall\, k \in [m],\ i \in [i_k^*]$.

2010; Salmerón and Apte, 2010; Mete and Zabinsky, 2010; Döyen et al., 2012). Moreover, most of the existing studies – with the exception of a few (e.g., Beraldi and Bruni, 2009; Rawls and Turnquist, 2011; Noyan, 2012; Hong et al., 2015) – propose risk-neutral stochastic programming models. However, making decisions based on the expected values may not be good enough for rarely occurring disaster events, and it may be essential to take into consideration the random variability inherent in chaotic disaster relief systems. To the best of our knowledge, risk-averse stochastic models for post-disaster relief network design are at best scarce, and models with chance constraints are absent. Even the more extensive literature on pre-disaster relief network design includes only a few studies that provide chance-constrained optimization models (Rawls and Turnquist, 2011; Hong et al., 2015).

The majority of the studies related to post-disaster humanitarian operations assume that the locations of the last mile facilities are known and focus on distribution problems addressing vehicle routing and/or supply allocation decisions. Only a few studies are concerned with last mile network design decisions such as the locations and capacities of the Points of Distribution (PODs) of the relief supplies. Moreover, most of the studies taking into account the decisions related to the locations of the last mile facilities either assume a deterministic setting and/or do not incorporate the concepts of accessibility and

equity. Based on these reflections, Noyan et al. (2016) contribute to the literature by introducing a last mile distribution network design problem and presenting a mathematical model that incorporates accessibility and equity while capturing the uncertain aspects of the post-disaster environment. We refer to their study and the references therein for the relevant literature on last mile humanitarian relief logistics and a detailed discussion on the significance of considering the equity and accessibility issues and the inherent uncertainties in the context of last mile relief network design. Kahvecioglu (2014) extends the study of Noyan et al. (2016) by studying a more elaborate integrated last mile network design problem, which relaxes the assumption that there exists a single Local Distribution Center (LDC) with a pre-determined location, and assumes that there already exist some resources located before a disaster occurs and integrates the decisions on the reallocation of pre-stocked relief supplies. At a high level, several aspects differentiate the current work from these previous studies. Noyan et al. (2016) propose a hybrid allocation policy that can balance the trade-off between equity and accessibility and develop a risk-neutral two-stage stochastic programming model with this hybrid supply allocation policy embedded. The current study introduces a new post-disaster relief network design problem and devises a different hybrid supply allocation policy that leverages the chance-constrained framework in focus in order to provide accessible and equitable service to the beneficiaries. Consequently, we move away from the risk-neutral paradigm and construct a novel risk-averse optimization model for post-disaster management following the modeling approach presented in Section 2.2. We elaborate more on the differences in the problem descriptions and model settings in the next section.

2.3.2 Problem Description

We aim to design a two-echelon system, where the relief supplies arriving at an LDC are sent to PODs in the first echelon, and the relief supplies are delivered from PODs to the beneficiaries in the second echelon. We assume that there is a single LDC with a pre-determined location, and consider a single type of relief item that can be a bundle (standard kit) of critical relief supplies, such as prepackaged food, medical kits, and water.

Last mile relief networks must be set up quickly before the relief organizations can collect accurate information about the post-disaster conditions. This essentially implies that the relief organizations need to make the design decisions before the uncertainties related to the post-disaster conditions are resolved. In line with this viewpoint, Noyan et al. (2016) develop a two-stage model, where the first-stage decisions are for locating the PODs and the second-stage decisions are related to the allocation of supplies to the PODs and the assignments of the demand points to the PODs. Alternatively, we consider the situations where the relief organizations need to make all network design decisions immediately in order to start delivering the relief supplies to the affected areas. To this

end, we propose a single-stage stochastic programming model, which jointly determines the decisions related to the following: *i)* the locations of the PODs, *ii)* the assignments of the demand points to the PODs, and *iii)* the distribution of the supplies to the PODs and the demand points.

Following suit with Noyan et al. (2016), we develop an optimization model incorporating the accessibility and equity issues critical to the design of the last mile networks. We follow their approach of characterizing accessibility, defined as the ease of access to the relief supplies. In particular, the accessibility in the first echelon of the last mile network is affected by the physical factors only (e.g., geographical, topographical), while accessibility in the second echelon is affected both by the physical and demographical/socio-economical factors (e.g., age, gender, disability). Noyan et al. (2016) consider an accessibility metric based on the sum of the expected total accessibility of the PODs from the LDC and the expected total accessibility of the PODs from the demand locations. We alternatively design a delivery amount-weighted version of this accessibility metric. Considering more detailed supply distribution decisions (in addition to the decisions on the deliveries from the LDC to the PODs we also determine the amounts of supplies delivered from the PODs to the demand points) allows us to obtain this finer accessibility metric. In order to be consistent with our cost minimization setup, we use the convention that lower accessibility scores indicate higher accessibility and define the accessibility scores as the weighted travel times. These accessibility scores correspond to the reciprocals of those used in Noyan et al. (2016).

We consider two types of equity: equitable accessibility and equitable supply allocation. As in Noyan et al. (2016), we ensure equitable accessibility to the PODs from the demand locations by defining the coverage sets according to a maximum threshold requirement in terms of the accessibility scores associated with the links. However, we take a different approach to modeling equity in supply allocation. Noyan et al. (2016) hybridize two supply allocation policies based on quantitative measures: a strict proportional allocation policy referred to as the PD Policy, which divides the available supply among the PODs in proportion to the total demands assigned to the PODs under each scenario, and the TD policy, which ensures that the shortage amount at each POD does not exceed a specified proportion of the corresponding total demand under each scenario. In particular, their hybrid approach enforces the TD policy and a relaxed version of the PD policy, and penalizes the deviations from the strict PD policy in the objective function in order to distribute the supplies among the PODs in proportion to their total demands as much as possible without compromising from the expected total accessibility. In contrast, in this study, we consider the supply allocations at the demand level instead of the POD level due to the focus on more detailed supply distribution decisions. This property of our model renders the existing TD and PD policies not directly applicable to our setup because the targeted demand levels in our demand satisfaction constraints are the individual

demands at each location. These demand values are input parameters while the total demands assigned to the PODs are decision variables implied by the assignment decisions. In our setting, we incorporate both a qualitative and a quantitative measure regarding the stochastic demand satisfaction constraints. As the qualitative measure, we introduce individual chance constraints on satisfying the demands at each location, where the lower bound on the reliability level is set to be equal for each demand location. The quantitative measure is in the form of (2.20) and closely related to the proportion of unsatisfied demand at each point. Focusing on such a proportional measure is in itself promising in terms of serving the population groups equitably. These approaches lead to a new hybrid supply allocation policy which enforces a set of individual chance constraints on satisfying the demands at each node and penalizes the cost of reliability associated with these demand satisfaction constraints.

2.3.3 Stochastic Optimization Model

We consider a network where each node represents a geographical area (a settlement such as a village and a town) according to the size of the affected region. We denote the set of demand nodes by I and the set of candidate PODs by J, where we assume that $J \subseteq I$ without loss of generality. The randomness in the node demands and the accessibility scores associated with the links of the network are represented by a finite set of scenarios. Taking a conservative approach, a POD can cover (serve) a demand node only if the worst (largest) accessibility score to the POD from the demand location under any scenario is no larger than a maximum threshold $\bar{\tau}$. Next, we provide the list of the additional input parameters:

p^i: probability of scenario $i,\ i \in S$,

ξ_k^i: demand at node k under scenario i, $k \in I,\ i \in S$; the corresponding random variable is denoted by ξ_k,

ν_{0j}^i: score for accessibility to candidate POD j from the LDC under scenario i, $j \in J$, $i \in S$,

ν_{kj}^i: score for accessibility to candidate POD j from demand node k under scenario i, $k \in I,\ j \in J,\ i \in S$ with $\nu_{jj}^i = 0$ for all $j \in J,\ i \in S$,

$N_k = \{j \in J \mid \max_{i \in S} \nu_{kj}^i \leq \bar{\tau}\}$: set of candidate PODs that can cover demand node k, $k \in I$,

$M_j = \{k \in I \mid \max_{i \in S} \nu_{kj}^i \leq \bar{\tau}\}$: set of demand nodes that can be covered by the candidate POD j, $j \in J$,

κ: maximum number of PODs to be opened ($\kappa \leq |J|$),

Θ: the total amount of available supplies,

K_j: an upper bound on the amount of supplies that can be delivered to POD j.

In our modeling framework, the network design decisions are made before the uncertainty in the parameters is resolved. We use the notation below for the network design decisions of interest:

- $y_j = 1$ if a POD is located at node $j \in J$, and $y_j = 0$ otherwise,
- $x_{kj} = 1$ if demand node $k \in I$ is served by POD $j \in N_k$, and $x_{kj} = 0$ otherwise,
- r_{kj} : the amount of supplies delivered to demand node k from POD $j \in J$.

Then, the amount of supplies delivered to POD j from the LDC is given by $\sum_{k \in M_j} r_{kj}$.

In our modeling approach, we ensure equitable accessibility by serving each demand point by a POD in its coverage set defined based on a common upper bound on the accessibility scores associated with the links. Regarding accessibility, we also design an efficiency-related objective function featuring the total accessibility. We note that the upper bound parameter $\bar{\tau}$ can be used to balance the trade-off between the equity as measured by the worst possible accessibility score and efficiency as quantified by the total accessibility. Moreover, we additionally take into account equity in supply allocation by enforcing a set of individual chance constraints to distribute the relief supplies to the demand points in such a way that the probability of satisfying the demand at each node is at least equal to a common a lower bound $(1 - \bar{\epsilon})$. In addition to this qualitative measure on the level of demand satisfaction, we incorporate a quantitative measure – based on the deviations from the desired levels of delivery – into the objective function to control for the demand shortage amounts. Following the modeling approach proposed in Section 2.2.2, we set the desired level of delivery for node k as $\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)$, $k \in I$, and calculate the proportion of unsatisfied demand (PUD) based on these lower threshold VaR values. This modeling approach based on the variable reliability levels allows us to balance the trade-off between the total accessibility and the cost of reliability associated with the demand satisfaction in the objective function. We next present the chance-constrained optimization model developed for our stochastic relief network design problem:

$$(\mathbf{CCSRND})$$

$$\begin{aligned}
\text{minimize} \quad & \sum_{i \in S} \sum_{j \in J} p^i \nu^i_{0j} \sum_{k \in M_j} r_{kj} + \sum_{i \in S} \sum_{k \in I} \sum_{j \in N_k} p^i \nu^i_{kj} r_{kj} \\
& + \sum_{k \in I} \sum_{i \in [i^*_k - 1]} p^i_k \left(\frac{a}{\xi^i_k} \left[\xi^i_k - \mathrm{VaR}_{(1-\epsilon_k)}(\xi_k) \right]_+ \right) && \text{(2.23a)} \\
\text{subject to} \quad & \sum_{j \in J} y_j \leq \kappa, && \text{(2.23b)}
\end{aligned}$$

$$\sum_{k \in M_j} r_{kj} \leq K_j y_j, \quad \forall\, j \in J \tag{2.23c}$$

$$\sum_{j \in J} \sum_{k \in M_j} r_{kj} = \Theta, \tag{2.23d}$$

$$\sum_{j \in N_k} r_{kj} \leq \max_{i \in S} \xi_k^i, \quad \forall\, k \in I \tag{2.23e}$$

$$\sum_{j \in N_k} x_{kj} = 1, \quad \forall\, k \in I \tag{2.23f}$$

$$x_{kj} \leq y_j, \quad \forall\, k \in I,\ j \in N_k \tag{2.23g}$$

$$x_{jj} \geq y_j, \quad \forall\, j \in J \tag{2.23h}$$

$$r_{kj} \leq x_{kj} U_{kj}, \quad \forall\, k \in I,\ j \in N_k \tag{2.23i}$$

$$P\left(\sum_{j \in N_k} r_{kj} \geq \xi_k\right) \geq 1 - \epsilon_k, \quad \forall\, k \in I \tag{2.23j}$$

$$0 \leq \epsilon_k \leq \bar{\epsilon}, \quad \forall\, k \in I \tag{2.23k}$$

$$y_j \in \{0, 1\}, \quad \forall\, j \in J, \tag{2.23l}$$

$$x_{kj} \in \{0, 1\}, \quad \forall\, k \in I,\ j \in N_k \tag{2.23m}$$

$$r_{kj} \geq 0, \quad \forall\, k \in I,\ j \in N_k. \tag{2.23n}$$

The objective function (2.23a) minimizes the expected total delivery amount-weighted accessibility score (ETWA) and the expected cost of the risk tolerances $\epsilon_k,\ k \in I$, where the expected values are estimated by sample averaging. The second cost term is based on the VaR-based PUD values referred to as V-PUD. Thus, the expected cost of the reliability levels is specified as a linear function of the expected total V-PUD. Considering the V-PUD values and assigning a common penalty coefficient a to all demand points are consistent with the intent of fair supply allocation. Constraint (2.23b) ensures that the number of established PODs is not larger than the specified limit κ. Constraints (2.23c) play a dual role by imposing a maximum capacity limit on the amount of supplies delivered to each open POD and mandating that there is a POD located at node j if there is any delivery to that POD. According to the constraints (2.23d) and (2.23e), all available supplies are distributed in the network while the total supply amount allocated to a demand node does not exceed its largest possible demand realization. Here, the underlying assumption is that $\Theta \leq \sum_{k \in I} \max_{i \in S} \xi_k^i$. This relation may only not hold in rare and uninteresting cases when the relief supplies in the last mile network are ample compared to the total demand. Constraints (2.23f) and (2.23g) guarantee that each demand node is assigned to a single open POD – a policy known as single sourcing and implemented commonly in practice. A demand node with an open POD should naturally be served by this POD – as is also favored by the objective coefficients $\nu_{jj}^i = 0,\ j \in J,\ i \in S$. However, under extremely restrictive capacity limits, the optimal solution may not obey this rule.

To avoid such peculiar solutions, we embed this rule explicitly via constraints (2.23h). Constraints (2.23i) set the delivery amounts to the demand nodes from their respective assigned PODs. The delivery amount to node k is naturally bounded from above by the capacity of its corresponding POD and the maximum possible demand realization at node k; that is, $U_{kj} := \min\{K_j, \max_{i\in S} \xi_k^i\}$ for all $k \in I$ and $j \in N_k$. The constraints (2.23j) and (2.23k) collectively prescribe that the probability of satisfying demand at each demand node is at least $1-\epsilon_k \geq 1-\bar{\epsilon}$. The rest of the constraints enforce the non-negativity and binary restrictions.

Drawing upon the tools of Sections 2.1 and 2.2 in order to reformulate the individual chance constraints (2.23j) and the cost of reliability in the objective function (2.23a) yields the MIP reformulation of $(\mathbf{CCSRND})$ below:

$$\begin{aligned}
\text{minimize} \quad & \sum_{i\in S}\sum_{j\in J} p^i \nu_{0j}^i \sum_{k\in M_j} r_{kj} + \sum_{i\in S}\sum_{k\in I}\sum_{j\in N_k} p^i \nu_{kj}^i r_{kj} \\
& + \sum_{k\in I}\sum_{i\in[i_k^*-1]} p_k^i \left(\frac{a}{\xi_k^i} w_{ki}\right) && \text{(2.24a)} \\
\text{subject to} \quad & \text{(2.23b)} - \text{(2.23i)},\ \text{(2.23k)} - \text{(2.23n)}, && \text{(2.24b)} \\
& \sum_{j\in N_k} r_{kj} \geq z_k, \qquad \forall\, k \in I && \text{(2.24c)} \\
& \text{(2.12c)} - \text{(2.12e)},\ \text{(2.12h)},\ \text{(2.19c)} - \text{(2.19e)}, \text{(2.21)} - \text{(2.22)}. && \text{(2.24d)}
\end{aligned}$$

The formulation (2.24) is precisely in the form of $(\mathbf{SMICC-VRTQA}-\boldsymbol{\beta})$. The one to one relation is easily observed by noting that $[m]$ and $[N]$ in $(\mathbf{SMICC-VRTQA}-\boldsymbol{\beta})$ correspond to I and S in $(\mathbf{CCSRND})$, respectively.

Remark 1 *As discussed in Section 2.1, additional constraints on the variable risk tolerance decisions may also be included in* (2.24)*. For instance, it would be natural to bound the summation of the risk tolerances as in* (2.16c)*, where the generic set* $[m]$ *is replaced by* I*. It is easy to show that adding* (2.16c) *to the formulation* (2.24) *guarantees that the joint chance constraint* $\mathbb{P}\left(\sum_{j\in N_k} r_{kj} \geq \xi_k, \quad \forall\, k \in I\right) \geq 1-\Gamma$ *holds for any of its feasible solutions. If the number of individual chance constraints is large, such a joint chance constraint can be very demanding for* $\Gamma < 1$*. In practical applications, setting* $\Gamma \geq 1$ *in* (2.16c) *may therefore be preferable – ignoring the connection with the joint chance constraint mentioned above. We also take this path in Section 2.4.*

For the post-disaster application in focus, the proposed modeling approach grants a flexibility to balance the trade-off between the accessibility captured by ETWA and the level of demand satisfaction. One can impact this trade-off by changing the values of the parameters a and $\bar{\epsilon}$. We intend to provide more insights about the model by presenting numerical results on a case study in Section 2.4.3.

2.4 Computational Study

In the computational study, we fulfill the objectives and claims set forth in the previous sections. This is accomplished in two parts. In the first part, we demonstrate the computational effectiveness of the proposed MIP formulations for the two types of chance-constrained optimization models of interest ($\mathbf{MICC-VRT}$) and ($\mathbf{MICC-VRTQ}$). For ($\mathbf{MICC-VRT}$), we provide clear evidence that the new formulations ($\mathbf{SMICC-VRT-}\boldsymbol{\beta}$) and ($\mathbf{SMICC-VRT-y}$) outperform the formulation ($\mathbf{MICC-VRT-DEF}$) due to Shen (2014) by a significant margin. The impact of the alternate VaR representations in Lemma 2 on the computational performance is investigated in the context of ($\mathbf{MICC-VRTQ}$) by benchmarking ($\mathbf{SMICC-VRTQ-}\boldsymbol{\beta}$) against its counterpart with the alternate VaR representation (2.11). The runs in this part also reveal that the formulations based on the $\beta-$ and $y-$variables may indeed exhibit disparate performances. In the second part, we illustrate the application of the proposed stochastic relief network design model ($\mathbf{CCSRND}$) on a case study developed based on the real-world data from the 2011 Van earthquake in Turkey. The take away from the numerical results here is that solving large practical instances to optimality is within the reach of the proposed formulations.

All runs are executed on a personal computer with a 3.2 GHz Intel® Core™ i7 960 processor and 24 GB of memory running on Windows 7. All MIP formulations are implemented and solved in `C++` using the `Concert Technology` component library of `IBM® ILOG® CPLEX® 12.6`. `CPLEX` is invoked with a time limit of 3600 seconds and its default set of options and parameters. All reported times are elapsed times in seconds. If optimality is not proven within the allotted time, we record both the best lower bound LB on the optimal objective value retrieved from `CPLEX` and the objective function value associated with the incumbent at termination – denoted by UB. Then, we calculate an upper bound on the relative optimality gap of the incumbent as $\mathrm{ROG} = (UB - LB)/LB$.

2.4.1 Generation of the Problem Instances

We performed our computational tests on two types of problems: a probabilistic version of the classical transportation problem with random demand (Luedtke et al., 2010) and the stochastic last mile relief network design problem with random demand and accessibility scores presented in Section 2.3.

For the first problem type, we use the data sets of Luedtke et al. (2010) with "general probabilities," as labeled by the authors. In these instances, the scenarios may have different probabilities. The authors randomly generated instances with the number of suppliers fixed at 40 and varying numbers of customers ($n = 100,\ 200$) and scenarios

($|S| = 1000,\ 2000$ for instances with 100 customers and $|S| = 2000,\ 3000$ for instances with 200 customers). Further details are available in the original paper. We refer to these problem instances as "transportation (TR) instances" and note that for some analyses we also created smaller TR instances with 100 customers by deleting scenarios from the original instances with $n = 100$ and $|S| = 1000$.

For the second problem type, we focus on the Van earthquake relief case study developed by Noyan et al. (2016). We used the readily available instances and also generated additional larger instances by following their data generation scheme. The network in this case study contains 94 demand points, each representing one settlement in the main district of Van. The authors also construct networks with 30 and 60 nodes by clustering the original set of settlements via a P-median model that minimizes the demand-weighted travel times. We refer to these problem instances as "relief network design (RND) instances". For the details of the data collection and scenario generation procedures, we refer the reader to Noyan et al. (2016) and the associated online supplement.

Here, we briefly recall some parameters from Noyan et al. (2016) and comment on the parameters whose values are specified differently from those in the original paper. The amount of available supplies Θ is set as 110% of the expected total demand. The parameter κ is selected as 6, 9, and 12 for the networks with 30, 60, and 94 nodes, respectively. The tightness of the capacity upper bounds are varied to illustrate their impact. In particular, we set $K_j = c\hat{\xi}_j$, where $\hat{\xi}_j$ denotes the estimated base value of the demand at node $j \in J$, and c is selected from the range $[1.75, 8]$. Another influential parameter is the upper bound $\bar{\tau}$ on the accessibility scores, which determines the size of the coverage sets, and is calculated as $\bar{\tau} = 1/\tau$. To avoid overly small coverage sets, τ is selected from the range $[0.01, 0.15]$.

2.4.2 Computational Performance

We summarize the common elements in the design of the numerical experiments in this section before proceeding with the performance analysis. First, recall that in general two types of constraints are imposed on the variable risk tolerances $\epsilon_k,\ k \in [m]$ – the constraints (2.6d) and the simple upper bound constraints (2.6e). Unless otherwise stated, here all formulations only include the latter, and the parameter $\bar{\epsilon}$ – the upper bound on the risk tolerances – is set as 0.3 for both types of problem instances. In some analyses with the TR instances, the risk budget constraint (2.16c) is incorporated in addition to the constraints (2.6e). In these experiments, the set of constraints (2.6d) takes the form of (2.16c), and Γ is set as 3 and 6 for the instances with 100 and 200 demand nodes, respectively. Second, for each combination of the parameters in the tables in this section, three instances were generated by following the instructions in Section 2.4.1. All results presented in the tables are therefore averaged over three instances. Third, unless otherwise

stated, CPLEX is allowed to use up to four parallel threads as specified by its Threads parameter. Finally, note that $(\mathbf{MICC-VRT-DEF})$, $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ and $(\mathbf{SMICC-VRT-y})$ are respectively shortened as $(\mathbf{VRT-DEF})$, $(\mathbf{VRT-}\boldsymbol{\beta})$ and $(\mathbf{VRT-y})$ in the column headers of the tables for ease of presentation.

Evaluation of the Alternate Formulations of $(\mathbf{MICC-VRT})$**.** In this subsection, we assess how our proposed formulations $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ and $(\mathbf{SMICC-VRT-y})$ fare against $(\mathbf{MICC-VRT-DEF})$ presented by Shen (2014). Here, we deliberately stick with the formulation $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ for the $\beta-$variables because the VaR representation embedded in this formulation and the VaR representation employed by Shen (2014) in $(\mathbf{MICC-VRT-DEF})$ both rely on Lemma 1-(iii). Thus, any difference in performance can mainly be attributed to how the quantiles are captured in the reformulation of the chance constraints based on (2.1).

		Time [ROG (%)]					
		Budget constraint (2.16c) **omitted**			**Budget constraint** (2.16c) **included**		
n	$\lvert S\rvert$	$(\mathbf{VRT-DEF})$	$(\mathbf{VRT-}\boldsymbol{\beta})$	$(\mathbf{VRT-y})$	$(\mathbf{VRT-DEF})$	$(\mathbf{VRT-}\boldsymbol{\beta})$	$(\mathbf{VRT-y})$
100	50	†††[53.4]	0.1	0.1	†††[38.8]	0.1	0.1
	100	†††[67.7]	0.2	0.2	†††[59.0]	0.2	0.2
	200	†††[77.7]	0.2	0.3	†††[76.0]	0.2	0.2
	500	†††[86.3]	0.4	0.5	†††[86.3]	0.4	0.4
	1000	†††[90.6]	0.9	0.9	†††[90.6]	0.8	0.8
	2000	†††[93.5]	1.9	1.7	†††[93.5]	2.0	1.5
200	2000	†††[94.6]	4.4	3.3	†††[94.5]	4.9	3.0
	3000	†††[95.9]	8.7	5.2	†††[95.9]	9.0	4.4
	Avg	3600 [82.5]	2.1	1.5	3600 [79.3]	2.2	1.3

†: Each dagger sign indicates one instance hitting the time limit with an integer feasible solution. A single thread is used in these experiments.

Table 2.2: Computational performance of the alternate MIPs for $(\mathbf{MICC-VRT})$ on the TR instances.

The computational effectiveness of all three formulations is assessed on the two benchmark sets specified in Section 2.4.1. The results for the TR and RND instances are depicted in Tables 2.2 and 2.3, respectively. In these tables, we report the average solution times and ROG values. The ROG information is skipped altogether if all three instances associated with a cell terminate with an optimal solution. The values of the parameter a are specified as 1,000,000 and 100,000 for the TR and RND instances, respectively. For the RND instances, the values of the parameters (τ, c) are a major determinant of instance difficulty as demonstrated in the subsequent sections. Here, (τ, c) are specified as (0.01, 2.00), (0.05, 2.25), and (0.09, 2.75) for the instances with $|I| = 30, 60$, and 94 nodes, respectively.

The figures in Tables 2.2 and 2.3 reveal that $(\mathbf{MICC-VRT-DEF})$ is outperformed drastically by our MIP formulations $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ and $(\mathbf{SMICC-VRT-y})$, which generally perform on a par. No significant pattern is

evident from a comparison of their solution times. $(\mathbf{MICC-VRT-DEF})$ fails to provide a single optimal solution within the time limit except for the RND instances with $|I| = 94$ and $|S| = 50$. In stark contrast, all average solution times associated with $(\mathbf{SMICC-VRT-}\boldsymbol{\beta})$ and $(\mathbf{SMICC-VRT-y})$ are below 10 and 90 seconds in Tables 2.2 and 2.3, respectively, except for the large RND instances with $|I| = 94$ and $|S| = 200$. For illustrative purposes, in Table 2.2 we also investigate the sensitivity of the formulations to the presence of (2.16c). We do not detect any effect and conclude that the results in Table 2.2 are robust with respect to this type of coupling risk budget constraint.

	Time [ROG (%)]								
	$\|I\| = 30$			$\|I\| = 60$			$\|I\| = 94$		
$\|S\|$	$(\mathbf{VRT-DEF})$	$(\mathbf{VRT-}\boldsymbol{\beta})$	$(\mathbf{VRT-y})$	$(\mathbf{VRT-DEF})$	$(\mathbf{VRT-}\boldsymbol{\beta})$	$(\mathbf{VRT-y})$	$(\mathbf{VRT-DEF})$	$(\mathbf{VRT-}\boldsymbol{\beta})$	$(\mathbf{VRT-y})$
50	$\dagger\dagger\dagger$ [36.8]	0.6	0.6	$\dagger\dagger\dagger$ [50.2]	40.7	47.8	164.4	7.7	8.2
100	$\dagger\dagger\star$ [68.2]	0.8	1.2	$\dagger\dagger\dagger$ [36.7]	48.7	39.0	$\dagger\dagger\dagger$ [26.7]	38.4	81.2
200	$\dagger\dagger\star$ [70.7]	1.1	1.2	$\dagger\dagger\dagger$ [39.5]	6.8	5.7	$\dagger\dagger\star$ [48.8]	596.8	1040.7
500	$\dagger\dagger\dagger$ [62.6]	1.7	1.5	$\dagger\dagger\dagger$ [49.8]	15.9	15.0	$\dagger\star\star$ [78.3]	9.8	9.4
Avg	3600 [59.6]	1.0	1.1	3600 [44.1]	28.0	26.9	2742.5 [38.4]	163.2	284.9

$\dagger$: Each dagger sign indicates one instance hitting the time limit with an integer feasible solution.
$\star$: Each asterisk sign indicates one instance hitting the time limit with no integer feasible solution. In this case, we assume that the associated ROG value is 100%.

Table 2.3: Computational performance of the alternate MIPs for $(\mathbf{MICC-VRT})$ on the RND instances.

Evaluation of the Alternate Formulations of $(\mathbf{MICC-VRTQ})$. In this subsection, we rate the performance of the formulations available for $(\mathbf{MICC-VRTQ})$. For the RND instances, the cost of reliability $h_k(z_k)$ in the objective function is specified as in (2.20), where the coefficients a_{ki} are defined as $\frac{a}{\xi_k^i}$. For diversity, we employ the term $h_k(z_k) = -a_k z_k = -a z_k$ for the TR instances. We observe that the VaR values can be significantly larger than the values of the reliability levels. Therefore, we determine the

		Time	
n	$\|S\|$	**VaR Rep** (2.10)	**VaR Rep** (2.11)
100	2000	13.9	78.2
200	2000	139.2	357.3
200	3000	199.3	753.1
	Avg	117.5	396.2

Table 2.4: Computational performance of the alternate VaR representations on the TR instances.

values of the parameter a by running some preliminary tests in order to ensure that the values of the risk tolerances in the optimal solutions are not necessarily always at the end

points of the range $[0,\ \bar{\epsilon}]$. For the analyses with the TR and RND instances, the parameter a assumes the values 50 and 1,000,000, respectively.

	Time					
	$\|I\| = 30$		$\|I\| = 60$		$\|I\| = 94$	
$\|S\|$	**Rep** (2.10)	**Rep** (2.11)	**Rep** (2.10)	**Rep** (2.11)	**Rep** (2.10)	**Rep** (2.11)
1000	18.2	18.7	101.1	81.0	102.9	108.8
2000	24.6	31.6	146.3	127.9	113.7	106.1
3000	38.9	49.5	133.4	221.2	187.3	155.2
5000	58.1	60.6	217.1	218.8	311.2	296.9
Avg	35.0	40.1	149.5	162.2	178.8	166.8

$\tau = 0.01, 0.05, 0.09$ and $c = 2.00, 2.25, 2.75$, for the instances with 30, 60 and 94 nodes, respectively.

Table 2.5: Computational performance of the alternate VaR representations on the RND instances.

Our first goal in this section is to expose the computational differences between the alternate VaR representations listed in Lemma 2 for our formulations based on the $\beta-$variables and stick to one of these for the rest of the computational study. To this end, we assess the performance of $(\mathbf{SMICC - VRTQ - \boldsymbol{\beta}})$ against that of its counterpart with the alternate VaR representation (2.11). The largest TR and RND instances are all solved to optimality with both representations, and the average solution times are presented in Tables 2.4 and 2.5, respectively. In this analysis, the risk budget constraint (2.16c) is included for the runs with the TR instances. As pointed out following the proof of Lemma 2, (2.11) is a stronger representation compared to (2.10) from a theoretical viewpoint. However, it appears that this fact does not necessarily translate into computational gains, and the results in Tables 2.4-2.5 refute the expectation that the representation (2.11) should consistently lead to smaller solution times. In complete contrast, the average solution times associated with (2.10) are substantially smaller in comparison to those obtained with (2.11) on the TR instances. For the RND instances, the picture is mixed. The presentation (2.10) has a slight edge over (2.11) on the smaller instances with $I = |30|$, while the reverse is true for the larger instances with $I = |94|$. No definitive pattern emerges for the remaining instances with $I = |60|$. We decide that the presentation (2.10) delivers a better performance overall, and going forward the VaR representation (2.10) is incorporated into the formulations $(\mathbf{SMICC - VRTQ - \boldsymbol{\beta}})$ and $(\mathbf{SMICC - VRTQA - \boldsymbol{\beta}})$ for the TR and RND instances, respectively.

		Time			
		Budget constraint (2.16c) **omitted**		**Budget constraint** (2.16c) **included**	
n	$\lvert S\rvert$	$(\mathbf{S-VRTQ-y})$	$(\mathbf{S-VRTQ-}\boldsymbol{\beta})$	$(\mathbf{S-VRTQ-y})$	$(\mathbf{S-VRTQ-}\boldsymbol{\beta})$
100	200	0.5	0.7	0.8	1.5
	500	1.0	1.2	1.6	6.0
	1000	1.6	0.9	3.1	26.4
	2000	2.5	1.5	5.2	13.9
200	2000	5.1	5.7	20.7	139.2
	3000	7.6	9.1	14.5	199.3
	Avg	3.1	3.2	7.7	64.4

Table 2.6: Computational performance of the alternate MIPs for $(\mathbf{MICC-VRTQ})$ on the TR instances.

Next, we turn our attention to whether the formulations based on the $\beta-$ and $y-$variables act similarly as they do in the previous section. The results for $(\mathbf{SMICC-VRTQ-y})$ and $(\mathbf{SMICC-VRTQ-}\boldsymbol{\beta})$ on the TR instances are provided in Table 2.6, where we distinguish between the presence and absence of the risk budget constraint (2.16c). Both formulations terminate with an optimal solution within the time limit for all instances, but there exists a marked difference in comparison to the results in the previous subsection in one important aspect. The two formulations are virtually indistinguishable in the absence of the budget constraint. However, the detrimental effect of the inclusion of (2.16c) on $(\mathbf{SMICC-VRTQ-}\boldsymbol{\beta})$ is much more substantial than on $(\mathbf{SMICC-VRTQ-y})$. For the RND instances, $(\mathbf{SMICC-VRTQA-y})$ is benchmarked against $(\mathbf{SMICC-VRTQA-}\boldsymbol{\beta})$ – recall that the latter boils down to the formulation $(\mathbf{CCSRND})$ presented in Section 2.3.3. Table 2.7 depicts the outcome, where the common string "SMICC – VRT" is dropped from the formulation names for brevity. The results in Table 2.7 also control for the values of the combination of the parameters τ and c, which play a pivotal role in the design of the relief networks. Demanding values of these parameters might lead to infeasible solutions – see Section 2.4.3, and they also affect the solution times, as is evident from Table 2.7. In any case, all instances are solved to optimality with both formulations; however, there is a shift in performance with increasing problem size. For $|S| = 500, 1000$, $(\mathbf{SMICC-VRTQA-y})$ is somewhat better; however, $(\mathbf{SMICC-VRTQA-}\boldsymbol{\beta})$ starts to dominate for larger number of scenarios.

One critical insight that needs to come across out of this computational study is that it is essential to experiment with the different formulations and VaR representations laid out in this chapter. The performance may very much be dependent on the problem and instance settings as clearly illustrated in this section.

		Time									
		$\|S\| = 500$		$\|S\| = 1000$		$\|S\| = 2000$		$\|S\| = 3000$		$\|S\| = 5000$	
$\|I\|$	(τ, c)	$(\mathbf{QA} - \mathbf{y})$	$(\mathbf{QA} - \boldsymbol{\beta})$	$(\mathbf{QA} - \mathbf{y})$	$(\mathbf{QA} - \boldsymbol{\beta})$	$(\mathbf{QA} - \mathbf{y})$	$(\mathbf{QA} - \boldsymbol{\beta})$	$(\mathbf{QA} - \mathbf{y})$	$(\mathbf{QA} - \boldsymbol{\beta})$	$(\mathbf{QA} - \mathbf{y})$	$(\mathbf{QA} - \boldsymbol{\beta})$
30	(0.01, 1.75)	18.0	15.5	17.6	30.2	27.2	33.1	65.5	63.8	142.0	98.9
	(0.01, 2.00)	9.0	7.1	10.9	18.2	25.3	24.6	46.2	38.9	132.2	58.1
	(0.02, 1.75)	4.4	7.6	9.6	18.8	22.8	26.8	54.5	42.7	156.0	71.0
	(0.02, 2.00)	4.7	10.3	10.6	13.0	23.0	18.6	53.0	37.2	139.9	41.2
60	(0.05, 2.00)	33.6	45.4	44.3	61.0	122.1	77.2	211.1	110.7	676.3	150.6
	(0.05, 2.25)	52.3	60.9	82.0	101.1	221.0	146.3	254.0	133.4	701.9	217.1
	(0.06, 2.00)	30.7	55.5	46.6	66.1	125.1	111.1	258.2	100.2	670.4	206.9
	(0.06, 2.25)	85.1	83.2	391.0	74.8	169.5	120.3	1064.0	201.2	700.7	200.0
94	(0.09, 2.25)	80.5	99.2	182.7	209.2	195.1	194.6	400.0	295.5	953.2	496.8
	(0.09, 2.75)	40.5	64.8	65.5	102.9	111.5	113.7	284.8	187.3	1124.1	311.2
	(0.11, 2.25)	43.2	61.9	69.1	74.4	117.1	95.4	291.1	246.8	1139.6	388.2
	(0.11, 2.75)	28.6	47.0	47.7	80.5	89.8	101.9	261.8	136.3	871.4	264.8
	Avg	35.9	46.5	81.5	70.9	104.1	88.6	270.3	132.8	617.3	208.7

Table 2.7: Computational performance of the alternate MIPs for $(\mathbf{MICC - VRTQ})$ with (2.20) on the RND instances.

2.4.3 Model Analysis

In this section, we first demonstrate the impact of the input parameters c, τ, a, and $\bar{\epsilon}$ on the trade-off between the accessibility and the level of demand satisfaction. This is followed by a comparison of the solutions attained by $(\mathbf{CCSRND})$ against those retrieved from a set of benchmark models with the end goal of highlighting the value of the proposed hybrid supply allocation policy. All analyses in this section are performed on the same base instance with $|I| = 30, |S| = 50$.

		$\tau = 0.01$		$\tau = 0.02$		$\tau = 0.03$		$\tau = 0.04$	
c	a	ETWA	ETV-PUD	ETWA	ETV-PUD	ETWA	ETV-PUD	ETWA	ETV-PUD
2	0×10^5	716,460	0.4913	716,460	0.4913	-	-	-	-
	1×10^5	726,165	0.2437	726,419	0.2408	-	-	-	-
	5×10^5	765,978	0.0842	765,853	0.0844	-	-	-	-
	10×10^5	777,922	0.0658	777,922	0.0658	-	-	-	-
	50×10^5	875,154	0.0165	945,378	0.0116	-	-	-	-
3	0×10^5	616,128	0.4913	616,128	0.4913	616,128	0.4913	-	-
	1×10^5	630,467	0.1557	630,461	0.1558	630,404	0.1564	-	-
	5×10^5	661,667	0.0216	661,673	0.0216	661,650	0.0217	-	-
	10×10^5	665,034	0.0165	665,424	0.0161	665,001	0.0166	-	-
	50×10^5	668,480	0.0146	668,480	0.0146	668,480	0.0146	-	-
4	0×10^5	612,163	0.4913	612,163	0.4913	612,163	0.4913	-	-
	1×10^5	625,590	0.1554	625,590	0.1554	625,590	0.1554	-	-
	5×10^5	652,204	0.0208	652,412	0.0204	652,402	0.0205	-	-
	10×10^5	657,747	0.0130	657,335	0.0134	657,747	0.0130	-	-
	50×10^5	660,239	0.0116	660,239	0.0116	660,239	0.0116	-	-
8	0×10^5	581,449	0.4913	581,449	0.4913	581,449	0.4913	592,197	0.4913
	1×10^5	591,662	0.1944	591,739	0.1937	591,464	0.1964	604,498	0.1603
	5×10^5	627,514	0.0235	627,514	0.0235	627,514	0.0235	634,141	0.0214
	10×10^5	631,732	0.0165	631,746	0.0165	631,746	0.0164	640,656	0.0124
	50×10^5	642,051	0.0116	642,051	0.0116	642,051	0.0116	642,051	0.0116

-: No feasible solution.

Table 2.8: Values of the objective function terms at optimality for varying parameter settings ($\bar{\epsilon} = 0.3$).

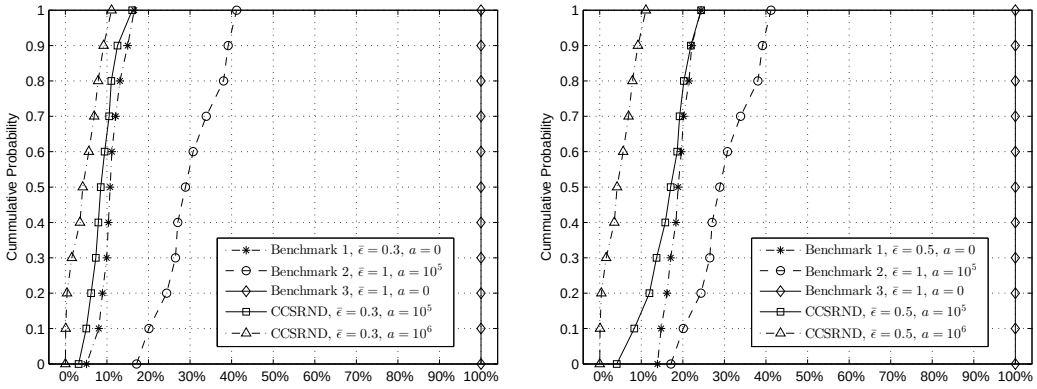
The parameters c and τ respectively determine the level of restrictiveness of the capacity limits and the coverage sets while the parameters a and $\bar{\epsilon}$ play critical roles in controlling the trade-off between accessibility and the cost of reliability associated with demand satisfaction. The effects of these parameters on the two conflicting objective function criteria ETWA and the expected total V-PUD – denoted by ETV-PUD – are highlighted in Table 2.8, where we specifically explore the interactions among c, τ, and a by keeping $\bar{\epsilon}$ fixed at 0.30. Larger values of c imply less restrictive capacity limits and clearly lead to reductions in the optimal objective function value ETWA + a ETV-PUD. These improvements generally translate to reductions in both of the objective function criteria; however, for a number of cases when c is increased from 4 to 8, the performance degrades with respect to ETV-PUD. We also observe that the deterioration in ETWA as a is increased is

more pronounced for $c = 2$ in comparison to the higher values of c. Another clear insight from Table 2.8 is that smaller values of c and larger values of τ can result in infeasibilities. However, when the value of τ is not large enough to cause infeasibilities, its effect appears to be marginal. To investigate the impact of the parameter $\bar{\epsilon}$, we analyze the attained reliability levels at each demand point under varying values of $\bar{\epsilon}$ and a. The values of c and τ are set as 2 and 0.01, respectively. In Table 2.9, the demand nodes are classified into six categories depending on their attained reliability levels / risk tolerances. To clarify, those demand nodes with a minimum attained risk tolerance of $0.95 * 0.20 = 0.19$ or $0.95 * 0.50 = 0.475$ are assigned to the row labeled as "$(95\%\text{-}100\%)\bar{\epsilon}$" if $\bar{\epsilon} = 0.20$ or $\bar{\epsilon} = 0.50$, respectively. On the one hand, small values of $\bar{\epsilon}$ already appear to be very effective in ensuring lower costs of reliability. Therefore, the risk tolerance decisions are less sensitive to the changes in the value of a if $\bar{\epsilon} = 0.2$. For $\bar{\epsilon} = 0.35, 0.50$ on the other hand, the expected trend is clearly present – as a assumes larger values the number of demand nodes in the groups with lower risk tolerances increases. In any case, the optimization fully exploits the flexibility in setting the risk tolerances, and the attained values are distributed over the allowable range $[0, \bar{\epsilon}]$.

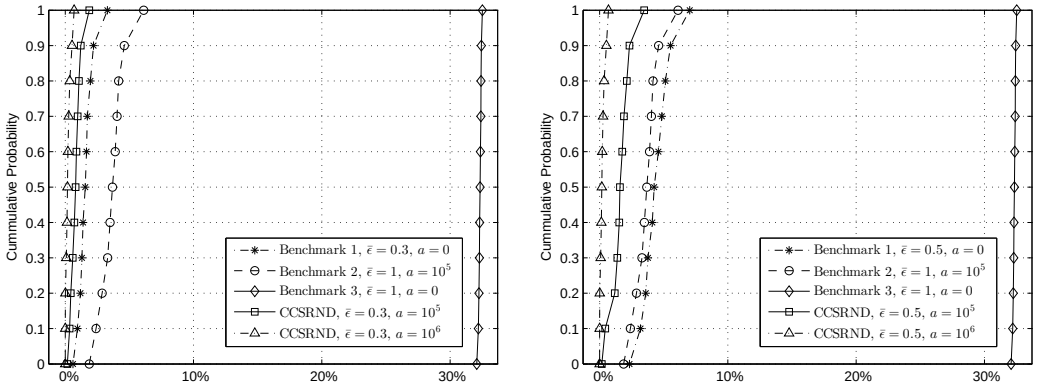
	# of demand nodes											
	$\bar{\epsilon} = 0.20$				$\bar{\epsilon} = 0.35$				$\bar{\epsilon} = 0.50$			
	a ($\times 10^5$)				a ($\times 10^5$)				a ($\times 10^5$)			
	1	2.5	5	10	1	2.5	5	10	1	2.5	5	10
$(0\% - 5\%)\bar{\epsilon}$	0	0	0	0	0	0	2	15	1	5	9	20
$(5\% - 25\%)\bar{\epsilon}$	3	3	3	4	7	13	19	10	9	13	15	5
$(25\% - 50\%)\bar{\epsilon}$	4	3	5	4	7	6	4	0	10	7	4	3
$(50\% - 75\%)\bar{\epsilon}$	4	8	8	7	7	7	3	3	3	3	2	2
$(75\% - 95\%)\bar{\epsilon}$	5	5	4	5	2	4	2	2	2	1	0	0
$(95\% - 100\%)\bar{\epsilon}$	14	11	10	10	7	0	0	0	5	1	0	0

Table 2.9: Summary of the attained risk tolerances (τ = 0.01 and c = 2).

We next assess the value of the proposed hybrid policy in terms of equity in supply allocation. Recall that this policy features both quantitative and qualitative measures regarding the demand satisfaction, and the parameters a and $\bar{\epsilon}$ are the two main levers for manipulating the reliability levels associated with the stochastic demand satisfaction constraints. In our analysis, we consider three benchmark models: *i*) the quantitative measure is excluded by setting $a = 0$ with $\bar{\epsilon} = 0.3$ (or $\bar{\epsilon} = 0.5$) for the qualitative measure (Benchmark 1), *ii*) the qualitative measure is excluded by setting $\bar{\epsilon} = 1$ with $a = 100,000$ for the quantitative measure (Benchmark 2), *iii*) both the quantitative and qualitative measures are excluded by setting $a = 0$ and $\bar{\epsilon} = 1$, respectively (Benchmark 3). In the second benchmark model, setting $\bar{\epsilon} = 1$ requires us to resort to a more traditional approach of accounting for the demand shortages by measuring the shortage amount at node k against



(a) MPUD ($\bar{\epsilon} = 0.3$) (b) MPUD ($\bar{\epsilon} = 0.50$)

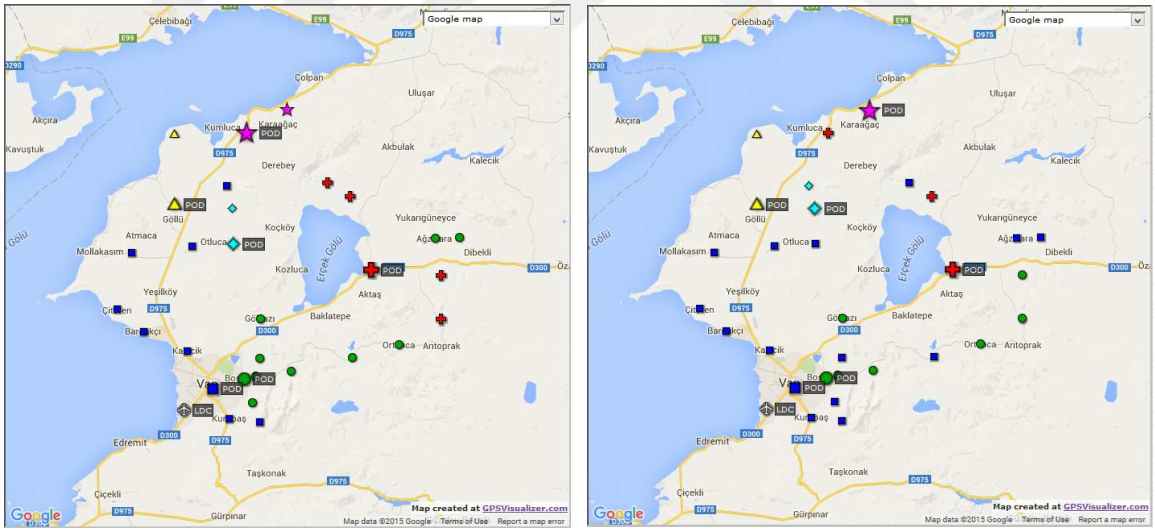


(c) APUD ($\bar{\epsilon} = 0.3$) (d) APUD ($\bar{\epsilon} = 0.50$)

Figure 2.1: Empirical cumulative distribution functions of MPUD and APUD ($\tau = 0.02, c = 2.0$).

the actual delivery amount $T_k\mathbf{x}$ instead of $\mathrm{VaR}_{(1-\epsilon_k)}(\xi_k)$. For all models involved, we compute the realizations of the maximum and average proportion of unsatisfied demand with respect to the delivery amounts – MPUD and APUD, respectively – across all demand points under each scenario. The resulting cumulative distribution functions (CDFs) are plotted in Figure 2.1. This figure illustrates that $(\mathbf{CCSRND})$ outperforms all three benchmark models with respect to both performance measures MPUD and APUD. The curves associated with $(\mathbf{CCSRND})$ always appear to the left of the rest, i.e., they are stochastically smaller than the others. For both $(\mathbf{CCSRND})$ and Benchmark 1, enforcing more restrictive chance-constraints by increasing the value of a or decreasing the value of $\bar{\epsilon}$ up to some threshold improves the reliability, and consequently, enhances the PUD-related performance measures. The interplay between a and $\bar{\epsilon}$ is delicate; an overly restrictive value of one of these parameters may render the model insensitive to the changes in the other. This is evident from the CDFs associated with $(\mathbf{CCSRND})$ for $a = 10^6$ which are virtually identical for both values of $\bar{\epsilon}$. $(\mathbf{CCSRND})$ with this setting

of a overemphasizes the quantitative measure in supply allocation and tips the trade-off in the objective function in disfavor of accessibility – as observed previously in Table 2.8. While this essentially promotes better PUD-related performance as illustrated in Figure 2.1, we do necessarily advocate it because it misses the point of balancing the trade-off between the total accessibility and the cost of reliability associated with the demand satisfaction in the objective function. The trade-off is captured much better by reducing a to 10^5. Then, increasing $\bar{\epsilon}$ from 0.3 to 0.5 leads to a right shift in the CDF associated with $(\mathbf{CCSRND})$ for both MPUD and APUD, and we can also rely on $\bar{\epsilon}$ to strike a satisfactory performance level in terms of both accessibility and reliability. Among the three benchmark models, Benchmark 1 is closest to $(\mathbf{CCSRND})$ in terms of MPUD. In the case of APUD, the ranking of Benchmarks 1 and 2 is determined by the value of $\bar{\epsilon}$ which impacts Benchmark 1 negatively. It is also fairly obvious that Benchmark 2 with a higher a value would track $(\mathbf{CCSRND})$ with $a = 10^6$ pretty closely as indicated by the insensitivity of $(\mathbf{CCSRND})$ with $a = 10^6$ to different values of $\bar{\epsilon}$. In summary, we can confidently state that the presented benchmarking analysis provides sufficient evidence for the value of our proposed framework with the embedded hybrid supply allocation policy as a flexible modeling tool to balance the trade-off between the total accessibility and the cost of reliability.



(a) MPUD (Case 1, $a = 10,000$) (b) MPUD (Case 2, $a = 10,000,000$)

Figure 2.2: Illustrative Example.

POD locations and POD to demand node assignments are marked on the map of the Van region. The LDC and the PODs are indicated with larger symbols and labeled. A POD and its assigned demand nodes share the same color. The maps are created using GPSVisualizer.com, a server that uses Google Maps to produce its output.

An illustrative example. Finally, we provide an example to illustrate how the decisions about the locations of the PODs and the assignments of the demand nodes to the PODs respond to different values of the parameter a. Our base instance is solved with $\tau = 0.02$,

$c = 2$, and $\bar{\epsilon} = 0.3$ for $a = 10,000$ and $a = 10,000,000$. The resulting POD locations and POD to demand node assignments are marked on the map of the Van region as depicted in Figure 2.2, and the details of these optimal solutions are presented in Table 2.10. We observe that four POD locations are common to both optimal solutions, but the remaining two POD locations do not coincide. Moreover, almost half of the demand nodes – 13 to be precise – are assigned to a different POD in the second case. These observations confirm our anticipation that the optimal solution is pretty sensitive to the value of a. The final column of Table 2.10 indicates that the second solution achieves a higher reliability at a significant expense of accessibility – 23 out of the 30 nodes have worse expected accessibility scores in the second solution.

	Case 1 ($a = 10,000$)				**Case 2** ($a = 10,000,000$)				**Differences**	
Demand Node k	**Assigned POD** j	$\bar{\nu}^{\dagger}_{kj}$	r_{kj}	$\bar{\nu}_{kj} r_{kj}$	**Assigned POD** j	$\bar{\nu}^{\dagger}_{kj}$	r_{kj}	$\bar{\nu}_{kj} r_{kj}$	$\bar{\nu}_{kj}$	$\bar{\nu}_{kj} r_{kj}$
1	1	0	172,484	0	1	0	166,774	0	0	0
2	2	0	10,292	0	2	0	11,171	0	0	0
3	3	0	3,462	0	3	0	3,856	0	0	0
4	2	9.5	583	5,517	1	14.1	646	9,134	-4.7	-3,617
5	1	31.2	1,667	52,059	1	31.2	1,816	56,712	0	-4,653
6	1	6.4	576	3,672	1	6.4	641	4,086	0	-414
7	1	27.5	2,712	74,519	1	27.5	2,894	79,520	0	-5,001
8	3	3.8	992	3,734	2	18.3	1,128	20,687	-14.6	-16,953
9	1	21	808	16,973	1	21	908	19,074	0	-2,101
10	2	9.2	1,674	15,339	2	9.2	1,843	16,887	0	-1,548
11	12	20.1	1,980	39,771	11	0	2,111	-	20.1	39,771
12	12	0	2,211	0	3	31.9	2,523	80,530	-31.9	-80,530
13	2	12	1,432	17,198	2	12	1,592	19,119	0	-1,921
14	2	7.5	519	3,892	1	13.7	576	7,864	-6.2	-3,972
15	1	26	1,944	50,591	1	26	2,223	57,851	0	-7,260
16	16	0	2,869	0	16	0	3,438	0	0	0
17	16	17.5	1,660	29,100	16	17.5	1,866	32,711	0	-3,611
18	3	4.3	571	2,431	3	4.3	625	2,660	0	-229
19	1	14.9	1,493	22,211	1	14.9	1,612	23,981	0	-1,770
20	3	6.4	1,057	6,712	2	19.6	1,204	23,568	-13.2	-16,856
21	2	6.4	2,243	14,410	2	6.4	2,404	15,445	0	-1,035
22	22	0	2,820	0	1	44.6	3,096	137,951	-44.6	-137,951
23	2	7.3	1,216	8,823	1	11.8	1,361	16,009	-4.5	-7,186
24	2	2.7	487	1,310	1	6.2	518	3,196	-3.5	-1,886
25	1	10.1	1,352	13,618	1	10.1	1,453	14,636	0	-1,018
26	22	13.9	2,288	31,840	26	0	2,573	0	13.9	31,840
27	2	17.7	947	16,729	2	17.7	1,094	19,326	0	-2,597
28	1	34.2	1,451	49,705	11	17.1	1,728	29,576	17.1	20,129
29	2	9.7	591	5,717	1	13.9	652	8,690	-4.2	-2,973
30	3	8.3	740	6,163	1	27.5	827	22,780	-19.2	-16,617

†: $\bar{\nu}_{kj} = \sum_{i \in S} p^i \nu^i_{kj}$ is the expected accessibility from node k to its assigned POD j.

Table 2.10: Optimal solutions of the illustrative example.

Chapter 3

Chance-Constrained Two-Stage Stochastic Programming Models

In this chapter, we consider a stochastic pre-disaster relief network design problem, which mainly determines the response facility location and stock pre-positioning decisions in order to improve the effectiveness of the post-disaster relief operations. We develop a risk-averse two-stage stochastic optimization model under random demand and transportation network conditions. It features a mean-risk objective based on the risk measure CVaR, and enforces a joint probabilistic constraint on the feasibility of the second-stage problem concerned with distributing the relief supplies to the affected areas in case of a disaster. To solve this computationally challenging stochastic optimization model, we employ an exact Benders decomposition-based branch-and-cut algorithm. We conduct an extensive computational study, which is based on a case study concerning the threat of hurricanes in the Southeastern part of the United States, to illustrate the computational performance of the proposed solution methods. For a more detailed introduction we refer the reader to Section 1.2.

The rest of the chapter is organized as follows. In Section 3.1, we provide a literature review. In Section 3.2 we introduce a relief network design problem and present a new risk-averse optimization model along with the corresponding mathematical programming formulations. Section 3.3 is dedicated to the Benders decomposition-based solution method. This is followed in Section 3.4 by the computational study.

3.1 Literature Review

In this section, we intend to provide additional discussions about the most relevant studies on pre-disaster network design problems and stochastic programming methods, instead of presenting a detailed literature review.

Pre-disaster network design in humanitarian relief. There is a rich literature on pre-

disaster relief network design in the presence of uncertainty. In Introduction, we discuss the most relevant studies to highlight our contributions. In addition, we would like to mention another related study by Rawls and Turnquist (2011). The authors extend their earlier model (Rawls and Turnquist, 2010) by introducing a lower bound on the probability that the demand at each node is satisfied and the resulting total delivery amount-weighted shipment distance associated with each commodity is less than a specified threshold. They consider the same risk-neutral objective function (the expected total cost of opening facilities, demand shortages, and purchasing, shipping and holding the relief supplies), but no solution method is developed. As mentioned in Introduction, Hong et al. (2015) also consider a chance-constrained variant of the model presented in Rawls and Turnquist (2010). The authors enforce a joint chance constraint to ensure the existence of a feasible flow of relief supplies to satisfy the demand across the network with a specified high probability. They focus on developing computationally effective mixed-integer linear programming (LP) reformulations of the proposed model to able to solve instances with a large number of scenarios. Their methods rely on using the Gale-Hoffman inequalities to characterize the conditions on the probabilistic existence of a feasible network flow for the second-stage problem. Thus, their developments depend on the particular network flow formulation of the second-stage problem. On the other hand, the modeling approach used in our study is applicable to any LP formulation of the second-stage problem. For other types of pre-disaster relief network design problems and optimization models we refer the reader to the following review papers: Caunhye et al. (2012), Çelik et al. (2012), and Liberatore et al. (2013a).

Risk-averse two-stage stochastic programming. Risk-averse two-stage optimization models have been receiving increasing interest in the recent stochastic programming literature. One stream of research focuses on incorporating risk measures into the objective function (see, e.g., Ahmed, 2006; Schultz and Tiedemann, 2006; Fábián, 2008; Miller and Ruszczyński, 2011; Noyan, 2012), while another stream focuses on enforcing a chance constraint (see, e.g., Luedtke, 2014; Hong et al., 2015; Liu et al., 2016; Prékopa and Unuvar, 2015). We next elaborate on such relevant risk-averse approaches.

Noyan (2012) studies a class of two-stage mean-risk stochastic programming model, where CVaR is specified as the risk measure on the random total cost function (including the recourse cost). Luedtke (2014) considers a deterministic cost function by focusing only on the feasibility of the second-stage problem and not capturing the second-stage (recourse) cost function. In particular, the author introduces a class of chance-constrained two-stage stochastic programs which guarantee the feasibility of the second-stage problem with a specified high probability, and develop a Benders decomposition-based branch-and-cut algorithm relying on a new class of feasibility cuts. Enforcing a joint chance constraint on the feasibility of the second-stage problem is a natural alternative to the conservative traditional approach of guaranteeing the feasibility under each

scenario. The idea of developing such a stochastic programming model with a joint probabilistic constraint on the feasibility of the second-stage problem is not new - it has been first proposed by Prékopa (1980) for a class of network flow type problems (without a solution method) - but adapting such a modeling approach for different types of problems and developing alternate effective solution methods (Luedtke, 2014; Hong et al., 2015; Prékopa and Unuvar, 2015) are recent. On the other hand, chance-constrained single-stage optimization models are widely used in many engineering applications in a variety of fields to restrict the probability of certain undesirable events; the interested reader is referred to Prékopa (1995), Prékopa (2003), Dentcheva (2006), Shapiro et al. (2009), and the references therein.

Liu et al. (2016) extend the modeling approach of Luedtke (2014) by considering the cost of the recovery actions taken when the second-stage problem under a scenario is infeasible, and propose a class of risk-neutral chance-constrained two-stage stochastic programming models. They develop a set of optimality cuts to ensure the exact calculation of the optimal second-stage objective function values in the Benders decomposition-based branch-and-cut framework. On the other hand, Bülbül et al. (2016) extend the study of Liu et al. (2016) and Noyan (2012) by incorporating CVaR as a risk measure into the objective function and enforcing a joint chance constraint on the feasibility of the second-stage problem, respectively. They adopt this approach to a single-machine scheduling problem under random processing times. In this variant involving CVaR, Benders decomposition-based algorithm also requires approximating the CVaR measure for the exact calculation of the optimal objective function values (as in, e.g., Noyan, 2012).

This study contributes to the literature by introducing a new pre-disaster relief network design problem under the uncertain aspects of the post-disaster environment. It develops a new risk-averse two-stage stochastic model, which takes into account both the quantitative and qualitative aspects of risk. To the best of our knowledge, adopting such a modeling approach and devising a decomposition-based branch-and-cut algorithm for such a chance-constrained model is a first in the humanitarian logistics literature.

3.2 A Stochastic Optimization Model for Designing Pre-Disaster Relief Networks

In this section, we describe the relief network design problem of interest and present our risk-averse modeling approach along with the corresponding mathematical programming formulations.

We consider a location-allocation problem for pre-disaster planning, which determines the locations and capacities of the response facilities and their inventory levels of the relief supplies well before a disaster strikes. There is usually a high level of uncer-

tainty while making such decisions in the absence of accurate predictions; for instance, there is uncertainty in the exact location and intensity of an earthquake, in the damage level of roads, and in the size of the population affected. We capture the inherent uncertainty through a finite set of scenarios representing the potential future outcomes; this scenario-based approach allows us to account for the dependency between uncertain parameters. As well recognized in the recent literature, the two-stage modeling framework is a good fit for the pre-disaster network design problem under uncertain parameters. In our two-stage model, the decisions on the locations, sizes, and inventory levels of the response facilities are made at the first stage before uncertainty is resolved. The decisions concerned with distributing the relief supplies under each scenario are made at the second stage given the predetermined first-stage decision variables. We consider a single type of relief supply; relief organizations usually bundle the critical relief supplies such as prepackaged food, medical kits, and water.

We define the total cost associated with a location-allocation policy as the total cost of opening facilities, demand shortages, and purchasing and shipping the relief supplies, and aim to minimize it while satisfying the demand across the network. The total cost and demand values are random due to the inherent uncertainty, and there are alternative modeling approaches to deal with the randomness in the objective function and in the constraints. In particular, we consider a mean-risk objective function, where CVaR is specified as the risk measure to model decision makers' risk preferences regarding the random total cost. This modeling approach provides flexibility to capture a wide range of risk preferences, including risk-neutral and worst-case approaches. We avoid the overly conservative approach of satisfying the demand at each node under each scenario by relaxing the demand satisfaction constraints in the second-stage. We keep the violation of the relaxed constraints under control by enforcing a joint chance constraint on the feasibility of the second-stage problem (a qualitative mean) and penalizing the violations in the objective function (a quantitative mean).

We can say that aiming to minimize the distance-based shipping costs help to improve the accessibility, which can be defined as the ease of access to the relief supplies. In addition, we intend to address equity in accessibility by defining coverage sets including the candidate facilities that satisfy a common threshold requirement in terms of the accessibility scores associated with the links of the network. In this way, the last mile relief network is designed in such a way that each demand location can be served only by a facility that satisfies the accessibility requirement. For example, Noyan et al. (2016) follow a similar approach; however, they consider a more elaborate characterization of accessibility, where the accessibility is affected both by the physical and demographical/socio-economical factors (e.g., age, gender); in particular, accessibility scores are defined as weighted travel times, where the weights are based on a mobility score reflecting the proportions of socially-disadvantaged populations (disabled people, elderly, and women with

children). However, they consider a post-disaster setup which focuses on serving a particular area (e.g., a city affected by an earthquake), where people in the affected regions mostly travel (or walk) to the distribution facilities to receive relief supplies and services. On the other hand, in our pre-disaster setup, we consider a wider range network, where the distance from a demand node to the assigned facility may take several hours. Therefore, we assume that the relief supplies will be delivered to the demand nodes and ignore the mobility scores related to the socially-disadvantaged populations; in particular, we define the accessibility scores as the travel times (which reflect the physical factors).

3.2.1 Mathematical Model and Formulations

We consider a network where each node represents a geographical area, and denote the set of demand nodes by I and the set of candidate facilities by J. We assume that $J \subseteq I$ without loss of generality; if there exist $j \in J$ such that $j \notin I$, we can simply set the demand of facility j as zero and modify the corresponding flow conservation constraints accordingly (more specifically, avoid inflow from other nodes). We consider a finite probability space $(\Omega, 2^{\Omega}, \mathbb{P})$ with $\Omega = \{\omega_1, \ldots, \omega_n\}$ and $\mathbb{P}(\omega_s) = p^s$, $s \in S$. Thus, $S := \{1, \ldots, n\}$ denotes the set of scenarios; each scenario represents the joint realization of the random input parameters (including node demands, the accessibility scores associated with the links, the damage levels of supplies and cost parameters). A facility can cover (serve) a demand node only if the accessibility score to the facility from the demand location under any scenario does not exceed a maximum threshold τ. Next, we provide the list of the additional input parameters:

Scenario-independent input parameters

L: size categories of a facility, indexed by l,

K_l: the overall capacity of a facility of size category $l, l \in L$,

F_{jl}: fixed cost of opening a facility of size category l at location j, $j \in J,\ l \in L$,

a_j: unit acquisition (and storage) cost for facility $j,\ j \in J$.

Scenario-dependent input parameters

d_i^s: demand at location i under scenario s, $i \in I,\ s \in S$,

ν_{ji}^s: score for accessibility to demand node i from candidate facility j from under scenario s, $i \in I,\ j \in J,\ s \in S$; smaller values are preferred,

$N_i^s = \{j \in J \mid \nu_{ji}^s \leq \tau\}$: set of candidate facilities that can cover demand node i under scenario s, $i \in I,\ s \in S$,

$M_j^s = \{i \in I \mid \nu_{ji}^s \leq \tau\}$: set of demand nodes that can be covered by the candidate facility j under scenario s, $j \in J,\ s \in S$,

γ_j^s: proportion of supplies at facility j that remains usable under scenario s ($0 \leq \gamma_j^s \leq 1$), $j \in J,\ s \in S$,

μ^s: unit penalty cost for the demand shortage under scenario s, $s \in S$,

c_{ji}^s: cost of shipping a unit of supply from facility j to demand node i under scenario $s,\ s \in S$.

First-stage decision variables

$y_{jl} = 1$ if a facility of size category $l \in L$ is located at node $j \in J$, and $y_{jl} = 0$ otherwise,

R_j: amount of supplies pre-located at facility $j \in J$.

Second-stage decision variables

x_{ji}^s: amount of supplies delivered to demand node $i \in I$ from facility j under scenario s, $s \in S,\ j \in N_i^s$.

ϱ_i^s: amount of unsatisfied demand (demand shortage) at node $i \in I$ under scenario $s,\ s \in S$.

Given the locations of the facilities and the pre-stocked inventory levels, we are concerned with distributing the relief supplies in the second-stage. We consider two versions of the second-stage problem: i) "*ideal*" second-stage problem (ISSP) requires satisfying the demand at each node; ii) "*non-ideal*" second-stage problem (NISSP) allows demand shortages at the expense of incurring additional penalties in the objective function. We denote these two states of the second-stage problem under elementary event ω by $\beta(\omega) = 0$ and $\beta(\omega) = 1$, respectively. Given the first-stage decision vector $(\mathbf{y}, \mathbf{R})$, the random input data vector $\boldsymbol{\xi}(\omega)$, and the state of the second-stage problem, $Q((\mathbf{y}, \mathbf{R}), \boldsymbol{\xi}(\omega), \beta(\omega))$ designates the random optimal second-stage objective function value. Denoting the realization of the random data under scenario $s \in S$ by $\boldsymbol{\xi}^s = (\mathbf{d}^s, \boldsymbol{\nu}^s, \boldsymbol{\gamma}^s, \mathbf{c}^s, \mu^s)$ we next present the formulations of the ideal and non-ideal second-stage problems.

$$(\mathbf{ISSP})$$

$$Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}^s, \beta^s = 0) =$$

$$\text{minimize} \quad \sum_{j \in J} \sum_{i \in M_j^s \setminus \{j\}} c_{ji}^s x_{ji}^s \tag{3.1a}$$

$$\text{subject to} \quad \sum_{j \in N_i^s} x_{ji}^s \geq d_i^s, \qquad i \in I \setminus J, \tag{3.1b}$$

$$R_j \gamma_j^s + \sum_{i \in N_j^s \setminus \{j\}} x_{ij}^s - \sum_{i \in M_j^s \setminus \{j\}} x_{ji}^s \geq d_j^s, \qquad j \in J, \tag{3.1c}$$

$$\sum_{i \in M_j^s \setminus \{j\}} x_{ji}^s \leq R_j \gamma_j^s, \qquad j \in J, \tag{3.1d}$$

$$\sum_{i \in N_j^s \setminus \{j\}} x_{ij}^s \leq (1 - \sum_{l \in L} y_{jl}) d_j^s, \qquad j \in J, \tag{3.1e}$$

$$x_{ji}^s \geq 0, \qquad j \in J,\ i \in M_j^s \setminus \{j\}. \tag{3.1f}$$

In this formulation, according to constraints (3.1b) - (3.1d), the demands at node $i \in I \setminus J$ and node $j \in J$ such that $R_j = 0$ (and consequently, $\sum_{i \in M_j^s \setminus \{j\}} x_{ji}^s = 0$) are satisfied by delivering a sufficient amount of relief supplies from the open facilities that can cover node i and node j, respectively. Constraints (3.1c) ensure the demand satisfaction at node $j \in J$ with $R_j > 0$ taking into consideration the amount of undamaged supplies that are prelocated at node j ($R_j \gamma_j^s$). Constraints (3.1d) play a dual role by imposing that the amount delivered from each open facility does not exceed the level of undamaged supplies that are prelocated and avoiding outgoing flow if there is no facility located. Constraints (3.1e) also play a dual role by mandating that there is no delivery to a node from other facilities if there is an open facility at that node and imposing a maximum limit on the amount delivered to the node, otherwise.

It is highly desirable to ensure that the demand constraints (3.1b)-(3.1c) are satisfied under each scenario. However, this would lead to overly conservative solutions. As a remedy, we follow suit with Liu et al. (2016) and enforce the feasibility of the ideal second-stage problem (3.1) with a specified high probability. In addition, to account for the undesirable outcomes in the objective under an infeasible scenario $s \in S$, we consider the following non-ideal second-stage problem, which relaxes the demand satisfaction constraints and penalizes the demand shortages.

$$(\textbf{NISSP})$$

$$Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}^s, \beta^s = 1) =$$

$$\text{minimize} \quad \sum_{j \in J} \sum_{i \in M_j^s \setminus \{j\}} c_{ji}^s x_{ji}^s + \sum_{i \in I} \mu^s \varrho_i^s \tag{3.2a}$$

$$\text{subject to} \quad \varrho_i^s \geq d_i^s - \sum_{j \in N_i^s} x_{ji}^s, \qquad i \in I \setminus J, \tag{3.2b}$$

$$\varrho_j^s \geq d_j^s + \sum_{i \in M_j^s \setminus \{j\}} x_{ji}^s - (R_j \gamma_j^s + \sum_{i \in N_j^s \setminus \{j\}} x_{ij}^s),\ j \in J, \tag{3.2c}$$

$$(3.1\text{d}) - (3.1\text{f}), \tag{3.2d}$$

$$\varrho_i^s \geq 0, \qquad i \in I. \tag{3.2e}$$

For any first-stage decision vector $(\mathbf{y}, \mathbf{R})$, $Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}(\omega), \beta^s = 1)$ is finite for all $s \in S$. Constraints (3.2b) and (3.2c), together with the objective of minimizing the second term in (3.2a), ensure the exact calculation of the demand shortages. Observe that $(\mathbf{NISSP})$ captures $(\mathbf{ISSP})$ as a special case; the ideal version enforces $\varrho_i^s = 0$ for all $i \in I$, while the non-ideal version relaxes this condition by allowing ϱ_i^s to take positive values.

We next provide a discussion to contrast our approach of formulating the second-stage problem with that of the relevant existing studies (Rawls and Turnquist, 2010; Noyan, 2012; Hong et al., 2015). The main difference relies in expressing the flow conservation. Denoting the set of links in the network by A and ignoring the surplus amounts, the following network flow conservation constraints, under the assumption that $I = J$, are used in Rawls and Turnquist (2010):

$$\varrho_j^s \geq d_j^s + \sum_{(j,i)\in A} x_{ji}^s - (R_j\gamma_j^s + \sum_{(i,j)\in A} x_{ij}^s) \;\; j \in J. \tag{3.3}$$

In addition to demand satisfaction constraints (3.3), the formulation of their second-stage problem only includes the constraints concerned with the arc capacity limits. Thus, the existing studies formulate the second-stage problem as a classical network flow model, which does not capture assignment-based coverage issues. On the other hand, we aim to guarantee a more structured flow by focusing on assigning the demand points to the facilities. This assignment-based formulation follows from a restricted version of (3.3) and seems to be aligned with the practices in relief operations. Each demand can be served by facilities that are sufficiently accessbile (for this purpose, coverage sets are used). A facility located at a node serves its own location (implying that there is no flow among facilities); while a facility can be assigned to serve people at other nodes, assigning people at its own location to other facilities may not be acceptable. Moreover, it may be reasonable to exclude the operational level decisions (detailed flow decisions) considering the arc capacities in the context of long-term pre-disaster planning. More specifically, the assignment-based model $(\mathbf{NISSP})$ can be considered as a restricted (more structured according to the issues in practice) variant of the network flow model in Rawls and Turnquist (2010) featuring (3.3), as described below:

- For demand node $j \in I \setminus J$: there is no pre-stocked inventory and outgoing flow is not allowed ($R_j = 0$ and $\sum_{(j,i)\in A} x_{ji}^s = 0$, yielding (3.2b)).
- For a candidate facility node $j \in J$ there are two cases: it can be both a demand and a facility node or it can be just a demand node. Therefore, for $j \in J$, we use the general form of (3.2c). If there is no facility located at node j (implying $\sum_{l\in L} y_{jl} = 0$), node j turns out to be only a demand node and (3.1d) ensures that (3.2c) takes the

form of (3.2b). If a facility is located at node $j \in J$, the ingoing flow from other nodes is not allowed by (3.1e), and (3.2c) becomes $\varrho_j^s \geq d_j^s + \sum_{i \in M_j^s \setminus \{j\}} x_{ji}^s - R_j \gamma_j^s$.

Remark 2 *Alternatively, one may penalize the proportion of unsatisfied demand by replacing ϱ_i^s with $\frac{\varrho_i^s}{\xi_i^s}$, $i \in I,\ s \in S$ in* (3.2a)*; this would serve the purpose of providing a better service in terms of equity in supply allocation (see, e.g., Noyan et al., 2016).*

Following the discussions above, the proposed chance-constrained two-stage mean-risk stochastic programming model of the pre-disaster relief network design problem of interest $(\mathbf{RA-PNDP})$ is given by

$$
\begin{aligned}
&(\mathbf{RA-PNDP})\\
&\text{minimize} \quad \sum_{j\in J}\sum_{l\in L} F_{jl}y_{jl} + \sum_{j\in J} a_j R_j \\
&\qquad + (1-\lambda)E\left(Q(\mathbf{y},\mathbf{R},\boldsymbol{\xi}(\omega),\beta(\omega))\right) && \text{(3.4a)}\\
&\qquad + \lambda\,\mathrm{CVaR}_\alpha\left(Q(\mathbf{y},\mathbf{R},\boldsymbol{\xi}(\omega),\beta(\omega))\right)\\
&\text{subject to} \quad \sum_{l\in L} y_{jl} \leq 1, \qquad j \in J, && \text{(3.4b)}\\
&\qquad R_j \leq \sum_{l\in L} K_l y_{jl}, \qquad j \in J, && \text{(3.4c)}\\
&\qquad y_{jl} \in \{0,1\}, \qquad j \in J,\ l \in L, && \text{(3.4d)}\\
&\qquad R_j \geq 0, \qquad j \in J, && \text{(3.4e)}\\
&\qquad \mathbb{P}\left\{(\mathbf{y},\mathbf{R}) \in P(\omega)\right\} \geq 1-\epsilon, && \text{(3.4f)}\\
&\qquad \beta(\omega) = 0 \implies (\mathbf{y},\mathbf{R}) \in P(\omega), \qquad \omega \in \Omega, && \text{(3.4g)}\\
&\qquad \beta(\omega) \in \{0,1\},\ \omega \in \Omega. && \text{(3.4h)}
\end{aligned}
$$

In this mean-risk approach, λ is a non-negative trade-off coefficient representing the exchange rate of mean cost for risk. The set $P(\omega)$, featured in (3.4f) and (3.4g), is defined as $P(\omega) = \{(\mathbf{y},\mathbf{R}) \in \mathbb{R}_+^{|J|\times|I|} \times \mathbb{R}_+^{|J|} : \exists\, \mathbf{x} \text{ satisfying (3.1b)} - \text{(3.1f)}\}$ for $\omega \in \Omega$; thus, $P(\omega^s)$ denotes the set of decisions for which the ideal second-stage problem under the scenario $s \in S$ is feasible. The objective function (3.4a) minimizes the mean-risk function of the total cost, where the resource cost function $Q(\mathbf{y},\mathbf{R},\boldsymbol{\xi}(\omega),\beta(\omega))$ is calculated by solving $(\mathbf{ISSP})$ or $(\mathbf{NISSP})$ depending on the state of the second-stage problem. Constraint (3.4b) ensures that at most one facility is located at any node. Constraints (3.4c) play a dual role by imposing a maximum capacity limit on the inventory level at each open facility and enforcing that there is a facility located at node j if its inventory level is positive. According to the constraints (3.4f) and (3.4g) (together with (3.4b)-(3.4e)), the model provides first-stage decisions for which the second-stage problem is guaranteed to be feasible with a specified high probability of $1-\epsilon$, i.e., $\boldsymbol{\varrho}^s = \mathbf{0}$ for $s \in \bar{S}$ such

that $\sum_{s\in\bar{S}} p^s \geq 1-\epsilon$. For the remaining scenarios, the positive violation amounts are penalized in the objective function. Thus, the model features quantitative and qualitative measures of risk in the objective (3.4a) and constraints (3.4f), respectively. The rest of the constraints are for the non-negativity and binary restrictions.

Risk measure - CVaR. In this subsection, we briefly discuss the concept of CVaR and present some LP representations of CVaR under the assumptions that all random variables are defined on some finite probability spaces; for further details we refer the reader to Rockafellar (2007).

For a random variable Z, the *conditional value-at-risk* at confidence level $\alpha \in [0,1)$ is related to the expectation of Z under the condition that it exceeds the α-quantile threshold. Its precise definition is given by (Rockafellar and Uryasev, 2000)

$$\mathrm{CVaR}_\alpha(Z) = \min\left\{\eta + \frac{1}{1-\alpha}\mathbb{E}\left([Z-\eta]_+\right) \ : \ \eta \in \mathbb{R}\right\}, \tag{3.5}$$

where $[a]_+ = \max(a,0)$ denotes the positive part of a number $a \in \mathbb{R}$. In fact, the minimum in (3.5) is attained at the α-quantile, which is known as the *value-at-risk* (VaR) at confidence level α: $\mathrm{VaR}_\alpha(Z) = \min\{\eta \in \mathbb{R} \ : \ P(Z \leq \eta) \geq \alpha\}$. For risk-averse decision makers typical choices for the confidence level are large values such as $\alpha = 0.9$.

Suppose that Z is a random variable with realizations $z^1, \ldots, z^n$ and corresponding probabilities $p^1, \ldots, p^n$. Then, the optimization problem in (3.5) can equivalently be formulated as the following linear program:

$$\min\{\eta + \frac{1}{1-\alpha}\sum_{s=1}^{n} p^s v^s \ : \ v^s \geq z^s - \eta \quad \forall\, s \in \{1,\ldots,n\}, \quad \boldsymbol{v} \in \mathbb{R}^n_+,\ \eta \in \mathbb{R}\}. \tag{3.6}$$

CVaR is a widely applied risk measure due to a number of useful properties. It captures a wide range of risk preferences, including risk-neutral (for $\alpha = 0$) and pessimistic worst-case (for sufficiently large values of α) preferences. In addition, it is a coherent risk measure; coherence refers to a set of axiomatic properties (monotonicity, subadditivity, positive homogeneity, and translation equivariance) desirable from the perspective of risk management (Artzner et al., 1999). It is also a spectral risk measure (Acerbi, 2002) and thus can be viewed as a weighted sum of the least favorable outcomes as illustrated by the following LP representation, which is equivalent to the LP dual of (3.6):

$$\max\left\{\frac{1}{1-\alpha}\sum_{s=1}^{n}\gamma^s z^s \ : \ \sum_{s=1}^{n}\gamma^s = 1-\alpha, \quad 0 \leq \gamma^s \leq p^s \quad \forall\, s \in \{1,\ldots,n\}\right\}. \tag{3.7}$$

In our setup, Z denotes the random total cost, i.e., $Z = \sum_{j\in J}\sum_{l\in L} F_{jl}y_{jl} + \sum_{j\in J} a_j R_j + Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}(\omega), \beta(\omega))$. $\mathrm{VaR}_\alpha(Z)$ provides an upper bound on the total cost that is exceeded only with a small probability of $1-\alpha$ and $\mathrm{CVaR}_\alpha(Z)$ is used to measure the severity of

the total cost if it is larger than $\mathrm{VaR}_\alpha(Z)$. Using the translation equivariance property of CVaR, we have the relation

$$\mathrm{CVaR}_\alpha(Z) = \sum_{j\in J}\sum_{l\in L} F_{jl}y_{jl} + \sum_{j\in J} a_j R_j + \mathrm{CVaR}_\alpha\left(Q(\mathbf{y},\mathbf{R},\boldsymbol{\xi}(\omega),\beta(\omega))\right),$$

which yields the mean-risk function of the form (3.8a). While expressing $\mathrm{CVaR}_\alpha(Z)$ in the mathematical programming reformulations of $(\mathbf{RA}-\mathbf{PNDP})$ (see, e.g., $(\mathbf{DEF})$), we use the classical LP representation of CVaR (3.6). On the other hand, the knapsak representation (3.7) provides an intuitive insight to understand the concept of CVaR; in fact, due to this representation CVaR is also known in the literature as *average value-at-risk* and *tail value-at-risk*.

Deterministic equivalent formulation. In this subsection, we present a mixed-integer linear programming formulation of the proposed model $(\mathbf{RA}-\mathbf{PNDP})$; this large-scale formulation is referred to as the deterministic equivalent formulation $(\mathbf{DEF})$. We estimate the expectation and CVaR of $Q(\mathbf{y},\mathbf{R},\boldsymbol{\xi}(\omega),\beta(\omega))$ via sample averaging.

$(\mathbf{DEF})$

$$\begin{aligned}
\text{minimize}\quad & \sum_{j\in J}\sum_{l\in L} F_{jl}y_{jl} + \sum_{j\in J} a_j R_j \\
& + (1-\lambda)\sum_{s\in S} p^s \left(\sum_{j\in J}\sum_{i\in M_j^s\setminus\{j\}} c_{ji}^s x_{ji}^s + \sum_{i\in I}\mu^s \varrho_i^s\right) && \text{(3.8a)}\\
& + \lambda\left(\eta + \frac{1}{1-\alpha}\sum_{s\in S} p^s \upsilon^s\right)
\end{aligned}$$

$$\text{subject to:}\quad \text{(3.4b)} - \text{(3.4e)} \qquad \text{(3.8b)}$$

$$\varrho_i^s \geq d_i^s - \sum_{j\in N_i^s} x_{ji}^s, \quad i\in I\setminus J,\ s\in S, \qquad \text{(3.8c)}$$

$$\varrho_j^s \geq d_j^s + \sum_{i\in M_j^s\setminus\{j\}} x_{ji}^s - (R_j\gamma_j^s + \sum_{i\in N_j^s\setminus\{j\}} x_{ij}^s), \quad j\in J,\ s\in S, \qquad \text{(3.8d)}$$

$$\sum_{i\in M_j^s\setminus\{j\}} x_{ji}^s \leq R_j\gamma_j^s, \quad j\in J,\ s\in S, \qquad \text{(3.8e)}$$

$$\sum_{i\in N_j^s\setminus\{j\}} x_{ij}^s \leq (1-\sum_{l\in L} y_{jl})d_j^s, \quad j\in J,\ s\in S, \qquad \text{(3.8f)}$$

$$\varrho_i^s \leq M_i^s\beta_s, \quad i\in I,\ s\in S, \qquad \text{(3.8g)}$$

$$\sum_{s\in S} p_s\beta_s \leq \epsilon, \qquad \text{(3.8h)}$$

$$\upsilon^s \geq \sum_{j\in J}\sum_{i\in M_j^s\setminus\{j\}} c_{ji}^s x_{ji}^s + \sum_{i\in I}\mu^s\varrho_i^s - \eta, \quad s\in S, \qquad \text{(3.8i)}$$

$$x_{ji}^s \geq 0, \quad j\in J,\ s\in S,\ i\in M_j^s\setminus\{j\}, \qquad \text{(3.8j)}$$

$$\varrho_i^s \geq 0, \quad i \in I,\ s \in S, \tag{3.8k}$$

$$\boldsymbol{\beta} \in \{0,1\}^n, \quad \boldsymbol{v} \in \mathbb{R}_+^n, \quad \eta \in \mathbb{R}_+. \tag{3.8l}$$

Here we only elaborate on the new type of constraints, namely (3.8g), (3.8h), and (3.8i). The constraints (3.8g)-(3.8h), where M_i^s is a big-M parameter (sufficiently large number), mandate that $\mathbb{P}\{(\mathbf{y}, \mathbf{R}) \in P(\omega)\} \geq 1 - \epsilon$; observe that $\beta_s = 0$ implies that $\varrho_i^s = 0$, and consequently $(\mathbf{ISSP})$ is feasible. In this formulation, the big-M parameter M_i^s can be set to the largest possible shortage value d_i^s at node $i \in I$ for scenario $s \in S$. Constraints (3.8i) ensure the exact calculation of the CVaR term in the objective function (3.4); see the linear programming representation of CVaR in (3.6).

It is important to highlight the generality of our hybrid model, which captures some well-known approaches as special cases: i) for $\lambda = 0$ and $\epsilon = 0$ we obtain the traditional risk-neutral counterpart where the feasibility of the second-stage problem is enforced under each scenario; ii) for $\lambda = 0$ and $\epsilon = 1$ we obtain the risk-neutral approach where the demand shortages are penalized as in Rawls and Turnquist (2010); iii) for $\epsilon = 1$ we obtain the risk-averse counterpart of the approach in (ii), which minimizes the mean-CVaR function as in Noyan (2012); iv) for $\lambda = 0$ we obtain the chance-constrained risk-neutral approach as in Liu et al. (2016), and v) setting $\lambda = 0$ and omitting the second-stage objective function term yields a chance-constrained approach as in Luedtke (2014) and Hong et al. (2015).

Finally, we note that one can consider other risk measures as an alternative to CVaR; for instance, Ahmed (2006) investigates several risk measures for which the mean-risk function preserves the convexity, and thus, leads to computationally tractable solution methods. In fact, for such convexity preserving risk measures (including absolute semideviation and quantile deviation), slight variants of our Benders decomposition-based algorithm can be applied.

3.3 Benders Decomposition-Based Branch-and-Cut Algorithm

In this section, we describe the proposed Benders decomposition-based branch-and-cut algorithm for solving $(\mathbf{RA-PNDP})$. This algorithm utilizes the strong valid inequalities proposed by Luedtke (2014) for the feasibility of the $(\mathbf{ISSP})$ as well as the strong optimality cuts proposed by Liu et al. (2016) for the additional cost incurred by the recourse decisions. In addition, following from Bülbül et al. (2016), it features classical LP representation of CVaR for the exact calculation of the mean-risk objective function.

We first describe how we obtain the relaxed master problem. The difficulty of $(\mathbf{RA-PNDP})$ mainly stems from the two features of the mathematical model; namely,

the joint chance constraint enforced for the feasibility of the second-stage problem and the exact calculation of the recourse function value. In our solution approach, we introduce auxiliary decision variables $\theta^s, s \in S$, to approximate the recourse function values and relax the constraints (3.4f) and (3.4g). We define the following sets to describe our algorithm.

Let $\mathcal{X}$ be the set of first-stage decisions $(\mathbf{y}, \mathbf{R})$ given by

$$\mathcal{X} = \left\{ (\mathbf{y}, \mathbf{R}) \in \mathbb{R}_+^{|J|\times|I|} \times \mathbb{R}_+^{|J|} \ : \ \text{(3.4b)} - \text{(3.4d)} \right\}. \tag{3.9}$$

In addition, the following two sets, $\mathcal{F}$ and $\mathcal{O}$ are approximated respectively by their polyhedral counterparts $\mathcal{C}$ and $\mathcal{D}$ within the branch-and-cut framework:

$$\mathcal{F} = \Bigg\{ (\mathbf{y}, \mathbf{R}, \boldsymbol{\beta}) \in \mathcal{X} \times \{0,1\}^n \ : \ \sum_{s\in S} \beta^s p^s \le \epsilon, \quad \beta^s = 0 \implies (\mathbf{y}, \mathbf{R}) \in P(\omega^s), \quad \forall s \in S \Bigg\}, \tag{3.10}$$

$$\mathcal{O} = \Bigg\{ (\mathbf{y}, \mathbf{R}, \boldsymbol{\beta}, \boldsymbol{\theta}) \in \mathcal{F} \times \mathbb{R}_+^n \ : \ \theta^s \ge Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}^s, \beta^s = 0)\,(1-\beta^s) + Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}^s, \beta^s = 1)\beta^s, \quad \forall s \in S \Bigg\}. \tag{3.11}$$

Using above notations, we define our relaxed master problem $(\mathbf{RMP})$ as follows:

$(\mathbf{RMP})$

$$\begin{aligned}
\text{minimize} \quad & \sum_{j\in J}\sum_{l\in L} F_{jl} y_{jl} + \sum_{j\in J} c_j^a R_j && \text{(3.12a)}\\
& + (1-\lambda)\sum_{s\in S} p^s \theta^s + \lambda\left(\eta + \frac{1}{1-\alpha}\sum p^s v^s\right) \\
\text{subject to:} \quad & (\mathbf{y}, \mathbf{R}) \in \mathcal{X}, && \text{(3.12b)}\\
& (\mathbf{y}, \mathbf{R}, \boldsymbol{\beta}) \in \mathcal{C}, && \text{(3.12c)}\\
& (\mathbf{y}, \mathbf{R}, \boldsymbol{\beta}, \boldsymbol{\theta}) \in \mathcal{D}, && \text{(3.12d)}\\
& v^s \ge \theta^s - \eta, \quad s \in S, && \text{(3.12e)}\\
& \text{(3.8h), (3.8l).} && \text{(3.12f)}
\end{aligned}$$

Here, the set $\mathcal{C}$ is a polyhedral outer approximation of the set $\mathcal{F}$, and the set $\mathcal{D}$ is a polyhedral outer approximation of the set $\mathcal{O}$. The sets $\mathcal{C}$ and $\mathcal{D}$ are iteratively defined by the feasibility and optimality cuts within our algorithm. Constraint set (3.12e) is a modified version of (3.8i) for the exact calculation of CVaR.

In classical applications of Benders decomposition-based algorithms, the $(\mathbf{RMP})$ is solved to optimality in every iteration and missing cuts are added to the $(\mathbf{RMP})$ before it is re-optimized. However, this framework might be inefficient since the $(\mathbf{RMP})$ is solved from scratch at every iteration, and potentially it gets harder to solve due to the increasing number of optimality and feasibility cuts. Thus, we employ the state-of-the-art way of embedding the generation of the optimality and feasibility cuts within a branch-and-bound algorithm. In particular, we check for missing optimality and feasibility cuts whenever an integer feasible solution is found. As a consequence, in our implementation a single search tree is generated and the $(\mathbf{RMP})$ is solved only once.

We next describe how to generate the optimality and feasibility cuts for a given first-stage decision $(\hat{\mathbf{y}}, \hat{\mathbf{R}}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}})$. The feasibility and optimality cuts ensure that the constraints (3.4f) and (3.4g) are satisfied and the optimal objective function values of the second-stage problems ($(\mathbf{ISSP})$ and $(\mathbf{NISSP})$) are calculated correctly.

We use the following general representations of the ideal and non-ideal second-stage problems to describe our algorithm.

$$Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}^s, \beta^s = 0) \;=\; \underset{\mathbf{u}}{\text{minimize}} \Big\{ (\mathbf{q}^s)^\top \mathbf{u} \;:\; W^s \mathbf{u} \geq \mathbf{b}^s - T^s \mathbf{z},\; \mathbf{u} \geq 0 \Big\}, \tag{3.13}$$

$$Q(\mathbf{y}, \mathbf{R}, \boldsymbol{\xi}^s, \beta^s = 1) \;=\; \underset{\bar{\mathbf{u}}}{\text{minimize}} \Big\{ (\bar{\mathbf{q}}^s)^\top \bar{\mathbf{u}} \;:\; \bar{W}^s \bar{\mathbf{u}} \geq \bar{\mathbf{b}}^s - \bar{T}^s \mathbf{z}),\; \bar{\mathbf{u}} \geq 0 \Big\}. \tag{3.14}$$

Here, $\mathbf{z} = (\mathbf{y}, \mathbf{R})$ denote the first-stage decision vector. In addition, $\mathbf{u}$ and $\bar{\mathbf{u}}$ respectively denote the ideal second-stage and non-ideal second-stage decision vectors. $T, \bar{T}, W$ and $\bar{W}$ are appropriately sized matrices. Note that for our problem $(\mathbf{RA} - \mathbf{PNDP})$, $\bar{\mathbf{b}}^s = \mathbf{b}^s$ and $\bar{T}^s = T$ for all $s \in S$.

3.3.1 Optimality Cuts

In this section, we describe how we generate the optimality cuts. There are two cases to consider depending on the value of the state variable.

Case 1: Suppose that at an iteration, we have $\hat{\beta}^s = 0$ for some $s \in S$. In this case, we solve the $(\mathbf{ISSP})$ and check whether it is feasible or not. If it is feasible, then it has an optimal solution. The $(\mathbf{ISSP})$ cannot be unbounded since the unboundedness of $(\mathbf{ISSP})$ would imply the unboundedness of the original problem. Let the objective function value of this optimal solution and the optimal dual variable values be denoted by $\hat{f}^s$ and $\boldsymbol{\pi}^s$, respectively. If $\hat{\theta}^s$ is lower than $\hat{f}^s$, then we need to add an optimality cut to the $(\mathbf{RMP})$ since the cost associated with the $(\mathbf{ISSP})$ for scenario s is underestimated. The following inequality is valid for $\mathcal{O}$ and cuts the current solution once it is added to the description of $\mathcal{D}$:

$$\theta^s \geq \boldsymbol{\pi}^s (\mathbf{b}^s - T^s \mathbf{z})(1 - \beta^s). \tag{3.15}$$

Here, note that when β^s is equal to 0, this inequality takes the form of the classical Benders

optimality cut. In addition, when β^s is equal to 1, it becomes redundant, because θ^s is always greater than or equal to 0.

Although we can have a convergent branch-and-cut algorithm by using the optimality cut described above, it is a very weak cut for the case when $\beta^s = 1$, due to the fact that it does not carry any additional information at all. Considering this issue, we make use of a class of stronger optimality cuts introduced by Liu et al. (2016). Let $\boldsymbol{\alpha} = (\boldsymbol{\pi}^s)^\top T^s$. Then the following inequality is valid for $\mathcal{O}$ and it cuts the current solution once it is added to the description of $\mathcal{D}$:

$$\theta^s \geq (\boldsymbol{\pi}^s)^\top (\mathbf{b}^s - T^s \mathbf{z}) - ((\boldsymbol{\pi}^s)^\top \mathbf{b}^s - \bar{v}^s(\boldsymbol{\alpha}))\beta^s. \tag{3.16}$$

In our setting this cut gets the following form:

$$\begin{aligned} \theta^s \geq & \sum_{i \in I \setminus J} d_i^s \pi_i^{(1)} + \sum_{j \in J} d_j^s \left(\pi_j^{(2)} - \pi_j^{(4)}\right) - \sum_{j \in J} R_j \gamma_j^s \left(\pi_j^{(2)} + \pi_j^{(3)}\right) \\ & + \sum_{j \in J} \sum_{l \in L} y_{jl} d_j^s \pi_j^{(4)} - \left(\sum_{i \in I \setminus J} d_i^s \pi_i^{(1)} + \sum_{j \in J} d_j^s \left(\pi_j^{(2)} - \pi_j^{(4)}\right) - \bar{v}^s(\boldsymbol{\alpha}) \right) \beta^s. \end{aligned} \tag{3.17}$$

Here, we solve the following problem to obtain $\bar{v}^s(\boldsymbol{\alpha})$:

$$\bar{v}^s(\boldsymbol{\alpha}) = \text{minimize} \left\{ \boldsymbol{\alpha}\mathbf{z} + (\bar{\mathbf{q}}^s)^\top \bar{\mathbf{u}} : \mathbf{z} \in \bar{\mathcal{X}}, \bar{W}^s \bar{\mathbf{u}} \geq \bar{\mathbf{b}}^s - \bar{T}^s \mathbf{z}, \bar{\mathbf{u}} \geq 0 \right\}. \tag{3.18}$$

Note that any $\bar{\mathcal{X}} \supseteq \mathcal{X}$ is suitable to generate a valid optimality cut. In our implementation, we obtain $\bar{\mathcal{X}}$ by relaxing the integrality restrictions on $\mathcal{X}$ and solve the following linear program to obtain $\bar{v}^s(\alpha)$, where $\pi_d^{(1)}\ d \in I \setminus J$, $\pi_d^{(2)}\ d \in J$, $\pi_d^{(3)}\ d \in J$ and $\pi_d^{(4)}\ d \in J$ are optimal values of the corresponding dual variables of the $(\mathbf{ISSP})$:

$$\begin{aligned} \bar{v}^s(\boldsymbol{\alpha}) = \quad \text{minimize} \quad & \sum_{j \in J} \sum_{i \in M_j^s \setminus \{j\}} c_{ji}^s x_{ji}^s + \sum_{i \in I} cp^s \varrho_i^s \\ & + \sum_{j \in J} R_j \gamma_j^s \left(\pi_j^{(2)} + \gamma_j^s \pi_j^{(3)}\right) && \text{(3.19a)} \\ & - \sum_{j \in J} \sum_{l \in L} y_{jl} d_j^s \pi_j^{(4)} \\ \text{subject to} \quad & \text{(3.4b)} - \text{(3.4c)}, && \text{(3.19b)} \\ & \text{(3.2b)} - \text{(3.2e)}, && \text{(3.19c)} \\ & y_{jl} \in [0,1], \qquad j \in J,\ l \in L, && \text{(3.19d)} \\ & R_j \geq 0, \qquad j \in J. && \text{(3.19e)} \end{aligned}$$

Observe that $\bar{v}^s(\boldsymbol{\alpha})$ is finite, because (3.19a) is bounded from below since all the decision variables and their objective function coefficients are non-negative.

Case 2: We next describe how we generate optimality cuts when $\hat{\beta}^s = 1$ for some $s \in S$. Note that the generation of the optimality cuts in this case is analogous to the previous case. We know that for any given first-stage decision, $(\mathbf{NISSP})$ has a feasible solution. The $(\mathbf{NISSP})$ cannot be unbounded since the unboundedness of the $(\mathbf{NISSP})$ would imply the unboundedness of the original problem. We solve the $(\mathbf{NISSP})$ and obtain its optimal objective function value $\hat{\bar{f}}^s$ and optimal dual variable values $\bar{\boldsymbol{\pi}}^s$. If $\hat{\theta}^s$ is lower than $\hat{\bar{f}}^s$, then we need to add an optimality cut to the $(\mathbf{RMP})$ since the cost associated with the $(\mathbf{NISSP})$ for scenario s is underestimated. The following inequality is valid for $\mathcal{O}$ and cuts the current solution once it is added to the description of $\mathcal{D}$:

$$\theta^s \geq \bar{\boldsymbol{\pi}}^s(\bar{\mathbf{b}}^s - \bar{T}^s\mathbf{z})\beta^s. \tag{3.20}$$

Instead of this cut, we can employ the stronger optimality cuts proposed by Liu et al. (2016). Let $\boldsymbol{\alpha} = (\bar{\boldsymbol{\pi}}^s)^\top \bar{T}^s$. Then, the following inequality is valid for $\mathcal{O}$ and cuts the current solution once it is added to the description of $\mathcal{D}$:

$$\theta^s \geq (\bar{\boldsymbol{\pi}}^s)^\top(\bar{\mathbf{b}}^s - \bar{T}^s\mathbf{z}) - ((\bar{\boldsymbol{\pi}}^s)^\top\bar{\mathbf{b}}^s - \upsilon^s((\bar{\boldsymbol{\pi}}^s)^\top\bar{T}^s))(1-\beta^s). \tag{3.21}$$

In our setting, this cut gets the following form.

$$\begin{aligned}\theta^s \geq & \sum_{i\in I\setminus J} d_i^s\bar{\pi}_i^{(1)} + \sum_{j\in J} d_j^s\left(\bar{\pi}_j^{(2)} - \bar{\pi}_j^{(4)}\right) \\ & - \sum_{j\in J} R_j\gamma_j^z\left(\bar{\pi}_j^{(2)} + \bar{\pi}_j^{(3)}\right) + \sum_{j\in J}\sum_{l\in L} y_{jl}d_j^z\bar{\pi}_j^{(4)} \\ & - \left(\sum_{i\in I\setminus J} d_i^s\bar{\pi}_i^{(1)} + \sum_{j\in J} d_j^s\left(\bar{\pi}_j^{(2)} - \bar{\pi}_j^{(4)}\right) - \upsilon^s(\boldsymbol{\alpha})\right)(1-\beta^s).\end{aligned} \tag{3.22}$$

Here, we solve the following problem to obtain $\upsilon^s(\boldsymbol{\alpha})$:

$$\upsilon^s(\boldsymbol{\alpha}) = \text{minimize}\left\{\boldsymbol{\alpha}\mathbf{z} + (\mathbf{q}^s)^\top\mathbf{u} : \mathbf{z}\in\bar{\mathcal{X}}, W^s\mathbf{u} \geq \mathbf{b}^s - T^s\mathbf{z}, \mathbf{u}\geq 0\right\}. \tag{3.23}$$

Note that any $\bar{\mathcal{X}} \supseteq \mathcal{X}$ is suitable to generate a valid optimality cut. In our implementation, we obtain $\bar{\mathcal{X}}$ by relaxing the integrality restrictions on $\mathcal{X}$ and solve the following linear program to obtain $\upsilon^s(\boldsymbol{\alpha})$, where $\bar{\pi}_d^{(1)}$ $d\in I\setminus J$, $\bar{\pi}_d^{(2)}$ $d\in J$, $\bar{\pi}_d^{(3)}$ $d\in J$ and $\bar{\pi}_d^{(4)}$ $d\in J$ are the optimal values of the corresponding dual variables of the $(\mathbf{NISSP})$:

$$\begin{aligned}\upsilon^s(\boldsymbol{\alpha}) = \quad \text{minimize} \quad & \sum_{j\in J}\sum_{i\in M_j^s\setminus\{j\}} c_{ji}^s x_{ji}^s \\ & + \sum_{j\in J} R_j\gamma_j^z\left(\bar{\pi}_j^{(2)} + \bar{\pi}_j^{(3)}\right)\end{aligned} \tag{3.24a}$$

$$- \sum_{j \in J} \sum_{l \in L} y_{jl} d_j^z \bar{\pi}_j^{(4)}$$

$$\text{subject to} \quad (3.4b) - (3.4c) \tag{3.24b}$$

$$(3.1b) - (3.1f) \tag{3.24c}$$

$$y_{jl} \in [0,1], \qquad j \in J,\ l \in L, \tag{3.24d}$$

$$R_j \geq 0, \qquad j \in J. \tag{3.24e}$$

Observe that $\upsilon^s(\boldsymbol{\alpha})$ is finite, because (3.24a) is bounded from below since all the decision variables and their objective function coefficients are non-negative.

Instead of adding a cut of the form given in (3.22), we could have added the following classical Benders optimality cut to the $(\mathbf{RMP})$:

$$\theta^s \geq \bar{\boldsymbol{\pi}}^s(\bar{\mathbf{b}}^s - \bar{T}^s \mathbf{z}). \tag{3.25}$$

We know that the $(\mathbf{NISSP})$ is a relaxation of the $(\mathbf{ISSP})$, and the objective function value of the $(\mathbf{NISSP})$ is always less than or equal to the objective function of the $(\mathbf{ISSP})$, for any given first-stage solution vector. Therefore, this cut is also valid when β^s is equal to 0 and has the following form in our setting:

$$\begin{aligned} \theta^s \geq \sum_{i \in I \setminus J} d_i^s \bar{\pi}_i^{(1)} + \sum_{j \in J} \left(d_j^s - R_j \gamma_j^s\right) \bar{\pi}_j^{(2)} \\ - \sum_{j \in J} R_j \gamma_j^s \bar{\pi}_j^{(3)} + \sum_{j \in J} \left(\sum_{l \in L} y_{jl} - 1 \right) d_j^s \bar{\pi}_j^{(4)}. \end{aligned} \tag{3.26}$$

The following procedure describes how we generate an optimality cut within our branch-and-cut algorithm. Whenever a missing optimality cut is identified, GenerateOptCut($\mathcal{D},\ \boldsymbol{\pi},\ s,\ \hat{\beta}^s$) is called and a valid cut is added to the description of $\mathcal{D}$.

Algorithm 1: Procedure: GenerateOptCut($\mathcal{D},\ \boldsymbol{\pi},\ s,\ \hat{\beta}^s$)

input : $\mathcal{D}$ - the set of optimality cuts added so far, $\boldsymbol{\pi}$ - optimal dual solution vector, s - current scenario index, $\hat{\beta}^s$ - current value of the state variable.

output: Add an optimality cut to the description of $\mathcal{D}$.

```
if β̂^s = 0 then
    Solve (3.19) and obtain ῡ^s(α);
    Add an optimality cut of the form (3.17) to the description of D;
else
    Add an optimality cut of the form (3.26) to the description of D;
end
```

3.3.2 Feasibility Cuts

In this section, we describe how we generate the feasibility cuts. Suppose that at an iteration, we have $\hat{\beta}^k = 0$ for some $k \in S$. In this case, we solve the $(\mathbf{ISSP})$, and check whether it is feasible or not. If it is infeasible, then the current solution $\hat{\mathbf{z}}$ with $\hat{\beta}^k = 0$ is not feasible for $(\mathbf{RA - PNDP})$ since it violates the constraint (3.4g). Thus, we need to add a feasibility cut to the description of $\mathcal{C}$. To this end, we obtain an extreme ray (denoted by $\boldsymbol{\sigma}^k$) of the dual feasible region of the $(\mathbf{ISSP})$. In addition, let $\boldsymbol{\alpha} = (\boldsymbol{\sigma}^k)^\top T^k$ and $\zeta = (\boldsymbol{\sigma}^k)^\top \mathbf{b}^k$. Then, the following inequality is valid for $\mathcal{F}$ and cuts the current solution once it is added to the description of $\mathcal{C}$.

$$\boldsymbol{\alpha}\mathbf{z} + \zeta\beta^k \geq \zeta. \tag{3.27}$$

For our problem equation (3.27) takes the following form:

$$\sum_{j\in J} R_j\gamma_j^k\left(\sigma_j^{(2)} + \sigma_j^{(3)}\right) - \sum_{j\in J}\sum_{l\in L} y_{jl}d_j^k\sigma_j^{(4)} \geq \left(\sum_{i\in I\setminus J} d_i^k\sigma_i^{(1)} + \sum_{j\in J} d_j^k\left(\sigma_j^{(2)} - \sigma_j^{(4)}\right)\right)(1-\beta^k).$$

Although the inequalities of the form (3.27) are sufficient to obtain a convergent branch-and-cut algorithm, Luedtke (2014) introduced a class of stronger valid inequalities than those of the type (3.27). We briefly explain the derivation of these strong valid inequalities for completeness, however, skip the proof of this procedure and refer the reader to Luedtke et al. (2010) and Luedtke (2014).

Firstly, given $\boldsymbol{\alpha}$, we solve the following problem for every scenario $s \in S$ and obtain their the objective function value, $h^s(\boldsymbol{\alpha})$:

$$h^s(\boldsymbol{\alpha}) = \text{minimize}\left\{\boldsymbol{\alpha}\mathbf{z} : \mathbf{z} \in \bar{\mathcal{X}}, W^s\mathbf{u} \geq \mathbf{b}^s - T^s\mathbf{z}, \mathbf{u} \geq 0\right\}. \tag{3.28}$$

Note that any $\bar{\mathcal{X}} \supseteq \mathcal{X}$ is suitable to generate a valid feasibility cut. In our implementation, we obtain $\bar{\mathcal{X}}$ by relaxing the integrality restrictions on $\mathcal{X}$ and solve the following linear program to obtain $h^s(\alpha)$, where $\sigma_d^{(1)}\ d \in I \setminus J$, $\sigma_d^{(2)}\ d \in J$, $\sigma_d^{(3)}\ d \in J$ and $\sigma_d^{(4)}\ d \in J$ are the coefficients of an extreme ray of the dual feasible region of the $(\mathbf{ISSP})$:

$$h^s(\boldsymbol{\alpha}) = \quad \text{minimize} \quad \sum_{j\in J} R_j\gamma_j^k\left(\sigma_j^{(2)} + \sigma_j^{(3)}\right) - \sum_{j\in J}\sum_{l\in L} y_{jl}d_j^k\sigma_j^{(4)} \tag{3.29a}$$

$$\text{subject to} \quad \text{(3.19b)} - \text{(3.19e)}. \tag{3.29b}$$

Observe that for a given $k \in S$ we solve $|S|$ many problems of the form (3.29) and obtain $|S|$ many $h(\boldsymbol{\alpha})$ values. Let $\boldsymbol{\eta}$ be a permutation of S describing a non-decreasing ordering

of $h(\boldsymbol{\alpha})$ values, i.e.,

$$h^{\eta_1}(\boldsymbol{\alpha}) \geq h^{\eta_2}(\boldsymbol{\alpha}) \geq ... \geq h^{\eta_{|S|}}(\boldsymbol{\alpha}).$$

To simplify the notation, let $h^s(\boldsymbol{\alpha}) = h^s$ for all $s \in S$. Consider the following inequality where $s = k$,

$$\boldsymbol{\alpha}\mathbf{z} + h^k \beta^k \geq h^k. \tag{3.30}$$

The equation (3.30) is a valid feasibility cut. Furthermore, it is stronger than the feasibility cut given in (3.27). To see this, note that $\boldsymbol{\alpha}\mathbf{z} \geq \zeta,\ \forall \mathbf{z} \in P(\omega^s)$ when $\hat{\beta}^k = 0$ by the equation (3.27) and $h^k = \min_{\mathbf{z}}\{\boldsymbol{\alpha}\mathbf{z} : \mathbf{z} \in P(\omega^k) \cap \bar{\mathcal{X}}\}$ is always greater than or equal to ζ. The equation (3.30) is equivalent to the equation (3.27) when $\bar{\mathcal{X}} = \mathbb{R}$.

Having solved $|S|$ many problems of the form (3.29), we have $|S|$ many inequalities of the form (3.30). Note that all these inequalities are valid for $\mathcal{F}$. Using these inequalities, we define set $\mathcal{K}$ as

$$\mathcal{K} = \Big\{(\mathbf{z}, \boldsymbol{\beta}) \in \mathcal{X} \times \{0,1\}^{|S|} \ :\ \text{(3.8h)},\ \boldsymbol{\alpha}\mathbf{z} + h^{\eta_s}\beta^{\eta_s} \geq h^{\eta_s},\ s \in S\Big\}. \tag{3.31}$$

Luedtke (2014) studied this special mixed integer set and obtained a tighter and more compact representation:

$$\mathcal{K} = \Big\{(\mathbf{z}, \boldsymbol{\beta}) \in \mathcal{X} \times \{0,1\}^{|S|} \ : \\ \text{(3.8h)},\ \boldsymbol{\alpha}\mathbf{z} + (h^{\eta_j} - h^{\eta_{\kappa+1}})\beta^{\eta_j} \geq h^{\eta_j},\ j \in 1, ..., \kappa\Big\},$$

where $\kappa \doteq \max\{t : \sum_{i=1}^{t} p^i \leq \epsilon\}$. The inequalities of $\mathcal{K}$ where $s = \eta^{\kappa+1}, ..., \eta^{|S|}$ are excluded in this description since they are redundant because of (3.8h). Furthermore, Luedtke et al. (2010) proposed stronger set of inequalities for this set, following from (Atamtürk et al., 2000; Günlük and Pochet, 2001):

$$\left\{\boldsymbol{\alpha}\mathbf{z} + \sum_{j=1}^{m}(h^{t_j} - h^{t_{j+1}})\beta^{t_{j+1}} \geq h^{t_j},\ \forall\, \mathcal{T} = \{t_1, ..., t_m\} \subseteq \{\eta_1, ..., \eta_\kappa\}\right\}, \tag{3.32}$$

where $h^{t_{m+1}} = h^{\eta_{\kappa+1}}$. These inequalities are referred as strengthened star inequalities in Luedtke et al. (2010), and they are valid for $\mathcal{F}$. There are exponentially many inequalities of the form (3.32). However, there are efficient ways of separating these inequalities given a first-stage decision $(\hat{\mathbf{y}}, \hat{\mathbf{R}}, \hat{\boldsymbol{\beta}})$, (see, e.g., Atamtürk et al., 2000; Günlük and Pochet, 2001).

In our setting, the separation problem can be solved much more efficiently since we add feasibility cuts only when an integer feasible $(\hat{\mathbf{y}}, \hat{\mathbf{R}}, \hat{\boldsymbol{\beta}})$ is obtained during the branch-and-cut algorithm. We make use of the exact separation procedure described in Bülbül et al. (2016). We let $\mathcal{T} = \{\eta_1\}$ if $\hat{\beta}^{\eta_1}$ is equal to 0. Otherwise we find the next scenario η_b, for $b \in \{2, ..., \kappa\}$ such that β^{η_b} is equal to 0 and let $\mathcal{T} = \{\eta_1, \eta_b\}$. In this way, we

can obtain the set $\mathcal{T}$ which gives the most violated inequality of the form (3.32) for the current solution $(\hat{\mathbf{y}}, \hat{\mathbf{R}}, \hat{\boldsymbol{\beta}})$. In our implementation, this cut has the following form:

$$\sum_{j\in J} R_j \gamma_j^k \left(\sigma_j^{(2)} + \sigma_j^{(3)}\right) - \sum_{j\in J}\sum_{l\in L} y_{jl} d_j^k \sigma_j^{(4)} + \sum_{i=1}^{|\mathcal{T}|} (h^{t_i}(\alpha) - h^{t_{i+1}}(\alpha))\beta^{t_i} \geq h_{t_i}(\alpha). \quad (3.33)$$

The following procedure describes how we generate a feasibility cut within our branch-and-cut algorithm. Whenever a missing feasibility cut is identified, GenerateFeaCut($\mathcal{C}$, $\boldsymbol{\sigma}$, s) is called and a valid cut is added to the description of $\mathcal{C}$.

Algorithm 2: Procedure: GenerateFeaCut($\mathcal{C}$, $\boldsymbol{\sigma}$, s)

input : $\mathcal{C}$ - the set of feasibility cuts added so far, σ - extreme ray vector, s - current scenario index.
output: Add an feasibility cut to the description of $\mathcal{C}$.

```
for k ∈ S do
    Solve (3.29) and obtain h^k(α);
end
Solve the separation problem to obtain the most violated feasibility cut;
Add a feasibility cut of the form (3.33) to the description of C;
```

We next present the pseudo-code of the Benders decomposition-based branch-and-cut algorithm.

Algorithm 3: Benders decomposition-based branch-and-cut algorithm for $(\mathbf{RA-PNDP})$

$\mathcal{C} \leftarrow \mathbb{R}_+^{|J|\times|I|} \times \mathbb{R}_+^{|J|} \times \{0,1\}^{|S|}$; $\mathcal{D} \leftarrow \mathbb{R}_+^{|J|\times|I|} \times \mathbb{R}_+^{|J|} \times \{0,1\}^{|S|} \times \mathbb{R}_+^{|S|}$;
while *CutFound = True* **do**
  Solve **RMP**($\mathcal{C}$, $\mathcal{D}$);
  if *infeasible* **then**
    The original problem $(\mathbf{DEF})$ is infeasible;
    break;
  else
    CutFound $\leftarrow$ False, FeaFlag $\leftarrow$ False;
    Let $\left(\hat{\mathbf{y}},\ \hat{\mathbf{R}},\ \hat{\boldsymbol{\theta}},\ \hat{\boldsymbol{\beta}}\right)$ be an optimal solution of $(\mathbf{RMP})$;
    for $s \in S$ **do**
      if *FeaFlag* **then**
        break;
      end
      if $\hat{\beta}^s = 0$ **then**
        Solve $(\mathbf{ISSP})$;
        if $(\mathbf{ISSP})$ *is infeasible* **then**
          Let $\boldsymbol{\sigma}$ be an extreme ray vector associated with the dual of $(\mathbf{ISSP})$;
          generateFeaCut($\mathcal{C}$, $\boldsymbol{\sigma}$, s), FeaFlag $\leftarrow$ True, CutFound $\leftarrow$ True;
        else
          Let $\boldsymbol{\pi}$ be an optimal dual solution vector of $(\mathbf{ISSP})$ and $\hat{f}$ be the optimal objective function value of $(\mathbf{ISSP})$;
          if $\theta^s < \hat{f}$ **then**
            generateOptCut($\mathcal{D}$, $\boldsymbol{\pi}$, s, $\hat{\beta}_s$), CutFound $\leftarrow$ True;
          end
        end
      else
        Solve $(\mathbf{NISSP})$;
        Let $\bar{\boldsymbol{\pi}}$ be an optimal dual solution vector of $(\mathbf{NISSP})$ and $\hat{\bar{f}}$ the optimal objective function value of $(\mathbf{NISSP})$;
        if $\theta^s < \hat{\bar{f}}$ **then**
          generateOptCut($\mathcal{D}$, $\bar{\boldsymbol{\pi}}$, s, $\hat{\beta}_s$), CutFound $\leftarrow$ True;
        end
      end
    end
  end
end

3.3.3 Computational Enhancements and Implementation Details

We first briefly explain our preprocessing procedure. Different from Luedtke (2014) and Liu et al. (2016), we do not assume that $P(\omega^s)$, $s \in S$, is a non-empty polyhedron.

Therefore, we apply a preprocessing procedure to identify the scenarios for which the $(\mathbf{ISSP})$ is infeasible for any $(\hat{\mathbf{y}}, \hat{\mathbf{R}})$ value. To this end, we solve the following problem for all $s \in S$:

$$\Delta^s = \text{minimize} \Big\{ (\mathbf{q}^s)^\top \mathbf{u} : \mathbf{z} \in \bar{\mathcal{X}}, \quad W^s \mathbf{u} \geq \mathbf{b}^s - T^s \mathbf{z}, \quad \mathbf{u} \geq 0 \Big\}. \tag{3.34}$$

The rationale behind solving this problem is twofold. Firstly, if this problem is infeasible, then we can fix β^s to 1, since no first-stage solution yields a feasible $(\mathbf{ISSP})$ for this scenario. Other than that, if this problem is feasible for any $s \in S$, then its optimal objective function value Δ^s gives a valid lower bound for θ^s when β^s is equal to zero. Then the following inequalities are valid:

$$\theta^s \geq \Delta^s (1 - \beta^s). \tag{3.35}$$

In our setting, we let $\bar{\mathcal{X}} = \mathcal{X}$ to obtain better lower bounds and solve the following mixed-integer program for all $s \in S$:

$$\begin{aligned}
\Delta^s = \quad & \text{minimize} && \sum_{j \in J} \sum_{i \in M_j^s \setminus \{j\}} c_{ji}^s x_{ji}^s && && \text{(3.36a)} \\
& \text{subject to} && \text{(3.4b)} - \text{(3.4c)}, && && \text{(3.36b)} \\
& && \text{(3.1b)} - \text{(3.1f)}, && && \text{(3.36c)} \\
& && y_{jl} \in \{0, 1\}, && j \in J,\ l \in L, && \text{(3.36d)} \\
& && R_j \geq 0, && j \in J. && \text{(3.36e)}
\end{aligned}$$

After detecting under which scenarios $\{P(\omega^s) \cap \mathcal{X}\}$ is an empty set, we also use this information to skip solving some of the other problems within the branch-and-cut algorithm. Assume that we find the problem (3.36) is infeasible for some scenario $s \in S$. While generating the strong feasibility cuts we skip solving the corresponding problem (3.29) to compute $h^s(\alpha)$ value. Moreover, recall that the objective function value of the problem (3.24), $v^s(\boldsymbol{\alpha})$, is necessary to create a cut of the form (3.22). However, if (3.36) is infeasible, then (3.24) is infeasible as well. Therefore, in such cases, we skip generating optimality cuts of the form (3.22), and generate classical Benders optimality cuts.

Furthermore, we create some initial optimality and feasibility cuts and add them to the $(\mathbf{RMP})$ as part of the preprocessing procedure. We generate an arbitrary first-stage decision and solve the second-stage problems for every scenario to add optimality and feasibility cuts. While generating feasibility cuts, we add the corresponding most violated cut together with $|S| \times 0.05$ many other facet defining valid inequalities of the form (3.32). This simple process appears to improve the convergence of our algorithm significantly; we present results regarding this enhancement in Section (3.4).

We next describe the implementation details about our algorithm. Suppose that the $(\mathbf{ISSP})$ is found to be infeasible under a particular scenario. Then, the current first-stage decision $(\hat{\mathbf{y}}, \hat{\mathbf{R}}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}})$ cannot be the optimal solution to the original problem. Thus, one can generate a feasibility cut and avoid generating optimality cuts for the remaining scenarios where the corresponding second-stage problem is feasible. However, in our preliminary analysis, we found that faster convergence is achieved when we continue to add optimality cuts. Hence, we employ this option in our implementation of the branch-and-cut algorithm.

Lastly, recall that we describe two alternative types of adding optimality cuts when $\hat{\beta}^s = 1$ in Section (3.3.1): The strong cut presented in Liu et al. (2016) and the classical Benders optimality cut. In our preliminary analysis, we found that the difference between using the cuts of the form (3.22) and (3.26) is negligible in terms of the computational performance. We use the cuts of the form (3.26) in our branch-and-cut algorithm.

3.4 Computational Study

In this computational study, we aim to show the effectiveness of our branch-and-cut algorithm to solve different instances of $(\mathbf{RA-PNDP})$. We first describe the data generation process and then present our computational experiments.

3.4.1 Generation of the Problem Instances

We focus on a case study which concerns the hurricane disasters in the Southeast part of the US. It is originally constructed by Rawls and Turnquist (2010) and extended in Hong et al. (2015) by using a scenario generation scheme that takes into account the dependency structures inherent in disaster relief networks. We follow the scenario generation approach from Rawls and Turnquist (2010) and Hong et al. (2015), and here only describe the differences and additional parameters.

The network studied in Rawls and Turnquist (2010) contains 30 demand points and 55 links in which every link has a certain capacity to send relief supplies. In this work, on the other hand, we consider a different modelling approach which is based on coverage type assignments, not on the flows through the links in the network. In particular, we do not assume that there are 55 links in the network. Instead, whether a link exists between a demand node and a facility node is determined by considering the accessibility scores between these two nodes. Our approach does not set a limit on the amount of supplies that can be delivered through a particular link, therefore, link capacities are ignored in our study. In addition, we consider water as the single commodity that is stored and distributed in case of a disaster.

We next describe how we generate additional random parameters for our model. We

first calculate the base travel time between every demand node $i \in I$ and facility node $j \in J$, then perturbate the base value according to the link degradation level under scenario $s \in S$ to generate the accessibility scores ν_{ji}^s. A similar approach is used to generate random vector $\boldsymbol{\gamma}^s, s \in S$. We first identify the facility nodes that are affected by the disaster using node damage levels under scenario $s \in S$. Then a random variable which follows $U(0.7, 1.0)$ is generated for every affected facility node $j \in J$. This value is perturbated according to the node damage levels again to generate γ_j^s. Note that the link degradation and node damage levels are calculated following from Hong et al. (2015). Lastly, we assume that the shortage penalty cost is five times the purchasing cost.

3.4.2 Computational Performance of the Solution Method

The computational experiments are conducted on a workstation with two 2.3 GHz Intel(R) Xeon(R) E5-2630 CPUs and 64 GB RAM running on Windows 10 operating system. We implement and solve all problems in C++ using Concert Technology component library of IBM ILOG CPLEX 12.6 with its default settings.

A time limit of 3600 seconds is enforced and a single thread is used. We turn off the presolve option of CPLEX for the branch-and-cut algorithm. We use Lazy Constraint Callback feature of CPLEX to implement the subroutines of GenerateOptCut($\mathcal{D}$, $\boldsymbol{\pi}$, s, $\hat{\beta}$) and GenerateFeaCut($\mathcal{C}$, $\boldsymbol{\sigma}$, s). All results depicted in this section are calculated as averages over three instances. For the purpose of this computational study, the value of τ is set to a large enough value to obtain a fully connected network for all instances.

We benchmark our branch-and-cut algorithm against the $(\mathbf{DEF})$ for various ϵ, λ and α combinations. We present the numerical results on the performances of the $(\mathbf{DEF})$ and the branch-and-cut method in Tables 3.1 and 3.2. If optimality is not proven in the given time limit, we calculate an optimality gap by using the objective function of the best integer solution (UB) and the best current bound (LB) provided by CPLEX. In this case the relative optimality gap (ROG) is calculated as $(UB - LB)/LB$. The columns with headers of $(\mathbf{DEF})$ and **B&C** report the average solution times (in seconds) and/or ROG values obtained by using the corresponding method. The solution time excludes the preprocessing time for the branch-and-cut algorithm. The time information is skipped if all three instances are terminated because of the time limit and the ROG information is skipped if all three instances are solved to optimality in the given time limit.

We compare our branch-and-cut algorithm against the $(\mathbf{DEF})$ for the instance where $|J| = 10$ in Table 3.1. We see that the branch-and-cut algorithm is the clear winner in most of the cases. In particular, the branch-and-cut algorithm outperforms the $(\mathbf{DEF})$ in almost all instances, except for the ones where $\lambda = 1$ and $\alpha = 0.9$. However, even the largest of those instances where $|S| = 1000$ are solved in less than five minutes by the branch-and-cut algorithm as well. We observe that the instances where $\lambda = 0$ seem to

			Time [ROG (%)]					
			$\epsilon = 0.10$		$\epsilon = 0.15$		$\epsilon = 0.20$	
$\|S\|$	λ	α	(DEF)	B&C	(DEF)	B&C	(DEF)	B&C
200	0	-	85	6	214	9	325	12
	0.5	0.7	86	8	103	10	9	11
	0.5	0.9	20	7	6	6	3	3
	1	0.7	24	6	10	7	5	4
	1	0.9	6	4	3	3	3	2
400	0	-	1676	42	†††[3.00]	68	3235††[0.65]	54
	0.5	0.7	1010	78	873	99	41	74
	0.5	0.9	151	40	45	41	13	25
	1	0.7	416	56	49	20	20	14
	1	0.9	11	12	8	12	9	6
600	0	-	3395††[8.86]	310	†††[15.63]	721	†††[7.66]	297
	0.5	0.7	2935††[1.67]	306	3168††[0.67]	847	290	254
	0.5	0.9	1213	158	178	164	38	51
	1	0.7	1120	154	166	149	46	37
	1	0.9	32	33	22	32	22	17
800	0	-	†††[16.45]	1210	†††[22.88]	1777†[0.38]	††⋆[39.84]	1762
	0.5	0.7	††⋆[36.04]	962	†††[2.78]	2228†[0.38]	1155	1146
	0.5	0.9	2118†[0.17]	701	287	502	82	106
	1	0.7	2640[0.18]	1157	610	537	135	57
	1	0.9	52	120	43	71	36	27
1000	0	-	††⋆[47.47]	2052	†⋆⋆[74.33]	2964††[1.66]	†⋆⋆[70.32]	†††[1.59]
	0.5	0.7	††⋆[37.19]	1732	†††[5.77]	†††[0.97]	2039†[0.06]	3319††[0.02]
	0.5	0.9	2569†[0.29]	1848†[0.04]	845	980	141	229
	1	0.7	3171††[0.44]	1080	1708	2455††[0.17]	188	116
	1	0.9	73	306	59	146	49	47

†: Each dagger sign indicates one instance hitting the time limit with an integer feasible solution.
⋆: Each asterisk sign indicates one instance hitting the time limit with no integer feasible solution. In this case, we assume that the associated ROG value is 100%.

Table 3.1: Computational performances of the (DEF) and the branch-and-cut algorithm for varying ϵ, λ and α where $|J| = 10$.

be the toughest instances for both solution methods. The (DEF) is not able to solve any of these instances where the number of scenarios is 600 or larger in the given time limit. On the other hand, the branch-and-cut algorithm successfully solves all these instances to optimality where $\epsilon = 0.1$ and solves most of the instances to optimality where $\epsilon = 0.15$ or 0.2. It also provides very small optimality gaps for the instances that could not have been solved in the given time limit. The success of the branch-and-cut algorithm is not restricted to the instances where $\lambda = 0$. For example, we observe from Table 3.1 that the instances where $\lambda = 0.5$ and $\alpha = 0.7$ are also challenging for the (DEF) but the branch-and-cut algorithm attains success in solving these instances as well. Hence, we conclude that the computational performance of our branch-and-cut algorithm appears to be significantly better than the (DEF) on the instances where $|J| = 10$.

We next illustrate the computational effectiveness of our branch-and-cut algorithm

			Time [ROG (%)]					
			$\epsilon = 0.10$		$\epsilon = 0.15$		$\epsilon = 0.20$	
$\|S\|$	λ	α	(DEF)	B&C	(DEF)	B&C	(DEF)	B&C
	0	-	393	204	905	101	821	100
	0.5	0.7	213	458	217	456	28	169
100	0.5	0.9	91	187	72	174	23	31
	1	0.7	36	350	63	51	35	120
	1	0.9	15	39	13	29	12	14
	0	-	†††[9.46]	627	†††[10.62]	1608†[0.06]	†††[3.97]	1317
	0.5	0.7	1627	2598††[0.32]	1387	1821†[0.02]	107	166
200	0.5	0.9	468	1753	157	322	61	57
	1	0.7	312	†††[0.33]	106	1301	131	252
	1	0.9	51	167	54	91	54	59
	0	-	†††[21.28]	†††[2.27]	††⋆[52.60]	2874††[2.18]	†⋆⋆[76.29]	2968††[0.64]
	0.5	0.7	†††[2.16]	†††[1.68]	†††[1.05]	†††[0.95]	371	327
300	0.5	0.9	2491†[0.09]	†††[0.47]	877	419	212	56
	1	0.7	1083	†††[0.78]	387	†††[0.49]	434	448
	1	0.9	96	291	119	107	100	74
	0	-	⋆⋆⋆[100.00]	†††[4.29]	†⋆⋆[72.63]	†††[2.80]	⋆⋆⋆[100.00]	†††[3.03]
	0.5	0.7	†††[4.28]	†††[2.23]	†††[0.59]	†††[1.20]	1432	1301
400	0.5	0.9	2835†[0.19]	†††[0.85]	1297	1681†[0.07]	448	130
	1	0.7	2606†[0.11]	†††[1.43]	916	†††[1.06]	894	820
	1	0.9	196	748	208	145	197	113
	0	-	††⋆[58.54]	†††[4.48]	⋆⋆⋆[100.00]	†††[5.90]	†††[39.03]	†††[5.08]
	0.5	0.7	††⋆[37.80]	†††[3.36]	†††[2.27]	†††[2.44]	2069	1674
500	0.5	0.9	3248†[0.57]	†††[1.53]	2045	2215†[0.06]	699	224
	1	0.7	†††[0.67]	†††[1.70]	2076	†††[0.94]	1778	2432†[0.09]
	1	0.9	340	1490†[0.06]	407	747	363	214

†: Each dagger sign indicates one instance hitting the time limit with an integer feasible solution.
⋆: Each asterisk sign indicates one instance hitting the time limit with no integer feasible solution. In this case, we assume that the associated ROG value is 100%.

Table 3.2: Computational performances of the (**DEF**) and the branch-and-cut algorithm for varying ϵ, λ and α where $|J| = 30$.

against the (**DEF**) on the instances where $|J| = 30$. These instances are more challenging than those with $|J| = 10$, therefore we report results for the smaller instances with at most 500 scenarios. We observe from Table 3.2 that the performances of the both approaches are comparable and very much depend on the type of the problem instance. For example, branch-and-cut algorithm is again the clear winner while solving the instances where $\lambda = 0$. In this case, it successfully solves some of the instances where the (**DEF**) is unable to provide optimal solutions. Furthermore, it provides much smaller ROG values for the remaining instances with larger number of scenarios. On the other hand, we see from Table 3.2 that there is no clear winner for instances with remaining λ and α combinations. Even though the (**DEF**) is more successful especially for the case where $\lambda = 1$ and $\alpha = 0.9$, there are several instances in which branch-and-cut performs better than the (**DEF**).

			(DEF)				B&C w/o Preprocessing				B&C w/ Preprocessing			
$\|J\|$	ϵ	$\|S\|$	LP Gap	Root Gap	No. of Cuts	Root Time	LP Gap	Root Gap	No. of Cuts	Root Time	LP Gap	Root Gap	No. of Cuts	Root Time
10	0.1	200	19.1	16.4	2852	19	100	100	0	1	33.9	10.0	33	1
		400	18.2	15.4	5244	65	100	100	0	1	32.4	10.4	38	3
		600	18.4	15.5	8446	157	100	100	0	2	33.0	9.5	59	6
		800	18.6	16.1	11683	259	100	100	0	2	33.7	9.3	55	9
	0.2	200	7.2	5.9	2942	18	100	100	0	1	54.5	6.6	36	1
		400	6.1	4.9	6293	60	100	100	0	1	53.9	5.7	30	4
		600	7.1	5.8	9715	127	100	100	0	2	54.1	6.6	84	9
		800	6.8	5.6	12274	265	100	100	0	2	55.2	5.6	69	13
30	0.1	50	12.8	9.3	1150	9	100	47.4	5	0	47.1	19.7	28	1
		100	14.4	11.8	2302	25	100	50.5	4	1	43.1	16.7	44	1
		200	16.9	14.1	5473	114	100	51.2	4	1	45.5	15.2	10	2
	0.2	50	7.0	5.6	1270	10	100	64.4	3	1	61.7	10.8	29	1
		100	7.5	6.2	2282	26	100	65.0	3	1	59.9	7.1	33	2
		200	5.6	4.2	5293	89	100	72.9	3	1	66.7	7.0	93	4
Average			11.8	9.8	5516	89	100	82.2	2	1	48.2	10.0	46	4

Table 3.3: Impact of Preprocessing on Chance-Constrained Risk-Neutral Instances ($\lambda = 0$).

We further investigate the impact of the preprocessing on the performance of the branch-and-cut algorithm, and report our results in Tables 3.3 and 3.4. In the columns labeled as "LP Gap"and "Root Gap", we report the initial LP relaxation gap and the root gap, respectively. The initial LP relaxation gap is calculated as $(z^* - z^{\text{LP}})/z^*$ and the root gap is calculated as $(z^* - z^{\text{R}})/z^*$ where z^* is the optimal objective function value of the original problem, z^{LP} is the optimal objective function value of the initial LP relaxation of the original problem at the root node and z^{R} is the optimal objective function value of the LP relaxation after CPLEX cuts are added at the root node. We report the number of cuts that CPLEX adds during the processing of the root node and root relaxation time (in seconds) under the columns FLFLNo. of CutsFL FL and FLFLRoot TimeFL FL, respectively.

We present the results for the chance-constrained risk-neutral instances in Table 3.3. One thing that is apparent from Table 3.3 is that the branch-and-cut algorithm which does not utilize any of the preprocessing steps explained in Section (3.3.3) has very high LP relaxation and root gaps. This is due to the fact that the $(\mathbf{RMP})$ possess very little information about the original problem before any optimality or feasibility cuts are added. It is important to note that the cuts added during the preprocessing steps have little impact on the LP relaxation gap, however, they significantly increase the number of cuts generated by CPLEX. This results in a significant decrease in the root gap for the branch-and-cut algorithm. Another interesting observation is that lower root gaps for the $(\mathbf{DEF})$ on the instances where $|J| = 30$ is not enough to beat branch-and-cut algorithm, as it was

demonstrated in Table 3.2. This might be due to the much longer root relaxation times the $(\mathbf{DEF})$ has compared to the branch-and-cut algorithm.

			$(\mathbf{DEF})$				**B&C w/o Preprocessing**				**B&C w/ Preprocessing**			
$\|J\|$	ϵ	$\|S\|$	**LP Gap**	**Root Gap**	**No. of Cuts**	**Root Time**	**LP Gap**	**Root Gap**	**No. of Cuts**	**Root Time**	**LP Gap**	**Root Gap**	**No. of Cuts**	**Root Time**
10	0.1	200	6.6	5.2	2187	12	100	100	0	1	43.2	4.6	114	3
		400	5.4	4.2	5421	30	100	100	0	1	41.7	3.8	235	7
		600	6.0	4.7	7537	68	100	100	0	1	42.7	4.6	357	13
		800	5.6	4.3	10403	119	100	100	0	2	43.3	4.8	244	15
	0.2	200	0.6	0.1	3418	6	100	100	0	1	65.7	1.0	100	3
		400	0.7	0.1	3665	19	100	100	0	1	65.4	1.1	230	9
		600	0.6	0.1	5328	38	100	100	0	2	65.7	1.6	260	12
		800	0.6	0.1	7492	66	100	100	0	2	67.0	2.0	315	19
30	0.1	50	2.4	1.1	799	6	100	62.1	4	0	58.2	3.5	28	2
		100	2.9	1.7	1838	14	100	64.2	4	1	55.1	3.9	57	4
		200	3.5	2.4	4361	50	100	64.3	4	1	57.0	5.0	81	7
	0.2	50	1.0	0.2	869	4	100	77.0	3	1	74.1	1.8	30	2
		100	0.9	0.2	1604	12	100	77.2	3	1	72.9	1.4	71	5
		200	0.7	0.1	3126	35	100	82.9	3	1	78.2	1.4	113	8
Average			2.7	1.8	4146	34	100	87.7	2	1	59.3	2.9	160	8

Table 3.4: Impact of Preprocessing on Risk-Averse Instances ($\lambda = 0.5$, $\alpha = 0.7$).

The impact of preprocessing on the branch-and-cut algorithm for the risk-averse instances is very similar to the chance-constrained risk-neutral instances. As shown in Table 3.4, the root gap is significantly lower for the branch-and-cut algorithm in which preprocessing is performed. We see that the LP relaxation gaps for the branch-and-cut algorithm with preprocessing is higher than the ones associated with chance-constrained risk-neutral instances contrary to the root gap values. This result can be attributed to the fact that CPLEX generates more cuts while solving the risk-averse instances compared to the chance-constrained risk-neutral instances. This is another example which underlines the significance of the cuts generated by CPLEX during the branch-and-cut algorithm. We also observe from Table 3.4 that root relaxation times are much lower for the $(\mathbf{DEF})$ on average compared to chance-constrained risk-neutral instances. This might be one of the factors why $(\mathbf{DEF})$ performs better on the risk-averse instances.

As a last remark, we note that the LP relaxation gaps and the root gaps depicted in Tables 3.3 and 3.4 give more insights about why chance-constrained risk-neutral instances are tougher than the risk-averse instances for both the $(\mathbf{DEF})$ and the branch-and-cut algorithm. We see that average LP relaxation and root gaps are much lower for the latter instances under both of the solution methods. Therefore, implementing more effective preprocessing steps for the chance-constrained risk-neutral instances which would lower the root gap might increase the performance of the branch-and-cut algorithm.

Chapter 4

Conclusion and Future Work

In the first part of the thesis, we develop strong MIP formulations for two classes of individual chance-constrained linear programs with variable reliability levels / risk tolerances by leveraging recent methodological advances. A new type of cost function, which requires capturing VaR associated with a variable reliability level, is an additional contribution of our work. Our computational study attests to the effectiveness of the proposed MIP formulations in solving the problems of interest. Optimal solutions of large practical-size instances of these formulations are within the reach of modern off-the-shelf solvers. The application of the proposed models to a novel stochastic last mile distribution network design problem and the associated case study based on real-life data further substantiates this claim. We hope that our work will invoke more awareness about the availability of powerful modeling tools for individual chance-constrained optimization models with and without variable reliability levels.

In the second part, we study a pre-disaster relief network design problem under random demand and network conditions, and introduce a risk-averse two-stage stochastic optimization model. We develop a Benders decomposition-based branch-and-cut solution algorithm and provide computational results which demonstrate the effectiveness of our solution methods. However, several enhancement strategies can be employed in order to further improve the performance of our implementation. First of all, such a scenario decomposition-based algorithm can significantly benefit from parallelization; more specifically, cut generation procedures can be implemented concurrently for different scenarios. Furthermore, one could invest on finding better initial first-stage decisions (instead of considering arbitrary ones as in the current implementation of our algorithm), which are used to generate the initial feasibility and optimality cuts. This might possibly improve the convergence of our algorithm. As part of our ongoing work, we intend to make use of such enhancements. In addition, we are planning to perform an extensive analysis to evaluate the importance of the key features of the proposed risk-averse stochastic programming model for decision making purposes. In particular, we will investigate the

effects of enforcing a joint chance constraint on the feasibility of the second-stage problem and incorporating a risk measure into the objective function.

Bibliography

Acerbi, C. (2002). Spectral measures of risk: a coherent representation of subjective risk aversion. *Journal of Banking and Finance*, 26(7):1505–1518.

Ahmed, S. (2006). Convexity and decomposition of mean-risk stochastic programs. *Mathematical Programming*, 106(3):433–446.

Artzner, P., Delbaen, F., Eber, J. M., and Heath, D. (1999). Coherent measures of risk. *Mathematical Finance*, 9:203–227.

Atamtürk, A., Nemhauser, G. L., and Savelsbergh, M. W. (2000). The mixed vertex packing problem. *Mathematical Programming*, 89(1):35–53.

Balcik, B. and Beamon, B. (2008). Facility location in humanitarian relief. *International Journal of Logistics: Research and Applications*, 11(2):101–121.

Beraldi, P. and Bruni, M. E. (2009). A probabilistic model applied to emergency service vehicle location. *European Journal of Operational Research*, 196(1):323–331.

Bülbül, K., Kucukyavuz, S., Noyan, N., and Şen, H. (2016). A two-stage chance-constrained mean-risk stochastic programming model for single-machine scheduling. Working Paper.

Caunhye, A. M., Nie, X., and Pokharel, S. (2012). Optimization models in emergency logistics: A literature review. *Socio-Economic Planning Sciences*, 46(1):4–13. Special Issue: Disaster Planning and Logistics: Part 1.

Çelik, M., Ergun, O., Johnson, B., Keskinocak, P., Lorca, A., Pekgün, P., and Swann, J. (2012). Humanitarian logistics. *appears in Tutorials in Operations Research, Mirchandani P.(ed.), INFORMS, Maryland*, pages 18–49.

Dentcheva, D. (2006). *Probabilistic and Randomized Methods for Design under Uncertainty*, chapter Optimization Models with Probabilistic Constraints. Springer-Verlag, London. Editors: G. Calafiore and F. Dabbene.

Döyen, A., Aras, N., and Barbarosoğlu, G. (2012). A two-echelon stochastic facility location model for humanitarian relief logistics. *Optimization Letters*, 6:1123–1145.

Fábián, C. I. (2008). Handling cvar objectives and constraints in two-stage stochastic models. *European Journal of Operational Research*, 191(3):888–911.

Günlük, O. and Pochet, Y. (2001). Mixing mixed-integer inequalities. *Mathematical Programming*, 90(3):429–457.

Haneveld, W. K. K. and van der Vlerk, M. H. (2015). Stochastic programming lecture notes. Available on request (only as hard copy; view Table of contents at `http://www.rug.nl/feb/mhvandervlerk/spcourse/contents.pdf`.

Hong, X., Lejeune, M. A., and Noyan, N. (2015). Stochastic network design for disaster preparedness. *IIE Transactions*, 47(4):329–357.

Kahvecioglu, G. (2014). Stochastic last mile relief network design with resource reallocation. MS Thesis, Sabanci University.

Kall, P. and Wallace, S. (1994). *Stochastic programming*. Wiley-Interscience series in systems and optimization. Wiley.

Küçükyavuz, S. and Noyan, N. (2016). Cut generation for optimization problems with multivariate risk constraints. *Mathematical Programming*, 159(1):165–199.

Kucukyavuz, S. (2012). On mixing sets arising in chance-constrained programming. *Mathematical Programming*, 132(1-2):31–56.

Lejeune, M. and Noyan, N. (2010). Mathematical programming approaches for generating p-efficient points. *European Journal of Operational Research*, pages 590–600.

Lejeune, M. A. (2012). Pattern-based modeling and solution of probabilistically constrained optimization problems. *Operations Research*, 2(1):1356–1372.

Lejeune, M. A. and Shen, S. (2016). Multi-objective probabilistically constrained programs with variable risk: Models for multi-portfolio financial optimization. *European Journal of Operational Research*, 252(2):522 – 539.

Liberatore, F., Pizarro, C., Blas, C. S., Ortuño, M. T., and Vitoriano, B. (2013a). *Decision Aid Models for Disaster Management and Emergencies*, chapter Uncertainty in Humanitarian Logistics for Disaster Management. A Review, pages 45–74. Atlantis Press, Paris.

Liberatore, F., Pizarro, C., de Blas, C., Ortuño, M., and Vitoriano, B. (2013b). Uncertainty in humanitarian logistics for disaster management. a review. In Vitoriano, B., Montero, J., and Ruan, D., editors, *Decision Aid Models for Disaster Management and Emergencies*, volume 7 of *Atlantis Computational Intelligence Systems*, pages 45–74. Atlantis Press.

Liu, X., Küçükyavuz, S., and Luedtke, J. (2016). Decomposition algorithms for two-stage chance-constrained programs. *Mathematical Programming*, 157(1):219–243.

Luedtke, J. (2014). A branch-and-cut decomposition algorithm for solving chance-constrained mathematical programs with finite support. *Mathematical Programming*, 146(1-2):219–244.

Luedtke, J., Shabbir, A., and Nemhauser, G. (2010). An integer programming approach for linear programs with probabilistic constraints. *Mathematical Programming*, 122(2):247–272.

Mete, H. O. and Zabinsky, Z. B. (2010). Stochastic optimization of medical supply location and distribution in disaster management. *International Journal of Production Economics*, 126(1):76–84.

Miller, N. and Ruszczyński, A. (2011). Risk-averse two-stage stochastic linear programming: Modeling and decomposition. *Operations Research*, 59(1):125–132.

Noyan, N. (2012). Risk-averse two-stage stochastic programming with an application to disaster management. *Computers and Operations Research*, 39(3):541–559.

Noyan, N., Balcik, B., and Atakan, S. (2016). A stochastic optimization model for designing last mile relief networks. *Transportation Science*, 50(3):1092–1113.

Prékopa, A. (1980). Network planning using two-stage programming under uncertainty. In *Recent results in stochastic programming (Proc. Meeting, Oberwolfach, 1979)*, volume 179 of *Lecture Notes in Econom. and Math. Systems*, pages 215–237. Springer, Berlin.

Prékopa, A. (1995). *Stochastic Programming*. Kluwer Academic, Dordrecht, Boston.

Prékopa, A. (2003). *Handbooks in Operations Research and Management Science, vol.10*, chapter Probabilistic Programming, pages 267–351. Elsevier, Amsterdam. Editors: A. Ruszczyński and A. Shapiro.

Prékopa, A. and Unuvar, M. (2015). Single commodity stochastic network design under probabilistic constraint with discrete random variables. *Operations Research*, 63(6):1512–1527.

Rawls, C. G. and Turnquist, M. A. (2010). Pre-positioning of emergency supplies for disaster response. *Transportation Research Part B: Methodological*, 44(4):521–534.

Rawls, C. G. and Turnquist, M. A. (2011). Pre-positioning planning for emergency response with service quality constraints. *OR Spectrum*, 33:481–198.

Rengarajan, T., Dimitrov, N., and Morton, D. (2013). Convex approximations of a probabilistic bicriteria model with disruptions. *INFORMS Journal on Computing*, 25(1):147–160.

Rengarajan, T. and Morton, D. (2009). Estimating the efficient frontier of a probabilistic bicriteria model. In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, pages 494–504.

Rockafellar, R. (2007). Coherent approaches to risk in optimization under uncertainty. In *Tutorials in Operations Research*, pages 38–61. INFORMS.

Rockafellar, R. and Uryasev, S. (2000). Optimization of conditional value at risk. *The journal of risk*, 2(3):21–41.

Ruszczyński, A. (2002). Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra. *Mathematical Programming*, 93(2):195–215.

Salmerón, J. and Apte, A. (2010). Stochastic optimization for natural disaster asset prepositioning. *Production and Operations Management*, 19(5):561–574.

Schultz, R. and Tiedemann, S. (2006). Conditional value-at-risk in stochastic programs with mixed-integer recourse. *Mathematical Programming*, 105(2):365–386.

Shapiro, A., Dentcheva, D., and Ruszczyński, A. (2009). *Lectures on stochastic programming: modeling and theory*. The society for industrial and applied mathematics and the mathematical programming society, Philadelphia, USA.

Shen, S. (2014). Using integer programming for balancing return and risk in problems with individual chance constraints. *Computers & Operations Research*, 49:59 – 70.