ROBUST OPTIMIZATION MODELS WITH MIXED-INTEGER UNCERTAINTY SETS

By

MEHDI ANSARI HADIPOUR

Bachelor of Science in Industrial Engineering
Sharif University of Technology
Tehran, Iran
2015

Master of Science in Industrial Engineering
Sharif University of Technology
Tehran, Iran
2017

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
May, 2023

ROBUST OPTIMIZATION MODELS WITH MIXED-INTEGER UNCERTAINTY SETS

Dissertation Approved:

Dr. Juan S. Borrero

Dissertation Advisor

Dr. Balabhaskar Balasundaram

Dr. Katie Jurewicz

Dr. Jorge Gonzalez Estrella

ACKNOWLEDGMENTS

Dedicated to

my beloved parents, Hadi and Nayereh.

Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Juan S. Borrero, for his invaluable guidance, support, and encouragement throughout my Ph.D. journey. Dr. Borrero has generously shared his expertise and knowledge to help me toward my academic achievements. His flexibility and willingness to accommodate my career goals have been truly appreciated. I consider myself extremely fortunate to have had the opportunity to work with Dr. Borrero during my Ph.D. program.

I am also immensely grateful to the IEM department for providing me with the opportunities and resources that have enabled me to pursue my doctoral degree. I extend my heartfelt appreciation to Drs. Balabhaskar Balasundaram, Sunderesh Heragu, and Austin Buchanan whose expertise and passion for teaching have been prominent in shaping my research interests and developing my skills. I am proud to be associated with the IEM department, and I am grateful for the role it has played in my academic and professional growth.

I would also like to acknowledge the efforts of my committee members, Drs. Balabhaskar Balasundaram, Katie Jurewicz, and Jorge Gonzalez Estrella in evaluating my progress during my Ph.D. program. Their feedback and assessments have been invaluable in ensuring that I stayed on track and met the program's expectations.

Finally, and above everything, I give a special feeling of gratitude to my parents, Hadi Ansari and Nayereh Aminizadeh for their unwavering love, support, and encouragement throughout my life. Their constant motivation and belief in me have been essential in helping me pursue my dreams and reach my goals. I would also like to acknowledge my sister, Golnaz, who has been my constant cheerleader and source of inspiration. I cannot thank them enough for their sacrifices, guidance, and dedication, which have shaped my character and propelled me toward success. I am blessed to have a family that has always been my pillar of strength and steadfast supporter.

Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Name: MEHDI ANSARI HADIPOUR

Date of Degree: May, 2023

Title of Study: ROBUST OPTIMIZATION MODELS WITH MIXED-INTEGER UNCERTAINTY SETS

Major Field: INDUSTRIAL ENGINEERING AND MANAGEMENT

Abstract: In many real-world decision-making problems, some parameters are uncertain and decision-makers have to solve optimization models under uncertainty. In this study, we address two specific classes of problems with uncertain parameters and we are interested in solutions that are robust under any realization of uncertainty. The first one is a class of minimum-cost flow problems where the arcs are subject to multiple ripple-effect disruptions that increase their usage cost. The locations of the disruptions' epicenters are uncertain, and the decision-maker seeks a flow that minimizes cost assuming the worst-case realization of the disruptions. We evaluate the damage to each arc using different methods in which the arcs' costs post-disruptions are represented with a mixed-integer feasible region. In the second class, we propose a novel problem that considers a decision-maker (a government agency or a public-private consortium) who seeks to efficiently allocate resources in retrofitting and recovery strategies to minimize social vulnerability under an uncertain tornado. As tornado paths cannot be forecast reliably, we model the problem using a two-stage robust optimization problem with a mixed-integer nonlinear uncertainty set that represents the tornado damage.

In this study, our primary objective is to develop a mathematical and algorithmic framework for modeling and solving these specific problems. To accomplish this goal, we utilize robust optimization to formulate these problems; develop decomposition algorithms inspired by methods of mixed-integer optimization; and exploit the geometrical characteristics of the uncertainty sets to enhance the formulations and algorithms. We test our proposed approaches over real datasets and synthetic instances. The numerical experiments show that our methods provide sound decision-making strategies under uncertainty and achieve orders of magnitude improvements in computational times over standard approaches from the literature.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER I

INTRODUCTION

Robust optimization is a field of study that has attracted significant interest in recent years because of its numerous practical applications in real-world decision-making problems under uncertainty such as finance, transportation, energy, and healthcare. Due to lack of historical data, data inaccuracies, and model errors resulting in insufficient knowledge about systems with uncertain behaviors, decision-makers are often required to make critical choices based on uncertain data. The need to optimize the performance of a system while considering the unpredictability of the parameters involved makes robust optimization an essential tool for decision-making.

Robust optimization methodology offers a promising avenue to tackle uncertainty that cannot be adequately modeled using probabilistic methods, often due to a scarcity of historical data or the unpredictable behavior of complex systems. Compared to methodologies like stochastic optimization or simulation, robust optimization adopts a more conservative approach to studying such systems. In particular, robust optimization is concerned with finding optimal solutions that are insensitive to changes in uncertain parameters, ensuring that the solution remains feasible under all potential scenarios. To this end, a decision-maker must identify the worst realization of uncertain parameters to find robust optimal solutions to the problem. In other words, she/he ensures that there does not exist a realization of uncertain parameters that gives a result worse than the robust optimal value.

This dissertation explores the application of robust optimization models to identify resilient

strategies for coping with unpredictable and rare adversarial disruptions such as natural disasters. The lack of adequate data for mathematical modeling of these events makes them challenging to address. Our proposed approach is to use robust optimization to create effective and efficient preparation, response, and resource allocation strategies before the occurrence of uncertainty. By doing so, we aim to provide practical solutions that help communities and individuals manage uncertainties and strengthen their resilience against these types of disruptions.

This study is motivated by two specific classes of problems. The first one is a class of adversarial minimum-cost flow problems where the arcs are subject to multiple ripple effect disruptions that increase their usage cost. The locations of the disruptions' epicenters are uncertain and the decision-maker seeks a flow that minimizes cost assuming the worst-case realization of the disruptions. This class of *minimum-cost flow problems under uncertainty* has a broad range of applications in military, telecommunication, and transportation problems. These minimum-cost flow problems can be studied as single-stage robust optimization problems since the decision-maker determine the flow through each arc before the realization of uncertainty, and there is a lack of historical data to reliably estimate probability distributions for the location of the epicenters.

The second class of problems of interest relates to identifying effective mitigation strategies for tornado hazards, in order to minimize their impact on the well-being of communities. Specifically, consider a decision-maker (a government agency or a public-private consortium) who seeks to minimize the social vulnerability after the occurrence of an uncertain tornado. Prior to a tornado, the decision-maker has the option to allocate resources toward retrofitting buildings in order to minimize potential damage. Moreover, after a tornado strikes, the decision-maker must decide whether to prioritize immediate recovery efforts for some affected buildings. This problem can be seen as a two-stage robust optimization problem in which the decision maker is allowed to wait and determine some part of decision variables after

the realization of uncertainty. Since many regions prone to tornadoes lack sufficient reliable historical data to apply probabilistic models that can forecast tornado paths, the decision-maker has to account for the worst-case scenario when making decisions related to tornado mitigation strategies.

The vast majority of the studies in the literature on robust optimization rely on the convexity assumption of uncertainty sets. However, for certain applications, such as those considered in this research, the uncertainty is better modeled by a non-convex set that contains possible realizations of uncertain parameters. Given the limitations of existing methods, the primary objective of this research is to develop a mathematical framework for efficiently modeling and solving this specific class of problems with mixed-integer uncertainty sets. We will explore the scarce existing research in this area by reviewing literature in Chapter II.

In this particular area of research, we focus on tackling min-max problems that involve decision-makers seeking to minimize an objective function whose parameters are part of a mixed-integer set. The min-max problem cannot be solved using conventional methods because the standard approach, which requires using duality assumptions, is not applicable for mixed-integer formulations in the inner problem. We, first, reformulate the min-max problem to a one-level problem using an epigraphic reformulation of the lower-level problem. However, this one-level reformulation can result in an exponential number of constraints regarding the number of scenarios present in the uncertainty set. To address this challenge, we employ decomposition algorithms that iteratively solves a relaxed version of the problem in the master problem. In each iteration of the algorithms, a subproblem must be solved to generate a new scenario in the uncertain set and accordingly add a new constraint in the next iteration to the master problem.

The performance of decomposition algorithms depends on the effectiveness of the solution approach for their subproblem. Therefore, this dissertation places significant emphasis on

studying the structure of the mixed-integer optimization model in the subproblem, with the goal of presenting efficient solution approaches. To achieve this objective, we propose alternative representations of the uncertainty sets for both classes that leverage valid cuts to enhance the quality of the mixed-integer subproblem. These representations enable us to obtain exact solutions for relatively large-scale settings of subproblems in every iteration of the decomposition algorithms.

1.1 Problem statement

We consider non-convex uncertainty sets in our study, in particular those that are formulated with mixed-integer programming (MIP). Consider the general optimization model $z^* = \min\{c^\top x : x \in X\}$, where x is the decision vector and X represents the set of feasible solutions. Suppose the parameter c in the objective function is uncertain and belongs to a non-convex set $\mathcal{U}$. We are interested in the robust solution of the problem when the worst realization of vector $c \in \mathcal{U}$ is revealed. So, we propose the following class of robust optimization problem

$$z^* = \min\Big\{ \max\{c^\top x : c \in \mathcal{U}\} : x \in X\Big\}. \tag{1.1}$$

Problem (1.1) gives the robust optimal value z^* when the worst scenario of $c \in \mathcal{U}$ happens.

We formulate and solve robust optimization problem (1.1) for special classes of MIP uncertainty sets arising from the *"Robust Minimum-Cost Flow Problems Under Multiple Ripple Effect Disruptions"* (RMCFP-RED). The uncertainty set $\mathcal{U}$ determines cost functions that calculate parameter c with MIP formulations. We develop a cutting-generation algorithm to solve the extended formulation of optimization problem (1.1) and enhance its performance with a strong reformulation of the subproblem. We identify the computational complexity of the problem for different structures of the uncertainty set.

Furthermore, we extend the methods solving (1.1) to address the two-stage version of

robust optimization problems. In two-stage robust optimization models, the decision-maker can determine part of the decision variables before the realization of uncertainty while the other part of variables are determined when uncertain parameters are revealed. In general, the two-stage robust optimization models are formulated as

$$w^* = \min \Big\{ c^\top x + \max \big\{ \min\{b^\top y : y \in F(x,u)\} : u \in \mathcal{U} \big\} : x \in X \Big\}, \tag{1.2}$$

where $x \in X$ is the first stage decision variable, $y \in F(x,y)$ is the second stage decision variable that belongs to the feasible set defined by x and the uncertain variable $u \in \mathcal{U}$.

We study an optimization problem of the form (1.2) to address *"Two-Stage Robust Optimization Approach for Enhanced Community Resilience under Tornado Hazards"*. We represent the uncertainty set $\mathcal{U}$ with a non-linear MIP formulation. We modify the column-and-constraint generation algorithm in the literature to solve a one-level reformulation of the problem (1.2). We analyze the resulting non-convex uncertainty set and develop a decomposition branch-and-cut algorithm to solve the corresponding subproblem. We also improve the performance of the solution approach by proposing valid cuts to the uncertainty set.

The remainder of this dissertation is presented as follows. Chapter II reviews the literature of robust optimization, bilevel programming, and our problems of interest. Chapter III studies the robust minimum-cost flow problems under multiple ripple effect disruptions. Chapter IV presents the two-stage robust optimization approach for enhanced community resilience under tornado hazards. Ultimately, ChapterV concludes this dissertation and discusses future extensions.

CHAPTER II

LITERATURE REVIEW

Over the past two decades, robust optimization has garnered extensive attention in the literature to study systems under uncertainty. The fundamental assumptions of robust optimization methods can be classified into three categories (Ben-Tal et al., 2009):

1. Decision variables reflect "here-and-now" and they are determined by solving a problem before obtaining real data in practical scenarios.

2. The decision-maker is entirely responsible for the outcomes of their decisions, with the exception of instances where uncertain parameters deviate from the assumed level of uncertainty after they are revealed.

3. Values within the uncertainty set are subject to the "hard constraints" defined in the model, which they must not breach.

These fundamental assumptions underscore the critical role of robust optimization in providing robust solutions that can withstand uncertainties and guarantee the feasibility of decision-making processes.

Soyster (1973) developed one of the first robust optimization models. Despite the approach's ability to ensure the robustness of solutions for all possible realizations of uncertain parameters, it may be overly conservative and might lead to outcomes that deviate significantly from nominal values. To tackle this concern, more recent works have proposed less conservative

approaches (Ben-Tal and Nemirovski, 1999; Bertsimas and Sim, 2004; Jalilvand-Nejad et al., 2016; Bertsimas et al., 2018).

Standard robust optimization models assume that the uncertain parameters belong to a convex set (Ben-Tal and Nemirovski, 1998, 1999, 2000, 2002; Bertsimas and Sim, 2004; Ben-Tal et al., 2006; Ben-Tal and Nemirovski, 2008; Bertsimas and Brown, 2009; Li et al., 2011; Bertsimas et al., 2016), which have many applications in scheduling (Lin et al., 2004), supply chain (Ben-Tal et al., 2005, 2011), inventory (Bertsimas and Thiele, 2006; Gorissen et al., 2015), portfolio optimization (Natarajan et al., 2009; Gregory et al., 2011; Moon and Yao, 2011), transportation (Yao et al., 2009), and power system problems (Xiong et al., 2017). Robust optimization problems where the uncertainty set depends on the decision-maker actions have also been studied, see Poss (2014); Nohadani and Sharma (2018).

Fewer works have considered mixed-integer uncertainty sets as the one that arises in this study. Bertsimas and Sim (2004) consider a class of problems where at most k out of n columns of a linear program are subjected to interval uncertainty and show that the problem can be reformulated as a linear program. Exact approaches for more general mixed-integer uncertainty sets can be seen as an extension of the integer L-shaped method (Laporte and Louveaux, 1993), in which a master problem solves a relaxation of the problem that considers a small subset of possible realizations of the uncertainty, while a subproblem (also referred to as a pessimization oracle) finds a worst-case realization of the uncertainty for the fixed solution vector obtained from the master problem (Mutapcic and Boyd, 2009; Ben-Tal et al., 2015; Ho-Nguyen and Kılınç-Karzan, 2018). When applied to our problem setting, each call to the subproblem requires solving a MIP, which imposes a heavy computational burden on this type of methods. Borrero and Lozano (2020) proposed an enhanced cutting-plane algorithm that considerably reduces the number of subproblems solved to optimality during the execution of the algorithm via a sampling mechanism and cut reoptimization.

2.1 Bilevel programming

Bilevel optimization can be adapted to formulate a robust optimization problem. A typical bilevel optimization problem involves upper- and lower-level of optimization problems where a *leader* and a *follower* solve sequential interdependent problems. Accordingly, there are two classes of decision vectors representing the leader's and follower's actions. Only optimal solutions of the lower-level problem that satisfies the upper-level constraints, are considered feasible to the leader problem. Two positions are commonly considered for the follower in bilevel problems; *optimistic* (weak), where the leader expects the follower to cooperate toward the best objective value in the upper-level, and *pessimistic* (strong), where the follower selects solutions for the worst-case realization of objective value in the leader problem. We refer the reader to Dempe et al. (2015) and Sinha et al. (2017) for details on bilevel programming.

Our problem setting could be seen as a special case of a bilevel mixed-integer program (BMIP) because they involve integer variables either at the upper- or lower-level. The main approaches for general BMIP are methods based on branching (DeNegre and Ralphs, 2009; Xu and Wang, 2014; Fischetti et al., 2017; Tahernejad et al., 2020) or cutting-plane algorithms based on an optimal-value-function reformulation of the follower's problem (Mitsos, 2010; Lozano and Smith, 2017a). In general, the bilevel optimization problem is NP-hard optimization problem, but some conditions can be considered to guarantee polynomial time complexity to verify the optimality of a solution (Hansen et al., 1992; Vicente et al., 1994).

2.2 Hedging against rippled disruptions

We review the existing studies associated with minimum-cost flow problems under ripple effect disruptions. The problem encompasses a wide array of applications stemming from different fields as communications, defense, transportation, among others.

The problem of finding an optimal set of disruptions for a given flow plan is related to the

maximal covering location problem (MCLP), which seeks to locate a series of facilities over a collection of sites to maximize the coverage of a weighted demand (Church and ReVelle, 1974; Church, 1984; Chazelle and Lee, 1986). Berman et al. (2010) showed that a one-facility planar version of MCLP can be solved in polynomial time by framing the problem in terms of a discrete location model. Berman and Krass (2002) consider problems with gradual covering where partial coverage of demands is given by a step function; non-increasing continuous gradual covering functions are studied in Berman et al. (2003). Some applications of gradual coverage include radio, TV, and cellular coverage, as well as emergency care facilities (Drezner et al., 2004). Berman et al. (2009) consider problems in which several facilities can *cooperate* to cover any given demand and Álvarez-Miranda and Sinnl (2019) propose MIP formulations for cooperative covering problems with gradual covering. In addition, the combination of two relaxations for continuous gradual maximal covering location problems have been recently discussed by Bagherinejad et al. (2018), Berman et al. (2019), Karatas and Dasci (2020). In Section 3.4 we present a strong formulation for the problem of finding an optimal set of disruptions for a given flow plan, which can be used to solve cooperative and planar gradual MCLPs.

Robust minimum-cost flow problems under multiple ripple effect disruptions, in this dissertation, could be seen as a special case of a general bilevel mixed-integer program (BMIP), in which a *leader* and a *follower* solve sequential interdependent optimization problems. Several studies have used interdiction techniques and bilevel programming to model and solve the problems where networks are subjected to disruptions that can impact the follower's objective, feasible region, or both (Israeli and Wood, 2002; Royset and Wood, 2007; Borrero et al., 2016; Lozano and Smith, 2017a; Borrero and Lozano, 2020). However, to the best of our knowledge, the minimum-cost flow problems under ripple effect disruptions have not been studied with robust optimization assumptions.

2.3 Tornado hazard mitigation under uncertainty

The determination of retrofitting and recovery strategies for tornado hazards, particularly using optimization, is a relatively new topic in the literature. Wen (2021) proposes a multi-objective optimization model to retrofit against tornado hazards. Their model, however, is single-stage, does not considers recovery, does not consider uncertain tornado paths, and aggregates all of the uncertainty of the problem into the parameters of the deterministic model. Simulation, on the other hand, is more commonly used to study the behavior of tornadoes, particularly to evaluate the impact of retrofitting strategies, see for example Strader et al. (2016); Wang et al. (2017); Masoomi and van de Lindt (2018); Fan and Pang (2019); Wang et al. (2021); Stoner and Pang (2021). These works, however, do not explicitly consider the decision problem of allocating resources in two stages of retrofitting and recovery.

Ben-Tal et al. (2004) extend the scope of standard single-stage robust optimization problems by introducing the adjustable robust optimization methodology, where a part of decision variables must be determined before the realization of the uncertainty set. The rest of the decision variables are chosen when the uncertain parameters are revealed. The two-stage approach provides a less conservative framework to deal with the uncertainty and it can model a broad range of applications in different areas such as transportation (Gabrel et al., 2014; Pu and Zhan, 2021; Rahmati et al., 2022), networks (Atamtürk and Zhang, 2007; Nguyen et al., 2021), investment (Takeda et al., 2008), and power systems problems (Zhao and Zeng, 2012; Jiang et al., 2012; Bertsimas et al., 2012; Shams et al., 2021; Li et al., 2021).

The closest models in the literature to the present work are two-stage robust optimization models for (general) disaster planning, see Yuan et al. (2016); Ma et al. (2018); Matthews et al. (2019); Velasquez et al. (2020); Cheng et al. (2021). These models, however, are geared towards supply and distribution problems in networks and do not capture the specific patterns of the disasters (such as, e.g., paths or ripple effects). Consequently, these models cannot be

adapted to deal with tornadoes and the specific retrofitting and recovery setting we study here.

In general, two-stage robust optimization problems are computationally expensive. To efficiently solve them, decomposition methods are widely employed, under the assumption that the second-stage problem is a linear programming problem (Thiele et al., 2009; Zhao and Zeng, 2012; Jiang et al., 2012; Gabrel et al., 2014). Zeng and Zhao (2013) presents the column and constraint generation method (C&CG) as an alternative to address this class of robust problems. They showed the C&CG can outperform previous decomposition for many problem settings. The C&CG approach has become a popular tool to solve two-stage robust optimization in the past decade (An et al., 2014; An and Zeng, 2015; Jabr et al., 2015; Neyshabouri and Berg, 2017; Ding et al., 2017; Yuan et al., 2016; Matthews et al., 2019; Velasquez et al., 2020; Cheng et al., 2021). An important feature of existing applications of the C&CG method is that the second-stage optimization problem is convex. In contrast, in our problem the second-stage problem is an IP problem and the uncertainty set is mixed-integer non-linear. Therefore, the standard 'dualize and combine' approach that is used in the literature to solve the subproblem in the C&CG does not apply to our case.

CHAPTER III

ROBUST MINIMUM-COST FLOW PROBLEMS UNDER MULTIPLE RIPPLE EFFECT DISRUPTIONS

3.1 Motivation

We study a class of robust minimum-cost flow problems, in which a decision maker, which we refer to as the defender, moves commodities between supply and demand nodes of a network. The geographical region where the network is located is vulnerable to a series of uncertain disruptions that affect the arcs of the network, increasing their cost. We consider disruptions that create a ripple effect for which arcs located closer to the epicenter of the disruption suffer a greater increase in their cost.

The defender's objective is to move flow between the supply and demand nodes at minimum cost and to decide the operation before the realization of the disruptions. Since the locations of the disruptions' epicenters are uncertain, the defender seeks to find a robust optimal solution. That is, for each possible flow solution, the defender assumes that the *worst-case* realization of the disruptions is going to happen, and he/she hedges against uncertainty by finding a flow such that its worst-case cost after the disruptions is minimized.

Formally, we consider a directed network $G = (N, A)$ located in a 2-dimensional space, where N is the set of nodes and $A \subseteq N \times N$ is the set of arcs. Let $\boldsymbol{b} = (b_i \colon i \in N)$ be a vector representing the supply/demand of each node in the network, where $\sum_{i \in N} b_i = 0$. Let $\boldsymbol{B} \in \{-1, 0, 1\}^{|N| \times |A|}$ denote the node-arc adjacency matrix of the network, and let

Parts of this document are reprinted with permission from Ansari et al. (2023).

$\boldsymbol{f} = (f_a \colon a \in A)$ be a nonnegative vector representing the flow capacity of arcs. We define the *robust minimum-cost flow problem under ripple effect disruptions* (RMCFP-RED) as the following optimization problem:

$$z^* = \min \left\{ \max\{c^\top y : c \in \mathcal{U}\} : \boldsymbol{B}y = \boldsymbol{b}, y \leq \boldsymbol{f}, y \in \mathbb{R}_+^{|A|} \right\}, \tag{3.1}$$

where $y = (y_a \colon a \in A)$ represents the amount of flow through each arc and $c = (c_a \colon a \in A)$ is the vector of costs resulting after the realization of the disruptions. We model the ripple effect disruptions by ensuring that c belongs to an *uncertainty set* $\mathcal{U} \subset \mathbb{R}^{|A|}$ that contains all the possible cost vectors resulting from the realization of $m \geq 1$ uncertain disruptions. We assume that disruption j induces $r_j \geq 1$ *ripples*, denoted by $R_{jk} \subset \mathbb{R}^2$, for $j \in [m] = \{1, \ldots, m\}$, $k \in [r_j] = \{1, \ldots, r_j\}$. An arc $a \in A$ belongs to ripple R_{jk} (which with a slight abuse of notation we denote by $a \in R_{jk}$) if the distance between the midpoint of the arc and the epicenter of disruption j is between $q_{j,k-1}$ and q_{jk}, where $0 = q_{j0} \leq q_{j1} < \ldots < q_{jr_j}$ are given and known for all $j \in [m]$. For simplicity, we focus on the induced ℓ^1-norm distance on the plane, and include remarks about the ℓ^2 and ℓ^∞ norms as needed.

We assume that there is an initial cost of moving flow thru arc $a \in A$, denoted by c_a^0, and for any arc $a \in R_{jk}$, there is a corresponding cost increase d_{jk}, where $d_{j1} > \ldots > d_{jr_j}$ are given and known for any $j \in [m]$. Also, assume that the defender knows the values of r_j, q_{jk}, and d_{jk} with certainty. This assumption is justified because the defender can know their value with precision from previous analysis, e.g., in the case of an earthquake of a given magnitude it is possible to have a reliable estimate on its effects on the ground; or in case of a bomb or explosion, it is possible to reliably estimate its damage on its surroundings. Note that even if the defender only has incomplete information about these values, for instance, interval estimates, the robust approach allows us to interpret r_j, q_{jk}, and d_{jk} as the largest possible number of ripples, ripple sizes, and cost increases, respectively, across all possible values.

We analyze two alternative cost functions to compute the effect of multiple disruptions on a given arc. The first cost function, referred to hereafter as the *linear model*, assumes that the damage of different disruptions on a single arc is cumulative. Thus, after the realization of the disruptions, the initial cost of arc $a \in A$ is increased by $\sum_{(j,k):\, a \in R_{jk}} d_{jk}$. The second cost function, referred to hereafter as the *maximum model*, assumes that the damage of different disruptions on a single arc is given by the maximum damage among the disruption affecting the arc, i.e., the initial cost of arc $a \in A$ is increased by $\max_{(j,k):\, a \in R_{jk}} \{d_{jk}\}$.

Figure 3.1 illustrates our problem setting with an example having two disruptions. The left disruption consists of three ripples, while the right disruption consists of two ripples. Arc (1,2) is closer to the epicenter of the left disruption than arcs $(s, 1)$ and $(s, 2)$, resulting in a greater increase to its cost.

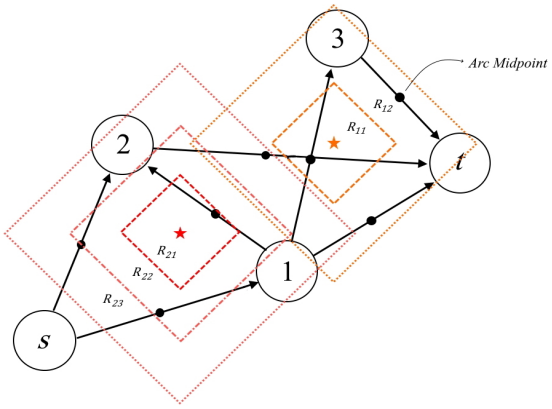


Figure 3.1: Illustrative example of a network subject to two disruptions under the ℓ^1 norm. The first one generates $r_1 = 3$ ripples and the second one generates $r_2 = 2$ ripples.

Our problem setting encompasses a wide array of applications stemming from different fields such as communications, defense, and transportation, among others. We motivate our study by presenting two examples of such problems below.

Supply operations in hostile territory. A military operation requires moving flow in a network located in enemy territory. The physical infrastructure is susceptible to bombs/artillery attacks from an adversary, which disrupt the arcs of the network following

a ripple effect, e.g., if the adversary air-drops 50-pound bombs the blast creates at least four ripples with decreasing effects of destruction (see for instance Cross et al. (2016)). The damage inflicted on the physical infrastructure is modeled by an increased cost of traversing the arcs in the network. Consider for example that the cost of an arc is proportional to the time it takes to traverse it. The flow is moved using all terrain vehicles that are capable of navigating over obstacles and rubble; however, traversing a heavily damaged road takes more time than traversing a slightly damaged road. The combined effect of multiple disruptions is adequately captured by our linear model which provides a worst-case cost analysis for the operation.

Supply operations after a disaster. A decision-maker seeks to move commodities in a network after a major disaster has caused leaks of hazardous materials (e.g., natural gas, carbon dioxide, biological agents, see Sengul et al. (2012)) at unknown locations, i.e., the leak locations might not be known when emergency supply plans are made. As a result, the operating crew is prepared with an assortment of protective equipment that ranges from light protective gear (e.g., a face mask) to heavy protective gear (e.g., a hazmat suit). Before traversing each arc, the crew measures the risk of exposure to hazardous material for the specific arc and uses appropriate protective gear while traversing the arc. The risk level follows a ripple effect, where arcs located closer to the epicenter of the spill are considered riskier and require stricter safety protocols thus increasing the transportation cost, which is assumed to be proportional to the traveling time over the arcs (Patel and Horowitz, 1994; Zhang et al., 2000; Verma and Verter, 2007). The effect of the interaction between hazardous materials is unlikely to be additive (e.g., a combination of hydrogen sulfide and carbon dioxide –both toxic gases in confined spaces– in the same area does not result in a more dangerous chemical hazard). As a result, our maximum model is better suited than the linear model to represent this problem and provides a worst-case cost analysis for the operation.

Borrero and Lozano (2020) studied a special case of RMCFP-RED in which there is a

single disruption under the linear cost model. They propose an exact cutting-plane algorithm akin to the integer L-shaped method (Laporte and Louveaux, 1993), in which the uncertainty set is modeled by a Mixed-Integer Program (MIP) via a big-M formulation. We continue this line of research and make the following contributions:

- We tackle a more general problem setting, which considers multiple disruptions with different number of ripples as well as two alternative cost functions to account for the interactions between the disruptions.

- We show that for the linear model, the problem of finding an optimal set of disruptions (worst-case disruptions) for a given flow plan can be solved to optimality by sequentially solving a series of problems considering a single disruption, resulting in a polynomial-time exact algorithm.

- We show that for the maximum model, the problem of finding an optimal set of disruptions (worst-case disruptions) for a given flow plan is NP-hard and, in contrast to the linear case, cannot be decomposed in terms of single disruptions.

- We propose an alternative representation of the uncertainty set using binary variables that does not include any big-M constants. We adapt it to solve the problem of finding an optimal set of disruptions (worst-case disruptions) for a given flow plan considering both the linear and maximum models.

- We connect our findings with the facility location literature and show that our binary formulation solves a planar case of the generalized *maximal covering location problem*. Previous works from the literature do not solve the planar case directly and instead solve these problems by using more challenging formulations built for more general discrete location problems.

- We extend both the linear and the maximum models to account for settings where at most $\Lambda \leq m$ of the disruptions can occur. The parameter Λ adjusts the level of conservativeness, thus less conservative uncertainty sets can be considered. We show that the properties of the basic linear and maximum models extend to this case as well.

- We provide computational experiments on test instances from the literature, which show that embedding our proposed approaches into a cutting-plane algorithm for RMCFP-RED leads to considerable reductions in running times compared to using previous big-M formulations from the literature.

The remainder of this chapter is organized as follows. Section 3.2 describes the cutting-plane framework used for solving RMCFP-RED to optimality. Section 3.3 presents our proposed approach for the linear model while Section 3.4 discusses the case for the maximum model. Section 3.5 introduces formulations for the adversary problems in which the level of uncertainty can be adjusted by the decision maker. Section 3.6 presents our computational experiments. Finally, Section 3.7 concludes this chapter.

3.2 A solution approach for RMCFP-RED

We present a general solution framework for RMCFP-RED. Section 3.2.1 presents a cutting-plane method that solves both the linear and maximum versions of the problem, while Sections 3.2.2 and 3.2.3 present our MIP representations for the uncertainty sets of both models.

3.2.1 A cutting-plane algorithm for the robust problem

A standard approach to solve RMCFP-RED (Mutapcic and Boyd, 2009; Borrero and Lozano, 2020) is a cutting-plane algorithm that reformulates the min-max problem (3.1) as the

following semi-infinite linear programming problem:

$$z^* = \min\{w\colon w - c^\top y \geq 0 \ \forall c \in \mathcal{U}, \boldsymbol{B}y = \boldsymbol{b}, y \leq \boldsymbol{f}, y \in \mathbb{R}_+^{|A|}, w \in \mathbb{R}\}. \tag{3.2}$$

Formulation (3.2) has (potentially) infinitely many constraints, which prevents directly using an LP solver. Nevertheless, it suggests a constraint generation approach as the one shown in Algorithm 1, which iteratively explores realizations of the uncertainty to refine upper and lower bounds on z^*.

Algorithm 1: A cutting-plane algorithm for RMCFP-RED

Data: Set $\mathcal{U}$, matrix $\boldsymbol{B}$, vectors $\boldsymbol{b}$ and $\boldsymbol{f}$

Result: y^ℓ, w^ℓ, and c^ℓ

1 Set $\ell = 0$, define set $\mathcal{U}^0 = \{c^0\}$, where $c^0 \in \mathcal{U}$, $w^0 = -\infty$, $v^0 = \infty$;

2 **while** $v^\ell - w^\ell > 0$ **do**

3 Set $\ell = \ell + 1$;

4 Solve the LP $w^\ell = \min\left\{w : w - c^\top y \geq 0 \ \forall c \in \mathcal{U}^{\ell-1}, \boldsymbol{B}y = \boldsymbol{b}, y \leq \boldsymbol{f}, y \in \mathbb{R}_+^{|A|}, w \in \mathbb{R}\right\}$ and let (w^ℓ, y^ℓ) be an optimal solution;

5 Solve the MIP $v^\ell = \max\{c^\top y^\ell\colon c \in \mathcal{U}\}$ and let c^ℓ be an optimal solution;

6 Update $\mathcal{U}^\ell = \mathcal{U}^{\ell-1} \cup \{c^\ell\}$;

At iteration ℓ, the algorithm solves a relaxation of problem (3.2) using a subset $\mathcal{U}^{\ell-1} \subseteq \mathcal{U}$ instead of $\mathcal{U}$, to obtain a lower bound on z^* given by w^ℓ. Denote by y^ℓ the optimal flow plan obtained from the relaxation of (3.2). Then, the algorithm proceeds to solve the so-called pessimization problem, which is a MIP given by $v^\ell = \max\{c^\top y^\ell\colon c \in \mathcal{U}\}$, establishing an upper bound on z^* given by v^ℓ. If $v^\ell - w^\ell > 0$, then the algorithm continues for another iteration by considering an additional realization of the uncertainty (cost vector) in set $\mathcal{U}^\ell$. Otherwise, the algorithm terminates with an optimal solution of (3.1) given by y^ℓ. Since the set conv.hull($\mathcal{U}$) is a bounded polyhedron (because $\mathcal{U}$ is a finite set of uncertain scenarios,

see Sections 3.2.2 and 3.2.3), Algorithm 1 converges in a finite number of iterations, see Proposition 1.

Proposition 1. *If $\mathcal{U}$ is bounded then Algorithm 1 converges in a finite number of iterations.*

Proof. Consider iteration ℓ of the algorithm and recall that the optimal solution of v^ℓ is denoted by $c^{\ell+1}$. For any $c^j \in \mathcal{U}^{\ell+1}$ let $\hat{c}_i^j$, $i = 1, \ldots, m^j$ be the extreme points of conv.hull($\mathcal{U}$) that appear with a positive coefficient in the convex decomposition of c^j, that is

$$c^j = \sum_{i=1}^{m^j} \gamma_i^j \hat{c}_i^j,$$

where $\sum_{i=1}^{m^j} \gamma_i^j = 1$ and $\gamma_i^j > 0$ for all $i \in [m^j]$. Let $E^j = \{\hat{c}_i^j \colon i \in [m^j]\}$.

We claim that if $E^{\ell+1} = E^j$ for some $j \in [\ell]$ then $v^\ell = w^\ell$. Indeed, note that the optimality of $c^{\ell+1}$ in v^ℓ and the fact that $v^\ell = \max\{c^\top y^\ell \colon c \in \mathcal{U}\} = \max\{c^\top y^\ell \colon c \in \text{conv.hull}(\mathcal{U})\}$ imply that $(\hat{c}_i^{\ell+1})^\top y^\ell = v^\ell$ for all $i \in [m^{\ell+1}]$. Therefore, $(c^j)^\top y^\ell = \sum_{i\in[m^j]} \gamma_i^j (\hat{c}_i^j)^\top y^\ell = \sum_{i\in[m^\ell]} \gamma_i^j (\hat{c}_i^\ell)^\top y^\ell = v^\ell \sum_{i\in[m^j]} \gamma_i^j = v^\ell$ because $\sum_{i\in[m^j]} \gamma_i^j = 1$. Now, by feasibility in the master relaxation problem we have that $w^\ell \geq c^\top y^\ell$ for all $c \in \mathcal{U}^\ell$ and thus $w^\ell \geq (c^j)^\top y^\ell$, that is, $w^\ell \geq v^\ell$ and we can conclude that $v^\ell = w^\ell$, as desired.

The above claim implies that if at iteration ℓ there is no convergence then $E^{\ell+1} \neq E^j$ for all $j \in [\ell]$. Since $2^{ext(\text{conv.hull}(\mathcal{U}))}$ (the set of all subsets of extreme points of conv.hull($\mathcal{U}$)) is a finite set, then it must be the case that the algorithm satisfies that $v^\ell = w^\ell$ in finite iterations, as desired. □

We remark that the only difference between solving the linear and the maximum versions of the problems with the cutting-plane algorithm, relates to the definition of the uncertainty set. We present MIP representations for both uncertainty sets in the subsections below.

3.2.2 Mixed-integer formulation for the uncertainty set in the linear model

For any flow plan $y \in \mathbb{R}_+^{|A|}$, let $v_L(y)$ be the worst-case total cost corresponding to y after the disruptions happen under the linear model, that is

$$v_L(y) = \max\{c^\top y \colon c \in \mathcal{U}_L\}, \tag{3.3}$$

where $\mathcal{U}_L$ denotes the set of possible cost vectors after the disruptions happen in the linear model. In order to define MIP formulation of $v_L(y)$, let $\alpha_j^{(1)}$ and $\alpha_j^{(2)}$ be continuous decision variables representing the x- and y-coordinates of the epicenter of disruption $j \in [m]$, respectively. For any arc $a \in A$, disruption $j \in [m]$, and ripple $k \in [r_j]$, let λ_{ajk} be a binary decision variable defined as

$$\lambda_{ajk} = \begin{cases} 1 \text{ if arc } a \text{ belongs to ripple } R_{jk} \\ 0 \text{ otherwise.} \end{cases} \tag{3.4}$$

We assume (without loss of generality) that the epicenters of the disruptions are located within a sufficiently large rectangle $[L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}]$, in which the network is contained. We define $D(p_a, \alpha_j)$ as the distance (measured by any ℓ^p-norm) between the midpoint of an arc $a \in A$, given by $p_a = (p_a^{(1)}, p_a^{(2)})$, and the epicenter of disruption j, denoted by $\alpha_j = (\alpha_j^{(1)}, \alpha_j^{(2)})$.

We formulate problem (3.3) as the following MIP:

$$
\begin{aligned}
v_L(y) = \max\ & c^\top y && && (3.5a)\\
\text{s.t. } & c_a = c_a^0 + \sum_{j\in[m]} \sum_{k\in[r_j]} d_{jk}\lambda_{ajk} && \forall a \in A && (3.5b)\\
& \sum_{k\in[r_j]} \lambda_{ajk} \le 1 && \forall a \in A,\ j \in [m] && (3.5c)\\
& D(p_a, \alpha_j) - M_a(1-\lambda_{ajk}) \le q_{jk} && \forall a \in A,\ j \in [m], k \in [r_j] && (3.5d)\\
& L^{(i)} \le \alpha_j^{(i)} \le U^{(i)} && \forall j \in [m], i = 1,2 && (3.5e)\\
& \alpha_j \in \mathbb{R}^2 && \forall j \in [m] && (3.5f)\\
& \lambda_{ajk} \in \{0,1\} && \forall a \in A,\ j \in [m], k \in [r_j] && (3.5g)\\
& c_a \ge 0 && \forall a \in A. && (3.5h)
\end{aligned}
$$

Constraints (3.5b) define the cost of each arc $a \in A$ after the disruptions, which is equal to the initial cost c_a^0 increased by the cumulative amount of damages generated by all the disruptions. Constraints (3.5c) ensure that arcs are assigned to at most one ripple from each disruption. Constraints (3.5d) define the λ-variables by ensuring that if $\lambda_{ajk} = 1$, then $D(p_a, \alpha_j) \le q_{jk}$, where M_a is a sufficiently large value that makes the constraint trivially satisfied if $\lambda_{ajk} = 0$. We note that tight values for these big-M constants can be computed efficiently in our problem (for instance, M_a can be set to be the largest possible distance between p_a and any point in $[L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}]$).

The algebraic representation of constraints (3.5d) depends on the specific distance function used. For the ℓ^1-norm, ripples have a diamond-shaped geometry as shown in Figure 3.1. In this case, constraints (3.5d) consider the "Manhattan" distance from arcs to epicenters given by:

$$
|p_a^{(1)} - \alpha_j^{(1)}| + |p_a^{(2)} - \alpha_j^{(2)}| - M_a(1-\lambda_{ajk}) \le q_{jk} \qquad \forall a \in A,\ k \in [r_j], j \in [m]. \qquad (3.6)
$$

We linearize (3.6) by adding auxiliary decision variables $\beta^+_{1aj}, \beta^-_{1aj}, \beta^+_{2aj}, \beta^-_{2aj} \in \mathbb{R}_+$ and formulate constraints (3.5d) under the ℓ^1-norm as:

$$p^{(1)}_a - \alpha^{(1)}_j = \beta^+_{1aj} - \beta^-_{1aj} \qquad \forall a \in A,\ j \in [m] \tag{3.7a}$$

$$p^{(2)}_a - \alpha^{(2)}_j = \beta^+_{2aj} - \beta^-_{2aj} \qquad \forall a \in A,\ j \in [m] \tag{3.7b}$$

$$\beta^+_{1aj} + \beta^-_{1aj} + \beta^+_{2aj} + \beta^-_{2aj} - M_a(1 - \lambda_{ajk}) \leq q_{jk} \qquad \forall a \in A, k \in [r_j], j \in [m] \tag{3.7c}$$

$$\beta^+_{1aj}, \beta^-_{1aj}, \beta^+_{2aj}, \beta^-_{2aj} \geq 0 \qquad \forall a \in A,\ j \in [m], \tag{3.7d}$$

noting that optimization ensures that in an optimal solution $\beta^+_{1aj} + \beta^-_{1aj} + \beta^+_{2aj} + \beta^-_{2aj} = |p^{(1)}_a - \alpha^{(1)}_j| + |p^{(2)}_a - \alpha^{(2)}_j|$. Similar formulations for the ℓ^∞ and ℓ^2 norms are in Appendix 6.1.

3.2.3 Mixed-integer formulation for the uncertainty set in the maximum model

Similar to the linear model, we define for any flow plan $y \in \mathbb{R}^{|A|}_+$ its corresponding worst-case cost $v_M(y)$ as

$$v_M(y) = \max\{c^\top y \colon c \in \mathcal{U}_M\} \tag{3.8}$$

where $\mathcal{U}_M$ denotes the set of possible cost vectors after the realization of the disruptions in the maximum model. Defining the same decision variables as before, $v_M(y)$ is given by:

$$
\begin{aligned}
v_M(y) = \max\ & c^\top y && && \text{(3.9a)}\\
\text{s.t. } & c_a = c_a^0 + \sum_{j\in[m]} \sum_{k\in[r_j]} d_{jk}\lambda_{ajk} && \forall a \in A && \text{(3.9b)}\\
& \sum_{j\in[m]} \sum_{k\in[r_j]} \lambda_{ajk} \leq 1 && \forall a \in A && \text{(3.9c)}\\
& D(p_a, \alpha_j) - M_a(1-\lambda_{ajk}) \leq q_{jk} && \forall a \in A,\ j \in [m], k \in [r_j] && \text{(3.9d)}\\
& L^{(i)} \leq \alpha_j^{(i)} \leq U^{(i)} && \forall j \in [m], i = 1,2 && \text{(3.9e)}\\
& \alpha_j \in \mathbb{R}^2 && \forall j \in [m] && \text{(3.9f)}\\
& \lambda_{ajk} \in \{0,1\} && \forall a \in A,\ j \in [m], k \in [r_j] && \text{(3.9g)}\\
& c_a \in \mathbb{R} && \forall a \in A. && \text{(3.9h)}
\end{aligned}
$$

Constraints (3.9c) now require that for each arc at most one disruption can be active. As a result, optimization ensures that at any optimal solution $\sum_{j\in[m]} \sum_{k\in[r_j]} d_{jk}\lambda_{ajk} = \max_{(j,k):\, a\in R_{jk}}\{d_{jk}\}$ in constraints (3.9b), as desired.

Interestingly, even though the formulations for $v_L(y)$ and $v_M(y)$ are almost identical (the only difference being constraints (3.5c) and (3.9c)), the feasible region of $v_L(y)$ decomposes while the feasible region of $v_M(y)$ does not, as shown in the coming sections. This fact has important consequences on the computational complexity of both problems and on the speed at which both problems can be solved.

3.3 Solution of the MIP $v_L(y)$ for the linear model

Algorithm 1 requires solving a challenging MIP (at line 5) in each iteration. We show that this issue can be alleviated by decomposing the *extended* MIP into m different *single disruption*

problems, which can be solved in polynomial time. As a result, the decomposition implies that $v_L(y)$ is polynomially solvable. Moreover, we show that under additional homogeneity conditions on the parameters of the disruptions, the multiple disruption problem can be further reduced to a single disruption problem with aggregated costs. We note that the results of this section do not depend on the distance metric and therefore we use the general formulation (3.5) throughout rather than the formulation based on the ℓ^1-norm.

3.3.1 A decomposition for the formulation of $v_L(y)$

For a flow vector $y \in \mathbb{R}_+^{|A|}$, consider the problem of finding the worst-case j^{th} disruption, for a fixed sequence of disruptions $u \in [j-1]$, represented by their epicenters, $\bar{\alpha}^u$, and their λ-values, $\bar{\lambda}^u_{ak}$. We formulate this problem as the following MIP (from here on we use both notations λ^j_{ak} and λ_{ajk} to refer to the binary λ-variables interchangeably depending on the context for clarity):

$$
\begin{aligned}
v_L^{(j)}(y) = \ & \max \ c^\top y && && \text{(3.10a)}\\
& \text{s.t. } c_a = c_a^0 + \sum_{u\in[j-1]} \sum_{k\in[r_u]} d_{uk}\bar{\lambda}^u_{ak} + \sum_{k\in[r_j]} d_{jk}\lambda^j_{ak} && \forall a \in A && \text{(3.10b)}\\
& \sum_{k\in[r_j]} \lambda^j_{ak} \le 1 && \forall a \in A && \text{(3.10c)}\\
& D(p_a, \alpha^j) - M_a(1-\lambda^j_{ak}) \le q_{jk} && \forall a \in A, k \in [r_j] && \text{(3.10d)}\\
& L^{(i)} \le \alpha_j^{(i)} \le U^{(i)} && i = 1,2 && \text{(3.10e)}\\
& \lambda^j_{ak} \in \{0,1\} && \forall a \in A, k \in [r_j] && \text{(3.10f)}\\
& c_a \in \mathbb{R} && \forall a \in A. && \text{(3.10g)}
\end{aligned}
$$

Intuitively, if constraints (3.5b) are removed and substituted in the objective function, then $v_L(y)$ can be solved by sequentially solving $v_L^{(1)}(y)$, ..., $v_L^{(m)}(y)$, see Proposition 2.

Proposition 2. *For any given number of disruptions $m \geq 1$ and a flow $y \in \mathbb{R}_+^{|A|}$, it follows that $v_L(y) = v_L^{(m)}(y)$, where $v_L^{(m)}(y)$ is obtained by sequentially solving $v_L^{(1)}(y)$, $v_L^{(2)}(y)$, ..., $v_L^{(m)}(y)$ and fixing the disruptions found at each step.*

Proof. Let $\mathcal{L}$ denote the set of λ-feasible solutions to (3.5) without including constraint (3.5b), that is,

$$\mathcal{L} = \Bigg\{ \lambda \in \Lambda : \exists \alpha_j \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}] \ \forall j \in [m] \text{ s.t. } \sum_{k \in [r_j]} \lambda_{ajk} \leq 1, \ \forall a \in A, j \in [m]; \\ D(p_a, \alpha_j) - M_a(1 - \lambda_{ajk}) \leq q_{jk}, \ \forall a \in A, k \in [r_j], j \in [m] \Bigg\}, \quad (3.11)$$

where $\Lambda = \{0,1\}^{|A| \times \sum_{j \in [m]} r_j}$ and similarly for $j \in [m]$, let $\mathcal{L}_j$ denote the set of λ^j-feasible solutions to (3.10) without including constraint (3.10b), that is

$$\mathcal{L}_j = \Bigg\{ \lambda^j \in \{0,1\}^{|A| \times r_j} : \exists \alpha_j \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}] \text{ s.t.} \\ \sum_{k \in [r_j]} \lambda_{ak}^j \leq 1 \ \forall a \in A; \ D(p_a, \alpha^j) - M_a(1 - \lambda_{ak}^j) \leq q_{jk} \ \forall a \in A, k \in [r_j] \Bigg\}. \quad (3.12)$$

Note that $\mathcal{L} = \mathcal{L}_1 \times \cdots \times \mathcal{L}_m$. In order to prove Proposition 2 we now show that $v_L^{(m)}(y) \leq v_L(y)$ and $v_L^{(m)}(y) \geq v_L(y)$:

(1) $v_L^{(m)}(y) \leq v_L(y)$: Consider the optimal value of $v_L^{(m)}(y)$ given by $\sum_{a \in A} \bar{c}_a y_a$ where

$$\bar{c}_a = c_a^0 + \sum_{u \in [m-1]} \sum_{k \in [r_u]} d_{uk} \bar{\lambda}_{ak}^u + \sum_{k \in [r_m]} d_{mk} \bar{\lambda}_{ak}^m \qquad \forall a \in A. \quad (3.13)$$

Because $\bar{\lambda}^u \in \mathcal{L}_u$ for all $u \in [m]$ and $\mathcal{L} = \mathcal{L}_1 \times \cdots \times \mathcal{L}_m$, it follows that $(\bar{c}, \bar{\alpha}, \bar{\lambda})$ is a feasible solution to problem (3.5), where $\bar{c}$ is given by (3.13), $\bar{\alpha} = (\bar{\alpha}^u : u \in [m])$, and $\bar{\lambda} = (\bar{\lambda}_{ak}^u : a \in A, u \in [m], k \in [r_u])$. Hence, $v_L^{(m)}(y) \leq v_L(y)$ as desired.

(2) $v_L^{(m)}(y) \geq v_L(y)$: Let $\lambda^* = (\lambda_1^*, \ldots, \lambda_m^*) \in \mathcal{L}$ be a λ-optimal solution associated with $v_L(y)$, where $\lambda_j^* = (\lambda_{ajk}^* \colon a \in A, k \in [r_j])$ for any $j \in [m]$. Because $\lambda_j^* \in \mathcal{L}_j$ for any $j \in [m]$, from the definition of $\bar{\lambda}^j$ it follows that:

$$\sum_{a \in A} \sum_{k \in [r_j]} d_{jk} \lambda_{ajk}^* y_a \leq \sum_{a \in A} \sum_{k \in [r_j]} d_{jk} \bar{\lambda}_{ak}^j y_a \qquad \forall j \in [m]. \tag{3.14}$$

By using (3.14) and adding across all $j \in [m]$ we conclude that

$$v_L(y) = \sum_{a \in A} \left(c_a^0 + \sum_{j \in [m]} \sum_{k \in [r_j]} d_{jk} \lambda_{ajk}^* \right) y_a \leq \sum_{a \in A} \left(c_a^0 + \sum_{j \in [m]} \sum_{k \in [r_j]} d_{jk} \bar{\lambda}_{ak}^j \right) y_a = v_L^{(m)}(y), \tag{3.15}$$

as desired. □

We now consider the case where all m disruptions are *homogeneous*. That is, all disruptions have the same number of ripples $r_j = r$, the same values for the distances $q_{jk} = q_k$ defining the ripples, and the same values for the damages $d_{jk} = d_k$ for all $j \in [m]$. In this case, the set of feasible solutions of λ_{ajk} for all disruptions j are the same. For any $y \in \mathbb{R}_+^{|A|}$ consider the following MIP:

$$v_H(y) = \max \; c^\top y \tag{3.16a}$$

$$\text{s.t. } c_a = c_a^0 + m \sum_{k \in [r]} d_k \lambda_{ak} \qquad \forall a \in A \tag{3.16b}$$

$$\sum_{k \in [r]} \lambda_{ak} \leq 1 \qquad \forall a \in A \tag{3.16c}$$

$$D(p_a, \alpha) - M_a(1 - \lambda_{ak}) \leq q_k \qquad \forall a \in A, k \in [r] \tag{3.16d}$$

$$L^{(i)} \leq \alpha^{(i)} \leq U^{(i)} \qquad i = 1, 2 \tag{3.16e}$$

$$\lambda_{ak} \in \{0, 1\} \qquad \forall a \in A, k \in [r] \tag{3.16f}$$

$$c_a \in \mathbb{R} \qquad \forall a \in A. \tag{3.16g}$$

Observe that (3.16) is the MIP of a single disruption problem with r ripples, damages given by md_k, and distances given by q_k, $k \in [r]$. A direct consequence of the decomposition approach discussed in Proposition 2 is that $v_L(y) = v_H(y)$, see Proposition 3.

Proposition 3. *Let a flow $y \in \mathbb{R}_+^{|A|}$ be given. If the disruptions are homogeneous then $v_L(y) = v_H(y)$.*

Proof. Suppose that $\lambda^H \in \mathcal{L}_H$ is the λ-optimal solution of $v_H(y)$ and let $\lambda^* = (\lambda_1^*, \ldots, \lambda_m^*) \in (\mathcal{L}_H)^m$ be the λ-optimal solution of $v_L(y)$. Define $\bar{\lambda}$ as m copies of λ^H one after the other, $\bar{\lambda} = (\lambda^H, \ldots, \lambda^H)$ and observe that $\bar{\lambda} \in (\mathcal{L}_H)^m$. From the optimality and feasibility of λ^* and $\bar{\lambda}$ in $v_L(y)$, respectively, it follows that

$$v_L(y) = \sum_{a \in A} \left(c_a^0 + \sum_{k \in [r]} d_k \sum_{j \in [m]} \lambda_{ajk}^* \right) y_a \geq \sum_{a \in A} \left(c_a^0 + m \sum_{k \in [r]} d_k \lambda_{ak}^H \right) y_a = v_H(y). \tag{3.17}$$

On the other hand, from the optimality of λ^H and feasibility of λ_j^*, $j \in [m]$ in $v_H(y)$, it follows that

$$v_H(y) = \sum_{a \in A} \left(c_a^0 + m \sum_{k \in [r]} d_k \lambda_{ak}^H \right) y_a \geq \sum_{a \in A} \left(c_a^0 + m \sum_{k \in [r]} d_k \lambda_{ajk}^* \right) y_a \qquad \forall j \in [m]. \tag{3.18}$$

The aggregation of (3.18) over all $j \in [m]$ gives

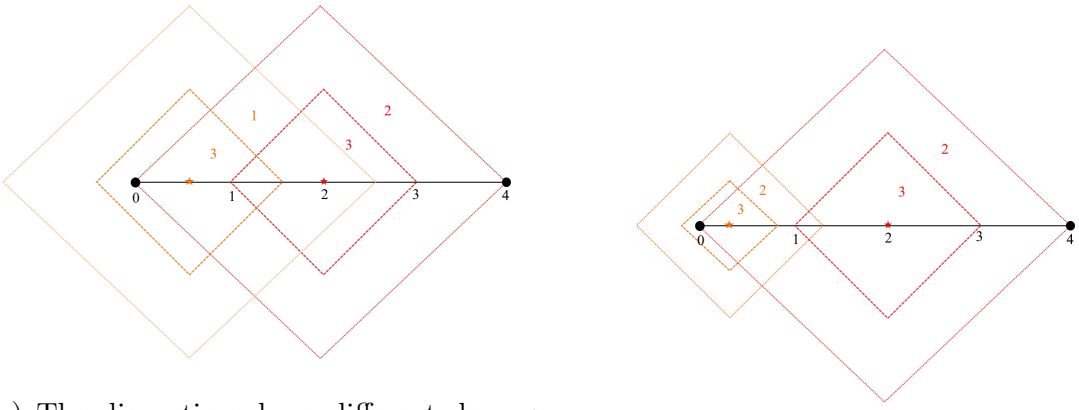
$$\begin{aligned} m \sum_{a \in A} \left(c_a^0 + m \sum_{k \in [r]} d_k \lambda_{ak}^H \right) y_a &\geq \sum_{a \in A} \left(m c_a^0 + m \sum_{j \in [m]} \sum_{k \in [r]} d_k \lambda_{ajk}^* \right) y_a \\ &= m \sum_{a \in A} \left(c_a^0 + \sum_{j \in [m]} \sum_{k \in [r]} d_k \lambda_{ajk}^* \right) y_a. \end{aligned} \tag{3.19}$$

Inequality (3.19) implies that $v_H(y) \geq v_L(y)$ and therefore $v_H(y) = v_L(y)$, as desired. □

Remark 1. *If the disruptions have different ripple radii or yield different damage amounts*

then the locations of their epicenters are not necessarily the same and the multiple disruptions problem cannot be reduced to a single disruption ripple effect problem. To this end, consider first two disruptions with the same number of ripples and the same radii but with different damage values, see Figure 3.2a. Suppose two arcs whose midpoints are located on the coordinates $p_{a_1} = (0, 0)$ *and* $p_{a_2} = (4, 0)$*, assume radii of* $q_{11} = q_{21} = 1$ *and* $q_{12} = q_{22} = 2$*, and different damage values given by* $d_{11} = d_{21} = 3, d_{12} = 2$*, and* $d_{22} = 1$*. Observe that it is necessary for an optimal solution to have the first disruption centered at* $(2, 0)$ *because it covers both arcs with the second ripple and gives a cost increase of 4 units. However, it is not optimal to have the epicenter of the second disruption at* $(2, 0)$ *because having the first ripple of the second disruption covering just one arc yields more damage (3 units) than covering both arcs (2 units).*

Second, consider two disruptions with different radii and the same damage values, see Figure 3.2b. Here $q_{11} = 1, q_{12} = 2$*,* $q_{21} = 0.5, q_{22} = 1$*,* $d_{11} = d_{21} = 3$ *and* $d_{12} = d_{22} = 2$*. As before, in this case it is necessary for an optimal solution to locate the epicenter of the first disruption at* $(2, 0)$*. It is clear, however, that having the epicenter of the second disruption at* $(2, 0)$ *is not optimal because it does not damage any arc.*



(a) The disruptions have different damage values.

(b) The disruptions have different radii.

Figure 3.2: Two problem instances having non-homogeneous disruptions.

3.3.2 A polynomial time algorithm for $v_L(y)$

Proposition 2 shows that for any given flow y, $v_L(y)$ can be decomposed by solving m single disruption problems. We now show that single disruption problems can be solved in polynomial time following arguments used in the literature of maximal covering location problems (Church, 1984; Berman and Krass, 2002; Berman et al., 2010). Let r be the number of ripples for the disruption under consideration and let q_k, $k \in [r]$ be the radii of the ripples. Assume that the distances are measured using the ℓ^1-induced norm (the argument for ℓ^2 and ℓ^∞ norms is similar). For a given arc $a \in A$, let $B_k(a)$ and $Q_k(a)$ be the (solid) diamond and the border of the diamond of center p_a and radius q_k in the plane:

$$B_k(a) = \{x \in \mathbb{R}^2 \colon ||p_a - x||_1 \leq q_k\} \text{ and } Q_k(a) = \{x \in \mathbb{R}^2 \colon ||p_a - x||_1 = q_k\}, \tag{3.20}$$

and for any $a, b \in A$ and $k, \ell \in [r]$ consider the intersection $Q_k(a) \cap Q_\ell(b)$. To compute the elements of $Q_k(a) \cap Q_\ell(b)$, we first determine the intersections of the *entire lines* induced by the diamond's borders. If an intersection point x also lies in $Q_k(a)$ and $Q_\ell(b)$, then x is added to the set of intersections. Besides being empty, there are only three possibilities for the number of elements of this intersection: it can consist of only one point if $Q_k(a)$ and $Q_\ell(b)$ meet at a corner; it can consist of infinitely many points if $Q_k(a)$ and $Q_\ell(b)$ overlap at one of the sides; or else, it consists of two points. Define $I_{ab}^{k\ell}$ as

$$I_{ab}^{k\ell} = \begin{cases} Q_k(a) \cap Q_\ell(b), & \text{if } |Q_k(a) \cap Q_\ell(b)| \leq 2 \\ \text{an arbitrary } x \in Q_k(a) \cap Q_\ell(b), & \text{if } |Q_k(a) \cap Q_\ell(b)| = \infty, \end{cases} \tag{3.21}$$

and let I be the set of all intersection points

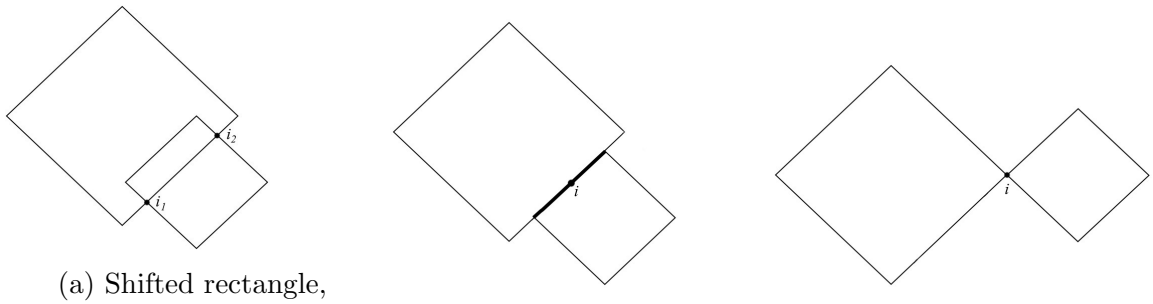
$$I = \bigcup_{k,\ell \in [r], a,b \in A} I_{ab}^{k\ell}. \tag{3.22}$$

In total, there are $O(|A|^2 r^2)$ pairs of diamonds that must be evaluated for their intersections. It is worth mentioning that the case for the ℓ^∞ norm has the same number of intersection points as the ℓ^1 norm, while for the ℓ^2 norm there are at most two intersection points. Notice that for the ℓ^2 norm, finding intersection points of the 'circle versions' of $Q_k(a)$ and $Q_\ell(b)$ simplifies to solving a quadratic equation.

Finally, define $P = \{x \in \mathbb{R}^2 \colon x = p_a \text{ for some } a \in A\}$, hence P is the set of locations of the arcs of the network. We have Proposition 4 that mentions the optimal epicenter can be found in $(I \cup P)$. In order to prove Proposition 4, we first have the following lemma.

Lemma 1. *Let $A = \{1, 2, \ldots, n\}$ and define $B_i = \{x \in \mathbb{R}^2 \colon ||p_i - x||_1 \leq v_i\}$ where $p_i \in \mathbb{R}^2$ and $v_i > 0$ for each $i \in A$ and $Q_i = \{x \in \mathbb{R}^2 \colon ||p_i - x||_1 = v_i\}$. For any $i, j \in A$ define I_{ij} in an analogous way to Equation* (3.21) *and let $I = \bigcup_{i,j \in A} I_{ij}$. Define $\mathcal{B} = \bigcap_{i \in A} B_i$. If $\mathcal{B} \neq \emptyset$ and $\mathcal{Q} = \bigcap_{i \in A} Q_i \neq \emptyset$ then: (*i*) $\mathcal{B}$ is either a solid rectangle shifted 45 degrees; a line segment; a set of two points; or a single point. If it is a rectangle, then each side is a subset of a side of a Q_k for some $k \in A$; if it is a line segment then $\mathcal{B}$ is a subset of a side of a Q_k for some $k \in A$, if $\mathcal{B}$ has two or one points, then these points belong to some Q_k, $k \in A$. (*ii*) $I \cap \mathcal{B}$ is non-empty.*

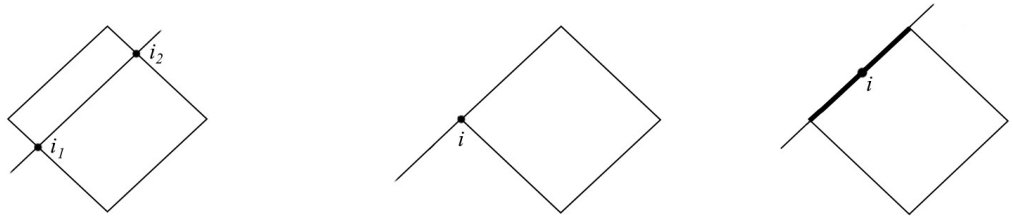
Proof. The proof is by induction on the number of elements of A. For the base case suppose $n = 2$. Consider the possible scenarios for the intersection $B_1 \cap B_2$ assuming that it is non-empty, see Figure 3.3.



(a) Shifted rectangle, $I_{12} = \{i_1, i_2\}$ (b) Line segment, $I_{12} = \{i\}$ (c) Single point, $I_{12} = \{i\}$

Figure 3.3: Possible intersection of two diamonds.

Observe that $B_1 \cap B_2$ its either a (shifted) rectangle, whose sides are subsets of the sides of Q_1 and Q_2, a line segment that is a subset of a side of each Q_1 and Q_2, or a single point in I. In either case, note that $I \cap (Q_1 \cap Q_2) \neq \emptyset$.



(a) Two points, $I_{12} = \{i_1, i_2\}$ (b) Single point, $I_{12} = \{i\}$ (c) Line segment, $I_{12} = \{i\}$

Figure 3.4: Possible intersection of diamond with a segment.

For the inductive step, we assume the result holds true for n and show it for $n+1$. Let $\mathcal{B}_n = \bigcap_{i \in [n]} B_i$, then by the inductive hypothesis $\mathcal{B}_n$ satisfies (i) and (ii) of the statement of the lemma. Suppose that $\mathcal{B}_n$ is a solid rectangle. Then $\mathcal{B} = \mathcal{B}_n \cap B_{n+1}$ is analyzed in the same way as in the base case and the result follows. Suppose that $\mathcal{B}_n$ is a line segment. In this case $\mathcal{B}_n \cap B_{n+1}$ consists of either two points, one point, or a line segment that is a subset of $\mathcal{B}_n$, see examples in Figure 3.4. If it is a line segment, then clearly $\mathcal{B}$ is a subset of a side of B_{n+1}. Moreover, note that independent of the case $I \cap \mathcal{B} \neq \emptyset$ because $\mathcal{B}_n$ is a subset of a side of some Q_k. If $\mathcal{B}_n$ consist of either two points or a single point, then clearly $\mathcal{B}$ can be either a set with two points or a set with a single point and its also evident that $I \cap \mathcal{B} \neq \emptyset$. □

Proposition 4. *Assume that $m = 1$, let a flow $y \in \mathbb{R}_+^{|A|}$ be given, and let $\mathcal{E}$ be the set of optimal epicenters of $v_L(y)$. Then $(I \cup P) \cap \mathcal{E} \neq \emptyset$.*

Proof. By the proof of Theorem 1 in Church (1984), it can be assumed that the optimal epicenter of the disruption is in the interior of the non-empty set

$$\widetilde{B} = \bigcap_{k \in [r]} \bigcap_{a \in \tilde{A}_k} B_k(a) \tag{3.23}$$

where $\tilde{A}_k$ is the set of arcs that are between a distance of q_{k-1} and q_k of the epicenter of the

disruption and where $\widetilde{\mathcal{Q}} = \bigcap_{k\in[r]} \bigcap_{a\in\tilde{A}_k} Q_k(a) \neq \emptyset$. Note that the epicenter can be moved to any point of $\widetilde{\mathcal{B}}$ without affecting the value of the disruptions of any of the arcs of the network. Now, observe that $\widetilde{B}$ is a set of the form $\mathcal{B}$ given in Lemma 1. In particular, this Lemma implies that $I \cap \widetilde{B} \neq \emptyset$, where I is given by Equation (3.22). Thus, it can be concluded that the epicenter of the disruption can be moved to a point of I without changing the value of the objective function, which gives the desired result. □

We note that the analogous version of Proposition 4 for the ℓ^∞ norm follows the same arguments of the ℓ^1 case. The proof of the ℓ^2 case follow from mimicking the proof of Theorem 1 of Church (1984).

As a consequence, single disruption problems can be solved by the following *Discrete Location Algorithm* (DLA): (i) find all the $O(|A|^2r^2)$ locations in $I \cup P$ and then (ii) evaluate the damage of the disruption in each of the locations in $I \cup P$; the one that attains the maximum value is an optimal epicenter for the disruption. DLA runs in polynomial time because evaluating the effect of a disruption takes at most $O(|A|r)$ time, which implies that even a naive implementation of DLA takes $O(|A|^3r^3)$ time.

For $m \geq 1$, $v_L(y)$ can be solved in $O(m|A|^3r^3)$ time by following a *Multiple Disruptions Discrete Location Algorithm* (MDDLA). MDDLA loops over all disruptions, $j = 1, \ldots, m$, fixing the c-optimal solution of $v_L^{(j-1)}(y)$ as the initial cost vector in $v_L^{(j)}(y)$ and solving the single disruption problem $v_L^{(j)}(y)$ using the DLA (we make the convention that the c-optimal solution of $v_L^{(0)}(y)$ is c^0). After the loop is done, MDDLA outputs the value of $v_L^{(m)}(y)$, which by Proposition 2, equals the optimal value of $v_L(y)$.

3.4 Solution of the MIP $v_M(y)$ for the maximum model

In contrast to the linear case, we show that decomposing $v_M(y)$ as single disruption problems is not possible in general and that in fact, finding an optimal solution to $v_M(y)$ is NP-hard.

We propose a reformulation for $v_M(y)$ that does not use big-M constants and is theoretically better than formulation (3.9). Moreover, our proposed reformulation also works for the linear model and surprisingly its computational performance is equivalent or better than the performance of the polynomial time algorithm proposed, as we show in the numerical experiments in Section 3.6. Finally, our reformulation also allows us to connect the disruption problems with maximum clique problems in intersection graphs.

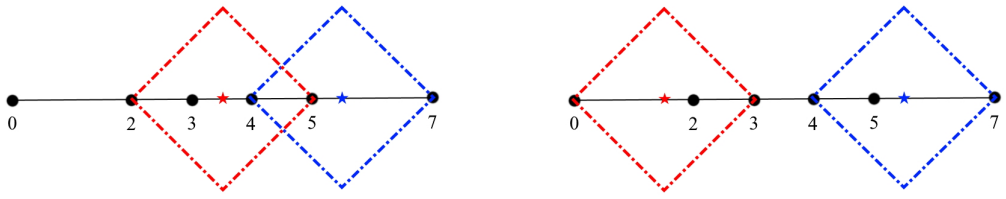
3.4.1 Problem $v_M(y)$ is NP-hard

First, we show with a counter example that problem $v_M(y)$ cannot be decomposed into simpler single disruption problems in general.

Remark 2. *Suppose that the network consists of six arcs with coordinates* $p_{a_1} = (0,0)$, $p_{a_2} = (2,0)$, $p_{a_3} = (3,0)$, $p_{a_4} = (4,0)$, $p_{a_5} = (5,0)$, *and* $p_{a_6} = (7,0)$. *Assume that there are two homogeneous disruptions with a single ripple characterized by* $q_j = 1.5$ *and* $d_j = 1$ *for* $j = 1, 2$. *Decompose* $v_M(y)$ *as two successive disruptions by adapting Proposition 2. Here, instead of using* $\sum_{u\in[j-1]} \sum_{k\in[r_u]} d_{uk}\bar{\lambda}_{auk} + \sum_{k\in[r_j]} d_{jk}\lambda_{ajk}$ *to evaluate the effect generated by disruption* j *on arc* a, *we use the maximum between* $\max\{\sum_{k\in[r_u]} d_{uk}\bar{\lambda}_{auk} \colon u \in [j-1]\}$ *and* $\sum_{k\in[r_j]} d_{jk}\lambda_{ajk}$. *This decomposition results in locations for the disruptions given by* $\alpha_1 = (3.5, 0)$ *and* $\alpha_2 = (5.5, 0)$ *with an objective value of* $v = \sum_{i\in[6]} c^0_{a_i} + 5$, *see Figure 3.5a. Observe that this is not an optimal solution for* $v_M(y)$ *because locating one disruption at* $\alpha_1 = (1.5, 0)$ *and another one at* $\alpha_2 = (5.5, 0)$ *results in an objective value of* $\sum_{i\in[6]} c^0_{a_i} + 6$, *see Figure 3.5b.*

We prove that $v_M(y)$ is NP-hard by a reduction from the BOX COVER problem stated as follows.

BOX COVER problem: We are given a collection of $m \geq 1$ squares (or boxes) in $\mathbb{R}^2$ of identical size and a finite set of points S in $\mathbb{R}^2$. All point coordinates and the length of all box



(a) Decomposition solution (b) Optimal solution

Figure 3.5: Counter example that illustrates that problem $v_M(y)$ cannot be decomposed into sequential single-disruptions problems.

sides are assumed to be integer. The answer of the problem is YES if and only if a subset of the squares, placed with their sides parallel to the axis of the plane, can cover all the points in S.

The BOX COVER problem is NP-complete and the disc version, where instead of boxes one has m identical circles, is also NP-complete, see Theorems 1 and 4 in Fowler et al. (1981).

Proposition 5. *Problem $v_M(y)$ is NP-hard under either the ℓ^1, ℓ^2, or the ℓ^∞ induced norm.*

Proof. We consider an arbitrary instance of BOX COVER with locations $S \neq \emptyset$ and $m > 1$ identical boxes of side $q > 0$. We construct a network G and a flow y such that the answer of BOX COVER is yes if and only if $v_M(y) \geq |S|$. Create a network $G = (V, E)$ with a single path as follows. Let (x_i, y_i) be the integer-valued coordinates of point $i \in S$, and assume without loss of generality that $x_i < x_{i+1}$; if $x_i = x_{i+1}$, then assume $y_i \leq y_{i+1}$. Consider a graph $G = (V, E)$ with $V = \{1, \ldots, |S|+1\}$, E initially empty, and denote the coordinates of vertex i as $(p_i^{(1)}, p_i^{(2)})$. Suppose first that $x_1 < x_2$ (the other case is similar). Then, define $(p_1^{(1)}, p_1^{(2)}) = (x_1, y_1 - 1)$ and $(p_2^{(1)}, p_2^{(2)}) = (x_1, y_1 + 1)$ and add edge (1,2) into E. Note that the midpoint of (1,2) is precisely (x_1, y_1). Now, for $i = 2, \ldots, |S|$, let $\delta_i^{(1)} = x_i - p_i^{(1)}$ and $\delta_i^{(2)} = y_i - p_i^{(2)}$ and define $p_{i+1}^{(1)} = x_i + \delta_i^{(1)}$ and $p_{i+1}^{(2)} = y_i + \delta_i^{(2)}$. Add $(i, i+1)$ to E and note that (x_i, y_i) is the midpoint of edge $(i, i+1)$. In this way, graph G consists of a single path with source at node 1 and sink at node $|S| + 1$ whose arcs' midpoints are the points in S, see Figure 3.6. Now, assume that the minimum cost flow problem is the shortest path problem where the source is 1 and the sink is

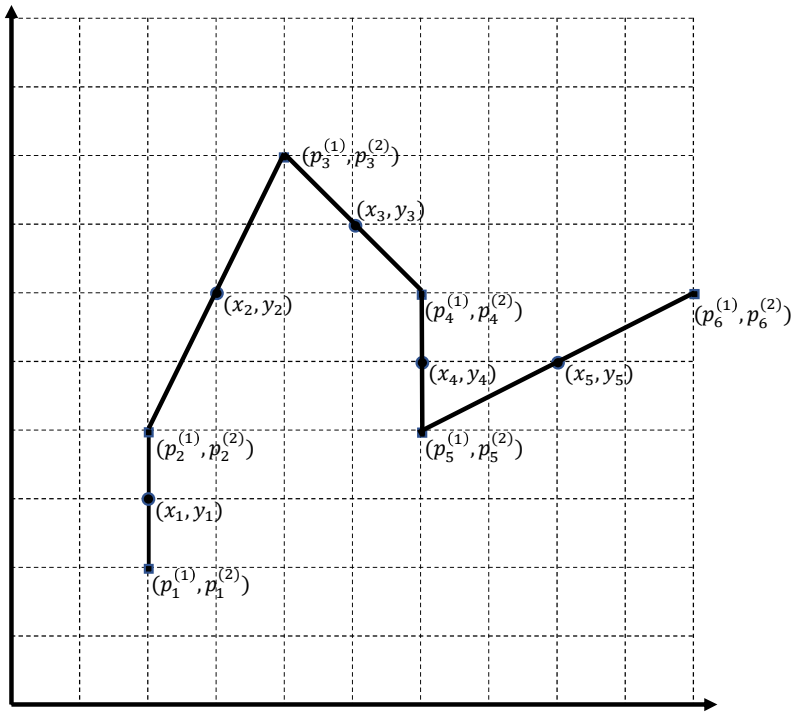


Figure 3.6: Example of the graph construction with $|S| = 5$. The black lines are the graph's edges, the intersections of the lines of the grid represent integer coordinates.

$|S| + 1$ (thus $y_a = 1$ for all $a \in A$), and let m be the number of disruptions. Let $c_a^0 = 0$ for all arcs $a \in A$, assume only one ripple $r_j = 1$ for all disruptions $j \in [m]$, set $d_j = 1$, and $q_j = q/2$.

For the ℓ^∞ norm, the optimal value of $v_M(y)$ on this instance is less than $|S|$ if and only if the solution of BOX COVER is NO. Otherwise, if $v_M(y) \geq |S|$ then the answer to BOX COVER is YES. As a result, problem $v_M(y)$ is NP-hard under the ℓ^∞ norm.

The proof for the ℓ^1 norm follows by rotating everything 45 degrees and changing q_j accordingly. The proof for the ℓ^2 norm is a consequence of the same arguments but assuming discs instead of boxes. □

3.4.2 An improved reformulation for $v_M(y)$

It is well-known that big-M formulations such as (3.9) can lead to weak LP relaxations which, in turn, impair the performance of the branch-and-cut methods that are typically employed

to solve MIPs. We introduce a set of valid constraints for $v_M(y)$ and propose a reformulation that does not rely on big-M coefficients. We show that our reformulation is stronger than the big-M formulation (3.9) under suitable assumptions on the big-M coefficients and remark the relationship between our reformulation and maximum clique problems in *intersection graphs*.

As before, for any disruption $j \in [m]$ define vector $\lambda^j = (\lambda_{ajk} \colon a \in A, k \in [r_j])$ and binary set $\mathcal{M}_j$ as

$$\mathcal{M}_j = \Big\{ \lambda^j \in \{0,1\}^{|A| \times r_j} \colon \exists \alpha_j \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}] \text{ s.t.} \\ D(p_a, \alpha_j) - M_a(1 - \lambda^j_{ak}) \leq q_{jk} \quad \forall a \in A, k \in [r_j] \Big\}. \tag{3.24}$$

Let us denote by $\mathcal{C}$ the set of binary points that satisfy the coupling constraint (3.9c), that is

$$\mathcal{C} = \Big\{ \lambda \in \{0,1\}^{|A| \times \sum_{j \in [m]} r_j} \colon \sum_{j \in [m]} \sum_{k \in [r_j]} \lambda_{ajk} \leq 1 \quad \forall a \in A \Big\}. \tag{3.25}$$

In terms of these new sets, the uncertainty set described by formulation (3.9) can be written as:

$$\mathcal{U} = \Big\{ c \in \mathbb{R}^{|A|}_+ \colon \exists \lambda \in \mathcal{C} \cap \prod_{j \in [m]} \mathcal{M}_j \text{ s.t. } c_a = c^0_a + \sum_{j \in [m]} \sum_{k \in [r_j]} d_{jk} \lambda_{ajk} \quad \forall a \in A \Big\}. \tag{3.26}$$

Our proposed reformulations describe sets $\mathcal{M}_j$ without using big-M constants. The constraints defining these sets are derived from the following lemma, which states that two arcs cannot be in two different ripples of one disruption if the distance between them is sufficiently large.

Lemma 2. *Let disruption $j \in [m]$, ripples $k_1, k_2 \in [r_j]$, and arcs $a, b \in A$ be given. If $D(p_a, p_b) > q_{jk_1} + q_{jk_2}$ then $a \notin R_{jk_1}$ or $b \notin R_{jk_2}$.*

Proof. The proof follows from the contrapositive. We claim that if $a \in R_{jk_1}$ and $b \in R_{jk_2}$

then $D(p_a, p_b) \leq q_{jk_1} + q_{jk_2}$. Indeed, we have that

$$D(p_a, p_b) \leq D(p_a, \alpha_j) + D(p_b, \alpha_j) \leq q_{jk_1} + q_{jk2}, \tag{3.27}$$

where the first inequality follows from the triangle inequality of distance functions and the second inequality because $a \in R_{jk_1}$ and $b \in R_{jk_2}$. □

Following Lemma 2, we define the set of infeasible pairwise combinations of ripples for disruption $j \in [m]$ given any two arcs $a, b \in A$, $a \neq b$, as follows:

$$\Gamma^j_{a,b} = \big\{(k_1, k_2) \in [r_j] \times [r_j] : D(p_a, p_b) > q_{jk_1} + q_{jk_2}\big\}, \tag{3.28}$$

and define the binary set $\mathcal{B}_j$ as

$$\mathcal{B}_j = \Big\{\lambda^j \in \{0,1\}^{|A| \times r_j} \colon \lambda^j_{ak_1} + \lambda^j_{bk_2} \leq 1 \quad \forall\ (k_1, k_2) \in \Gamma^j_{a,b},\ a, b \in A, a \neq b.\Big\}. \tag{3.29}$$

We have that the elements of $\mathcal{M}_j$ satisfy the *conflict constraints* defining $\mathcal{B}_j$. This statement is formalized in the next result.

Proposition 6. *For any given disruption $j \in [m]$ and a distance measure given by an ℓ^p-norm it follows that $\mathcal{M}_j \subseteq \mathcal{B}_j$.*

Proof. Let $\lambda^j \in \mathcal{M}_j$. Note that for any given arcs $a, b \in A$ and ripples $k_1, k_2 \in [r_j]$ it follows that:

$$D(p_a, \alpha_j) - M_a(1 - \lambda^j_{ak_1}) \leq q_{jk_1}, \tag{3.30a}$$

$$D(p_b, \alpha_j) - M_b(1 - \lambda^j_{bk_2}) \leq q_{jk_2}. \tag{3.30b}$$

Adding constraints (3.30a) and (3.30b) gives

$$D(p_a, \alpha_j) + D(p_b, \alpha_j) - M_a(1 - \lambda^j_{ak_1}) - M_b(1 - \lambda^j_{bk_2}) \leq q_{jk_1} + q_{jk_2}. \tag{3.31}$$

By the triangle inequality, we have

$$D(p_a, \alpha_j) + D(p_b, \alpha_j) \geq D(p_a, p_b) \tag{3.32}$$

Let $M = \max\{M_a, M_b\}$, then we obtain the following from (3.31):

$$D(p_a, p_b) - M(2 - \lambda^j_{ak_1} - \lambda^j_{bk_2}) \leq q_{jk_1} + q_{jk_2}. \tag{3.33}$$

Note that if $D(p_a, p_b) > q_{jk_1} + q_{jk_2}$, i.e., $(k_1, k_2) \in \Gamma^j_{a,b}$, then (3.33) holds if and only if

$$\lambda^j_{ak_1} + \lambda^j_{bk_2} \leq 1, \tag{3.34}$$

and therefore constraint $\lambda^j_{ak_1} + \lambda^j_{bk_2} \leq 1$ is satisfied by any pair of arcs $a, b \in A$ over all $(k_1, k_2) \in \Gamma^j_{a,b}$, i.e., $\lambda^j \in \mathcal{B}_j$. □

The converse of Proposition 6 is also valid under the distances induced by the ℓ^1 and ℓ^∞ norms, under the assumption that the coupling constraints in $\mathcal{C}$ are satisfied, as shown next.

Proposition 7. *For a disruption $j \in [m]$ under the distance measure induced by the ℓ^1 norm, if $\lambda = (\lambda^1, \ldots, \lambda^m) \in \mathcal{C}$ and $\lambda^j \in \mathcal{B}_j$, then $\lambda^j \in \mathcal{M}_j$.*

Proof. For a given $a \in A$, if $\lambda^j_{ak} = 0$ for all $k \in [r_j]$, then the constraints in $\mathcal{M}_j$ are trivially satisfied for arc a. Henceforth, we consider the set of arcs $A_j \subseteq A$ such that $\lambda^j_{ak} = 1$ for some $k \in [r_j]$. Since $\lambda \in \mathcal{C}$, then for each arc $a \in A_j$ there is exactly one ripple, denoted by k_a, for which $\lambda^j_{ak_a} = 1$ and we have that $\lambda^j_{ak} = 0$ for all $k \in [r_j], k \neq k_a$.

Constraints in $\mathcal{M}_j$ for arc a and ripple $k \neq k_a$ are trivially satisfied since $\lambda^j_{ak} = 0$. We

show that constraints in $\mathcal{M}_j$ hold for any arc $a \in A_j$ and $k = k_a$ as there exists a location $\alpha_j \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}]$ such that

$$|p_a^{(1)} - \alpha_j^{(1)}| + |p_a^{(2)} - \alpha_j^{(2)}| \leq q_{jk_a} \qquad \forall a \in A_j. \tag{3.35}$$

Observe that (3.35) is equivalent to

$$-q_{jk_a} + (p_a^{(1)} + p_a^{(2)}) \leq \alpha_j^{(1)} + \alpha_j^{(2)} \leq q_{jk_a} + (p_a^{(1)} + p_a^{(2)}) \qquad \forall a \in A_j \tag{3.36a}$$

$$-q_{jk_a} + (p_a^{(1)} - p_a^{(2)}) \leq \alpha_j^{(1)} - \alpha_j^{(2)} \leq q_{jk_a} + (p_a^{(1)} - p_a^{(2)}) \qquad \forall a \in A_j. \tag{3.36b}$$

In turn, (3.36) holds if and only if

$$\max_{a \in A_j}\{-q_{jk_a} + (p_a^{(1)} + p_a^{(2)})\} \leq \min_{a \in A_j}\{q_{jk_a} + (p_a^{(1)} + p_a^{(2)})\} \tag{3.37a}$$

$$\max_{a \in A_j}\{-q_{jk_a} + (p_a^{(1)} - p_a^{(2)})\} \leq \min_{a \in A_j}\{q_{jk_a} + (p_a^{(1)} - p_a^{(2)})\}. \tag{3.37b}$$

To see why the equivalence is true, first observe that constraints (3.36) imply that (3.37) hold. Now, assume that (3.37) hold and denote by M_1 and M_2 the values of the minima in (3.37) and by m_1 and m_2 the value of the maxima, thus (3.37) is equivalently written as $m_1 \leq M_1$ and $m_2 \leq M_2$. Define $\alpha_j^{(1)} = (m_1 + m_2)/2$ and $\alpha_j^{(2)} = (m_1 - m_2)/2$. Then $\alpha_j^{(1)} + \alpha_j^{(2)} = m_1$ and $\alpha_j^{(1)} - \alpha_j^{(2)} = m_2$ and therefore $m_1 \leq \alpha_j^{(1)} + \alpha_j^{(2)} \leq M_1$ and $m_2 \leq \alpha_j^{(1)} - \alpha_j^{(2)} \leq M_2$, as desired.

Let $\max_{a \in A_j}\{-q_{jk_a} + (p_a^{(1)} + p_a^{(2)})\} = -q_{jk_e} + (p_e^{(1)} + p_e^{(2)})$, $\min_{a \in A_j}\{q_{jk_a} + (p_a^{(1)} + p_a^{(2)})\} = q_{jk_f} + (p_f^{(1)} + p_f^{(2)})$, $\max_{a \in A_j}\{-q_{jk_a} + (p_a^{(1)} - p_a^{(2)})\} = -q_{jk_g} + (p_g^{(1)} - p_g^{(2)})$, and $\min_{a \in A_j}\{q_{jk_a} +$

$(p_a^{(1)} - p_a^{(2)})\} = q_{jk_h} + (p_h^{(1)} - p_h^{(2)})$. So, (3.37) holds if and only if the following inequalities hold:

$$(p_e^{(1)} - p_f^{(1)}) + (p_e^{(2)} - p_f^{(2)}) \leq q_{jk_e} + q_{jk_f} \tag{3.38a}$$

$$(p_g^{(1)} - p_h^{(1)}) + (p_h^{(2)} - p_g^{(2)}) \leq q_{jk_g} + q_{jk_h}. \tag{3.38b}$$

For inequality (3.38a) there are two cases to consider: If $e = f$ then the inequality is trivially satisfied. If $e \neq f$, then $(e, f) \notin \Gamma^j_{e,f}$ because $\lambda^j \in \mathcal{B}_j$ and $\lambda^j_{ek_e} = \lambda^j_{fk_f} = 1$. Therefore, it follows that $||p_e - p_f||_1 \leq q_{jk_e} + q_{jk_f}$, which implies that (3.38a) is satisfied, as desired. An analogous argument shows that inequality (3.38b) is also satisfied, which gives the desired result. □

Proposition 7 also holds for the ℓ^∞ norm (see Appendix 6.2). By combining propositions 6 and 7 with Equation (3.26) we get the following result.

Corollary 1. *Under the distance measure induced by the ℓ^1 or ℓ^∞ norm and for a flow $y \in \mathbb{R}_+^{|A|}$, a reformulation of problem $v_M(y)$ is given by the following MIP:*

$$v_M(y) = \max \; c^\top y \tag{3.39a}$$

$$s.t. \; c_a = c_a^0 + \sum_{j\in[m]} \sum_{k\in[r_j]} d_{jk}\lambda_{ajk} \qquad \forall a \in A \tag{3.39b}$$

$$\sum_{j\in[m]} \sum_{k\in[r_j]} \lambda_{ajk} \leq 1 \qquad \forall a \in A \tag{3.39c}$$

$$\lambda_{ajk_1} + \lambda_{bjk_2} \leq 1 \qquad \forall (k_1, k_2) \in \Gamma^j_{a,b}, a, b \in A, a \neq b, j \in [m] \tag{3.39d}$$

$$\lambda_{ajk} \in \{0, 1\} \qquad \forall a \in A, \; j \in [m], k \in [r_j] \tag{3.39e}$$

$$c_a \in \mathbb{R} \qquad \forall a \in A. \tag{3.39f}$$

As a consequence of Corollary 1, under the distances induced by the ℓ^1 or ℓ^∞ norm, we can replace the big-M formulation (3.9) by formulation (3.39), which does not use any big-M constants. We remark that Proposition 7, and thus Corollary 1, do not hold for the ℓ^2 norm.

Remark 3. *As a counter example showing that $\mathcal{B}_j$ might not be contained in $\mathcal{M}_j$ even if $\lambda \in \mathcal{C}$, for the distance induced by the ℓ^2 norm, consider an instance with one disruption denoted by j_1, one ripple denoted by k_1, and three arcs moving flow $a, b, c \in A$. Suppose that the locations of the arcs are given by $p_a = (p_a^{(1)}, p_a^{(2)}) = (0, 0)$, $p_b = (p_b^{(1)}, p_b^{(2)}) = (2, 0)$, and $p_c = (p_c^{(1)}, p_c^{(2)}) = (1, 1 + \epsilon)$ for some value $0 < \epsilon \leq \sqrt{3} - 1$ and assume that the radius of the disruption is $q_{j_1 k_1} = 1$, see Figure 3.7. Note that $(k_1, k_1) \notin \Gamma_{a,b}^{j_1}$, $(k_1, k_1) \notin \Gamma_{b,c}^{j_1}$, and $(k_1, k_1) \notin \Gamma_{a,c}^{j_1}$. Therefore, if $\hat{\lambda}_{aj_1k_1} = \hat{\lambda}_{bj_1k_1} = \hat{\lambda}_{cj_1k_1} = 1$ then $\hat{\lambda} = (\hat{\lambda}_{aj_1k_1}, \hat{\lambda}_{bj_1k_1}, \hat{\lambda}_{cj_1k_1}) \in \mathcal{B}_{j_1}$ and $\hat{\lambda} \in \mathcal{C}$. However, there does not exist $\alpha_{j_1}^{(1)}$ and $\alpha_{j_2}^{(2)}$ that satisfies* (3.10d) *for $\hat{\lambda}$, i.e., $\hat{\lambda} \notin \mathcal{M}_{j_1}$. Indeed, there is no circle of radius one in Figure 3.7 that can cover p_a, p_b, and p_c simultaneously.*

The underlying geometric reason for Proposition 7 to hold for the ℓ^1 and ℓ^∞ case and not for the ℓ^2 case is that rectangles satisfy the Helly property *(Dourado et al., 2012), i.e., every family of pairwise intersecting rectangles has a non-empty intersection, while circles do not. Nevertheless, for the ℓ^2 norm the constraints in $\mathcal{B}_j$ remain valid, i.e., $\mathcal{M}_j \subseteq \mathcal{B}_j$.*

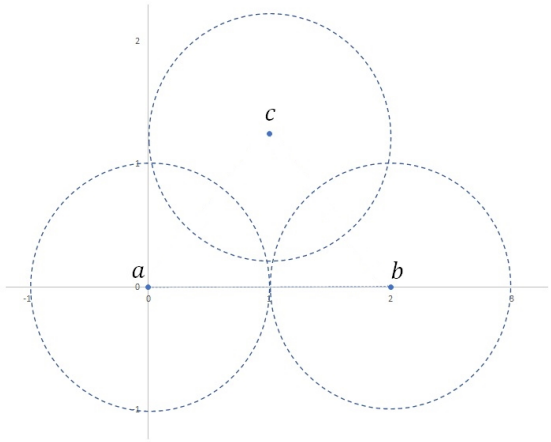


Figure 3.7: A counter example to Proposition 7 under the ℓ^2 norm. Each circle represents the locations for the epicenter of a disruption having one ripple of radius 1, for which the corresponding arc would be damaged.

Interestingly, formulation (3.39) does not include variables for the locations of the epicenters of the disruptions, which, however, are implicitly given by the λ-variables. Let λ^* be an optimal solution of (3.39), fix $j \in [m]$, define $A_j = \{a \in A \colon \exists k \in [r_j] \text{ such that } \lambda^*_{ajk} = 1\}$

and for each $a \in A$ define $k_{aj} = \text{argmax}\{\lambda^*_{ajk} \colon k \in [r_j]\}$. Let

$$m^*_{1j} = \max_{a \in A_j}\{-q_{jk_{aj}} + (p^{(1)}_a + p^{(2)}_a)\}, \quad M^*_{1j} = \min_{a \in A_j}\{q_{jk_{aj}} + (p^{(1)}_a + p^{(2)}_a)\}$$

$$m^*_{2j} = \max_{a \in A_j}\{-q_{jk_{aj}} + (p^{(1)}_a - p^{(2)}_a)\}, \quad M^*_{2j} = \min_{a \in A_j}\{q_{jk_{aj}} + (p^{(1)}_a - p^{(2)}_a)\}.$$

Then, from the proof of Proposition 7, it follows that any point that belongs to the segment S^j is an optimal epicenter for disruption j corresponding to λ^*, where

$$S^j = \Big\{(x,y) \in \mathbb{R}^2 \colon \exists \gamma \in [0,1] \text{ s.t. } (x,y) = \gamma\Big(\frac{m^*_{1j} + m^*_{2j}}{2}, \frac{m^*_{1j} - m^*_{2j}}{2}\Big) + (1-\gamma)\Big(\frac{M^*_{1j} + M^*_{2j}}{2}, \frac{M^*_{1j} - M^*_{2j}}{2}\Big)\Big\}. \tag{3.40}$$

A similar result holds for the ℓ^∞ norm (see Appendix 6.2).

We now show that under appropriate assumptions on the big-M constants, formulation (3.39) is stronger than formulation (3.9).

Proposition 8. *Suppose that the distance measure is induced by the ℓ^1 norm. Let polyhedrons $\mathcal{P}$ and $\hat{\mathcal{P}}$ be the LP relaxations of formulations* (3.9) *and* (3.39)*, respectively:*

i. For any $a \in A$, if the corresponding big-M coefficient M_a satisfies that

$$M_a \geq \max_{p \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}], j \in [m]} \left(\|p_a - p\|_1 - q_{j1} \right), \tag{3.41}$$

then $\hat{\mathcal{P}} \subseteq \mathcal{P}$.

ii. In addition, if there is a disruption $j \in [m]$ for which there exist arcs $b, c \in A$ such that $\Gamma^j_{b,c} \neq \emptyset$ and such that for any $a \in A$ the coefficient M_a also satisfies that

$$M_a > 2 \left(\max_{p \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}], j' \in [m], k \in [r_{j'}]} \left| \|p_a - p\|_1 - q_{j'k} \right| \right), \tag{3.42}$$

then $\hat{\mathcal{P}} \subsetneq \mathcal{P}$.

Proof. First, we show part (ii) by showing that there exist feasible solutions in $\mathcal{P}$ that are not feasible for $\hat{\mathcal{P}}$. Assume, without loss of generality, that for any disruption all arcs belong to at least one ripple (which can be ensured by including an artificially large ripple with a damage increase of zero).

Let (λ^*, α^*) be an optimal solution to problem (3.9). For any arc $a \in A$ let j_a be the disruption that is active at arc a and let k_a be the ripple of disruption j_a to where a belongs in the optimal solution. On the other hand, pick two ripples k_1 and k_2 such that $(k_1, k_2) \in \Gamma^j_{b,c}$ (notice that these ripples exist because we assume that $\Gamma^j_{b,c} \neq \emptyset$). Note that $\lambda^*_{bjk_1} + \lambda^*_{cjk_2} \leq 1$ and consider the following two cases:

Case 1. $\lambda^*_{bjk_1} = 1, \lambda^*_{cjk_2} = 0$**:** Because $\lambda^*_{cj_ck_c} = 1$, define a new feasible $\tilde{\lambda}$ as λ^* but set $\tilde{\lambda}_{cj_ck_c} = 1 - \epsilon$, and $\tilde{\lambda}_{cjk_2} = \epsilon$ where

$$\epsilon = 1 - \frac{|\|p_c - \alpha_j\| - q_{jk_2}|}{M_c}$$

From our assumptions on M_c in (3.42) it follows that $\tilde{\lambda} \in \mathcal{P}$. Moreover, in this case, $\tilde{\lambda}_{bjk_1} + \tilde{\lambda}_{cjk_2} > 1$, which violates (3.39d) and thus $\tilde{\lambda} \notin \hat{\mathcal{P}}$.

Case 2. $\lambda^*_{bjk_1} = \lambda^*_{cjk_2} = 0$**:** Recall that $\lambda^*_{bj_bk_b} = \lambda^*_{cj_ck_c} = 1$ and define a new solution $\tilde{\lambda}$ as λ^* but change $\tilde{\lambda}_{bj_bk_b} = 1 - \epsilon_b$, $\tilde{\lambda}_{bjk_1} = \epsilon_b$, $\tilde{\lambda}_{cj_ck_c} = 1 - \epsilon_c$, and $\tilde{\lambda}_{cjk_2} = \epsilon_c$ where

$$\epsilon_b = 1 - \frac{|\|p_b - \alpha_j\| - q_{jk_1}|}{M_b}$$

$$\epsilon_c = 1 - \frac{|\|p_c - \alpha_j\| - q_{jk_2}|}{M_c}$$

From our assumption in (3.42) it follows that $\tilde{\lambda} \in \mathcal{P}$. Moreover, it is concluded that $\epsilon_b > 1/2$ and $\epsilon_c > 1/2$ which implies that $\tilde{\lambda}_{bjk_1} + \tilde{\lambda}_{cjk_2} > 1$ violating (3.39d). So, $\tilde{\lambda} \notin \hat{\mathcal{P}}$.

Now, we prove (i), i.e., that any feasible solution $\hat{\lambda}$ in $\hat{\mathcal{P}}$ satisfies the constraints in (3.9).

Similar to Proposition 7, we must show that for any disruption $j \in [m]$ there exist $\alpha_j^{(1)}$ and $\alpha_j^{(2)}$ that make the constraints hold. In other words, we aim to prove that there exist $\alpha_j^{(1)}$ and $\alpha_j^{(2)}$ such Equations (3.43) hold:

$$\begin{aligned} -q_{jk} - M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} + p_a^{(2)}) &\leq \alpha_j^{(1)} + \alpha_j^{(2)} \\ &\leq q_{jk} + M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} + p_a^{(2)}) \ \forall a \in A, k \in [r_j] \qquad (3.43a) \\ -q_{jk} - M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} - p_a^{(2)}) &\leq \alpha_j^{(1)} - \alpha_j^{(2)} \\ &\leq q_{jk} + M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} - p_a^{(2)}) \ \forall a \in A, k \in [r_j]. \qquad (3.43b) \end{aligned}$$

To this end, let $\max_{a\in A,k\in[r_j]}\{-q_{jk} - M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} + p_a^{(2)})\} = -q_{jk_e} - M_e(1-\hat{\lambda}_{ejk_e}) + (p_e^{(1)} + p_e^{(2)})$, $\min_{a\in A,k\in[r_j]}\{q_{jk} + M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} + p_a^{(2)})\} = q_{jk_f} + M_f(1-\hat{\lambda}_{fjk}) + (p_f^{(1)} + p_f^{(2)})$, $\max_{a\in A,k\in[r_j]}\{-q_{jk} - M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} - p_a^{(2)})\} = -q_{jk_g} - M_g(1-\hat{\lambda}_{gjk_g}) + (p_g^{(1)} - p_g^{(2)})$, and $\min_{a\in A,k\in[r_j]}\{q_{jk} + M_a(1-\hat{\lambda}_{ajk}) + (p_a^{(1)} - p_a^{(2)})\} = q_{jk_h} + M_h(1-\hat{\lambda}_{hjk_h}) + (p_h^{(1)} - p_h^{(2)})$. Therefore, Equations (3.43) hold if and only if Equations (3.44) hold:

$$(p_e^{(1)} - p_f^{(1)}) + (p_e^{(2)} - p_f^{(2)}) - M_e(1-\hat{\lambda}_{ejk_e}) - M_f(1-\hat{\lambda}_{fjk_f}) \leq q_{jk_e} + q_{jk_f} \qquad (3.44a)$$

$$(p_g^{(1)} - p_h^{(1)}) + (p_h^{(2)} - p_g^{(2)}) - M_g(1-\hat{\lambda}_{gjk_g}) - M_h(1-\hat{\lambda}_{hjk_h}) \leq q_{jk_g} + q_{jk_h}. \qquad (3.44b)$$

If $(k_e, k_f) \notin \Gamma_{e,f}^j$, then (3.44a) holds from the definition of $\Gamma_{e,f}^j$. Otherwise, suppose that $(k_e, k_f) \in \Gamma_{e,f}^j$, which implies that $\hat{\lambda}_{ejk_e} + \hat{\lambda}_{fjk_f} \leq 1$. Let $M = \min\{M_e, M_f\}$, then we have

$$\begin{aligned} &(p_e^{(1)} - p_f^{(1)}) + (p_e^{(2)} - p_f^{(2)}) - M_e(1-\hat{\lambda}_{ejk_e}) - M_f(1-\hat{\lambda}_{fjk_f}) \qquad (3.45) \\ &\leq (p_e^{(1)} - p_f^{(1)}) + (p_e^{(2)} - p_f^{(2)}) - M(2-\hat{\lambda}_{ejk_e} - \hat{\lambda}_{fjk_f}) \qquad (3.46) \\ &\leq (p_e^{(1)} - p_f^{(1)}) + (p_e^{(2)} - p_f^{(2)}) - M \leq 0 \leq q_{jk_e} + q_{jk_f}. \qquad (3.47) \end{aligned}$$

Hence, the inequality (3.44a) holds from the assumptions on the 'big-M' in Equation (3.41).

Similar arguments as above give the result for (3.44b). □

Proposition 8 also holds for the ℓ^∞ norm (see Appendix 6.2). We close this subsection by noting that all the results presented for $v_M(y)$ also hold true for $v_L(y)$ (see Appendix 6.3).

3.4.3 The multiple disruption problems and maximum cliques

A direct consequence of our proposed reformulations is that both $v_M(y)$ and $v_L(y)$ can be framed as weighted maximum clique problems in auxiliary graphs. This equivalence allows the use of graph algorithms to solve the MIPs. Importantly, it implies that if the disruptions only have one ripple, then $v_L(y)$ can be found by solving a sequence of polynomially solvable maximum clique problems in intersection graphs.

Consider the auxiliary undirected graph $G_M = (V_M, E_M)$, where the vertex set is

$$V_M = \big\{(a, j, k) \colon a \in A, k \in [r_j], j \in [m]\big\}, \tag{3.48}$$

and the edge set is

$$E_M = E_M^0 \cup \bigcup_{j \in [m]} E_M^j, \tag{3.49}$$

where for each $j \in [m]$:

$$E_M^j = \Big\{\big\{(a, j, k), (b, j, \ell)\big\} \colon a, b \in A, a \neq b, (k, \ell) \notin \Gamma_{a,b}^j\Big\}, \tag{3.50}$$

with

$$E_M^0 = \Big\{\big\{(a, j, k), (b, i, \ell)\big\} \colon a, b \in A, a \neq b, (k, \ell) \in [r_j] \times [r_i], i, j \in [m], i \neq j\Big\}. \tag{3.51}$$

Assume that G_M has vertex weights that depend of the flow y and are given by

$$w_{(a,j,k)} = d_{jk} y_a \qquad \text{for all } a \in A, k \in [r_j], j \in [m]. \tag{3.52}$$

Let $\omega_M(y)$ be the value of the weighted maximum clique problem in G_M using the flow vector y in the definition of the weights. Using the integer programming formulation for the maximum clique problem (Bomze et al., 1999), where $\gamma_{ajk} = 1$ if and only if vertex (a, j, k) is in the clique and zero otherwise, we have that

$$\omega_M(y) = \max \sum_{a \in A} \sum_{j \in [m]} \sum_{k \in [r_j]} d_{jk} y_a \gamma_{ajk} \tag{3.53a}$$

$$\text{s.t. } \gamma_{ajk} + \gamma_{ai\ell} \leq 1 \qquad \forall a \in A, (k, \ell) \in [r_j] \times [r_i], j, i \in [m], (j, k) \neq (i, \ell) \tag{3.53b}$$

$$\gamma_{ajk} + \gamma_{bj\ell} \leq 1 \qquad \forall (k, \ell) \in \Gamma^j_{a,b}, j \in [m], a, b \in A, a \neq b \tag{3.53c}$$

$$\gamma_{ajk} \in \{0, 1\} \qquad \forall a \in A, k \in [r_j], j \in [m]. \tag{3.53d}$$

Observe that the constraints in (3.53b) are equivalent to the constraints in (3.39c), therefore we have proven the following result.

Proposition 9. *Let a flow $y \in \mathbb{R}^{|A|}_+$ be given. If the distance measure is induced by the ℓ^1 or ℓ^∞ norm then $v_M(y) = \omega_M(y) + (c^0)^\top y$.*

An analogous result holds for $v_L(y)$; here the auxiliary graph $G_L = (V_L, E_L)$ has more edges than G_M. Specifically, we have $V_L = V_M$ and $E_L = E^0_L \cup \bigcup_{j \in [m]} E^j_L$, where $E^j_L = E^j_M$ for any $j \in [m]$ and

$$E^0_L = \Big\{ \big\{ (a, j, k), (b, i, \ell) \big\} \colon a, b \in A, (k, \ell) \in [r_j] \times [r_i], i, j \in [m], i \neq j \Big\}. \tag{3.54}$$

Notice that E_L has more edges than E_M because in E_L edges of the form $\{(a, k, j), (a, \ell, i)\}$,

with $i \neq j$, are allowed. Let $\omega_L(y)$ be the value of the maximum weighted clique in G_L, where the weights are defined as in Equation (3.52). Then we have the following result:

Proposition 10. *Let a flow $y \in \mathbb{R}_+^{|A|}$ be given. If the distance measure is induced by the ℓ^1 or ℓ^∞ norm then $v_L(y) = \omega_L(y) + (c^0)^\top y$.*

Of particular interest is the application of Proposition 10 for single-disruption problems. In this case, for a fixed j, V_L and E_L simplify to $V_L = \{(a,k)\colon a \in A, k \in [r_j]\}$ and $E_L = \big\{\{(a,k),(b,\ell)\}\colon (k,\ell) \notin \Gamma^j_{a,b}, a,b \in A, a \neq b\big\}$. This graph looks 'almost' like an intersection graph and its single-ripple version is, in fact, an intersection graph. Indeed, given a collection of subsets $\mathcal{S} = \{S_1, \ldots, S_i\}$, $i \geq 1$, in $\mathbb{R}^2$, its (undirected) *intersection graph* $I_{\mathcal{S}} = (\mathcal{S}, E_{\mathcal{S}})$ has a vertex for each element of $\mathcal{S}$ and has an edge between two subsets $S, S' \in \mathcal{S}$ if and only if their intersection is non-empty, i.e., $E_{\mathcal{S}} = \{(S,S') \in \mathcal{S} \times \mathcal{S}\colon S \neq S', S \cap S' \neq \emptyset\}$. Assume that disruption j has only one ripple and for any $a \in A$ and $s \geq 1$ let B_{asj} be the ball in $\mathbb{R}^2$ centered at p_a with radius q_j using the distance induced by the ℓ^s-norm. That is, $B_{asj} = \{p \in \mathbb{R}^2\colon ||p_a - p||_s \leq q_j\}$. Then, from the definitions of G_L and $\Gamma^j_{a,b}$, it is readily checked that if $r_j = 1$ then $G_L = I_{\mathcal{S}_{sj}}$, where $\mathcal{S}_{sj} = \{B_{asj}\colon a \in A\}$.

The importance of the above discussion is that there are fast polynomial time algorithms to find weighted maximum cliques in intersection graphs of balls based on the distance induced by the ℓ^1 and ℓ^∞ norms (Imai and Asano, 1983). When applied to $I_{\mathcal{S}_{sj}}$ these methods yield a solution algorithm for $v_L(y)$ for a single disruption and a single ripple in $O\big(|A|\log(|A|)\big)$ time. For the distance induced by the ℓ^2 norm, similar results are also available, via slightly different techniques. Indeed, in this case $v_L(y) = (c^0)^\top y + \tilde{\omega}_L(y)$ where $\tilde{\omega}_L(y)$ is the value of the maximum weighted *geometric* clique in $I_{\mathcal{S}_{sj}}$ (De et al., 2014). It turns out that finding such maximum weighted geometric cliques in $I_{\mathcal{S}_{sj}}$ can be done in $O(|A|^2)$ time (Chazelle and Lee, 1986).

3.5 Adjustable uncertainty sets

Our proposed models, with uncertainty sets $\mathcal{U}_L$ and $\mathcal{U}_M$, assume that all m disruptions occur at the same time. This assumption might be too conservative for some real-world applications. Hence, we introduce extensions of our formulations that consider a budget constraint to adjust the level of conservativeness for the models. Particularly, the budget constraint limits the number of disruptions (out of m possible disruptions) that can occur to be less than or equal to a budget parameter.

Let $\Lambda \leq m$ be the budget parameter and let binary decision variables $\rho_j, \forall j \in [m]$, take a value of 1 if disruption j occurs. Our adjustable uncertainty sets $\mathcal{U}_L^\Lambda$ and $\mathcal{U}_M^\Lambda$ are defined exactly as before with the addition of the following constraints:

$$\sum_{j \in [m]} \rho_j \leq \Lambda \tag{3.55}$$

$$\sum_{k \in [r_j]} \lambda_{ajk} \leq r_j \rho_j \qquad \forall a \in A, j \in [m]. \tag{3.56}$$

Constraint (3.55) ensures that at most Λ out of m disruptions occur, while constraints (3.56) enforce that λ-variables can only take a value of 1 if their corresponding disruption $j \in [m]$ is active.

Our adjustable uncertainty sets follow the budgeted uncertainty discussed in detail by Bertsimas and Sim (2004). The case $\Lambda = m$ corresponds to the most conservative approach that implies all m disruptions contribute to changing the nominal values in cost vector c (which we analyzed in previous sections). This scenario follows the conservative modeling of robust optimization by Soyster (1973). On the other hand, $\Lambda = 0$ results in the nominal minimum–cost flow problem since no disruption affects the nominal cost values and thus $c = c^0$.

3.5.1 Adjustable uncertainty set for the linear model

We define the adjustable uncertainty set for the linear model as $\mathcal{U}_L^\Lambda = \{c \in \mathbb{R}_+^{|A|} \ : \ \exists \alpha_j \in \mathbb{R}^2, \ \forall j \in [m]; \ \lambda_{ajk} \in \{0,1\}, \ \forall a \in A, j \in [m], k \in [r_j]; \ \rho_j \in \{0,1\}, \ \forall j \in [m]$ s.t. Eqs. (3.57a)-(3.57e) hold$\}$, where

$$c_a = c_a^0 + \sum_{j \in [m]} \sum_{k \in [r_j]} d_{jk} \lambda_{ajk} \qquad \forall a \in A \tag{3.57a}$$

$$\sum_{k \in [r_j]} \lambda_{ajk} \leq \rho_j \qquad \forall a \in A, \ j \in [m] \tag{3.57b}$$

$$\sum_{j \in [m]} \rho_j \leq \Lambda \tag{3.57c}$$

$$D(p_a, \alpha_j) - M_a(1 - \lambda_{ajk}) \leq q_{jk} \ \forall a \in A, \ j \in [m], \ k \in [r_j] \tag{3.57d}$$

$$L^{(i)} \leq \alpha_j^{(i)} \leq U^{(i)} \qquad \forall j \in [m], \ i = 1, 2. \tag{3.57e}$$

Note that constraints (3.5c) and (3.56) are reduced to constraint (3.57b), that $\mathcal{U}_L^\Lambda \subseteq \mathcal{U}_L$, and if $\Lambda = m$, then $\mathcal{U}_L^\Lambda = \mathcal{U}_L$.

Consider the adversarial attacker problem over adjustable uncertainty set $\mathcal{U}_L^\Lambda$ given by

$$v_L^\Lambda(y) = \max\{c^\top y : c \in \mathcal{U}_L^\Lambda\}. \tag{3.58}$$

In Section 3.3, we showed that the adversarial attacker problem defined over uncertainty set $\mathcal{U}_L$ (without the budget constraint) could be solved in polynomial time by MDDLA. Here, we propose a similar approach to address adversarial problem $v_L^\Lambda(y)$ that solves the problem in polynomial time as well.

For a given flow vector $y \in \mathbb{R}_+^{|A|}$, define the problem of finding the worst-case location for

the j^{th} disruption (ignoring the effects of all the other disruptions) as follows:

$$u^{(j)}(y) = \max \sum_{a \in A} y_a \left(\sum_{k \in [r_j]} d_{jk} \lambda^j_{ak} \right) \tag{3.59a}$$

$$\text{s.t.} \sum_{k \in [r_j]} \lambda^j_{ak} \leq 1 \qquad \forall a \in A \tag{3.59b}$$

$$D(p_a, \alpha_j) - M_a(1 - \lambda^j_{ak}) \leq q_{jk} \qquad \forall a \in A, k \in [r_j] \tag{3.59c}$$

$$L^{(i)} \leq \alpha^{(i)}_j \leq U^{(i)} \qquad i = 1, 2 \tag{3.59d}$$

$$\lambda^j_{ak} \in \{0, 1\} \qquad \forall a \in A,\ k \in [r_j]. \tag{3.59e}$$

Problem (3.59) is a *Single Disruption Problem*, which can be solved in $O(|A|^3 r^3)$ time as discussed in Section 3.3.2.

We propose a *Sorting Multiple Disruptions Discrete Location Algorithm*, that solves problem (3.59) m times, once for each possible disruption, to obtain a sequence of optimal values $u^{(1)}(y), u^{(2)}(y), \ldots, u^{(m)}(y)$. Our approach then sorts the sequence of values in a descending order and records the largest Λ values in the sequence and their corresponding optimal solutions. Proposition 11 shows that this provides an optimal solution for $v^{\Lambda}_L(y)$.

Proposition 11. *Let $u^{(1)}(y), u^{(2)}(y), \ldots, u^{(\Lambda)}(y)$ be the largest Λ values of the sorted sequence described above. Then,*

$$v^{\Lambda}_L(y) = c_0^{\top} y + \sum_{j \in [\Lambda]} u^{(j)}(y). \tag{3.60}$$

Proof. Let $(\lambda^*, \rho^*) = (\lambda^*_1, \ldots, \lambda^*_m, \rho^*_1, \ldots, \rho^*_m)$ be an optimal solution to problem (3.58), where $\lambda^*_j = (\lambda^*_{ajk} : a \in A, k \in [r_j])$ is the corresponding λ-vector for disruption $j \in [m]$. Since λ^*_j is a feasible solution to problem $u^{(j)}(y)$ for all $j \in [m]$, we obtain that

$$\sum_{a \in A} y_a \left(\sum_{k \in [r_j]} d_{jk} \lambda^*_{ajk} \right) \leq \sum_{a \in A} y_a \left(\sum_{k \in [r_j]} d_{jk} \tilde{\lambda}^j_{ak} \right) \qquad \forall j \in [m], \tag{3.61}$$

where $\tilde{\lambda}^j_{ak}$ denotes an optimal solution to problem $u^{(j)}(y)$. Constraint (3.57c) ensures that at most Λ variables ρ^*_j are different from zero. Let $\mathcal{M} \subseteq [m]$ be the index set of disruptions associated with non-zero ρ^*_j variables. Therefore, from (3.61) we have that

$$\sum_{j\in\mathcal{M}}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\lambda^*_{ajk}\right) \leq \sum_{j\in\mathcal{M}}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right), \tag{3.62}$$

and by adding constant $c_0^\top y$ to both sides we obtain

$$v^\Lambda_L(y) \leq c_0^\top y + \sum_{j\in\mathcal{M}}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right). \tag{3.63}$$

Now consider a vector $\tilde{\lambda} = (\tilde{\lambda}_1, \ldots, \tilde{\lambda}_\Lambda, \mathbf{0}, \ldots, \mathbf{0})$, i.e., the first Λ positions correspond to optimal solutions associated with $u^{(1)}(y)$, $u^{(2)}(y)$, $\ldots$, $u^{(\Lambda)}(y)$ in the ordered sequence (reordering the indexes if necessary). Also consider a vector $\tilde{\rho} = (1, \ldots, 1, 0, \ldots, 0)$ in which the first Λ positions are equal to 1 and the remaining positions are equal to 0. Since $(\tilde{\lambda}, \tilde{\rho})$ is a feasible solution to problem $v^\Lambda_L(y)$, it follows that

$$c_0^\top y + \sum_{j\in[\Lambda]}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right) \leq v^\Lambda_L(y). \tag{3.64}$$

From (3.63) and (3.64), we conclude that

$$c_0^\top y + \sum_{j\in[\Lambda]}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right) \leq v^\Lambda_L(y) \leq c_0^\top y + \sum_{j\in\mathcal{M}}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right). \tag{3.65}$$

Now we show that

$$c_0^\top y + \sum_{j\in[\Lambda]}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right) = c_0^\top y + \sum_{j\in\mathcal{M}}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right). \tag{3.66}$$

Suppose by contradiction that

$$\sum_{j\in[\Lambda]}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right) < \sum_{j\in\mathcal{M}}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right), \tag{3.67}$$

Since $\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}_{ajk}\right) \geq 0$ for all $j \in [m]$, then there must exist at least one $j' \in \mathcal{M} \setminus [\Lambda]$ and $j'' \in [\Lambda]$ such that

$$\sum_{a\in A} y_a \left(\sum_{k\in[r_{j''}]} d_{j''k}\tilde{\lambda}^{j''}_{ak}\right) < \sum_{a\in A} y_a \left(\sum_{k\in[r_{j'}]} d_{j'k}\tilde{\lambda}^{j'}_{ak}\right), \tag{3.68}$$

which implies that $u^{(j'')}(y) < u^{(j')}(y)$ and contradicts the assumption that the $u^{(j)}(y)$ values are sorted in a descending order. We conclude that

$$v^{\Lambda}_L(y) = c_0^\top y + \sum_{j\in[\Lambda]}\sum_{a\in A} y_a \left(\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}\right) = c_0^\top y + \sum_{j\in[\Lambda]} u^{(j)}(y), \tag{3.69}$$

as desired. □

For any $a \in A$, the corresponding optimal cost obtained by our proposed approach is

$$c^*_a = c^0_a + \sum_{j\in[\Lambda]}\sum_{k\in[r_j]} d_{jk}\tilde{\lambda}^j_{ak}, \tag{3.70}$$

where the j indexes are reordered according to the ordered sequence of optimal values discussed above. Our approach solves m single disruption problems, each one takes $O(|A|^3 r^3)$ time, and performs a sorting procedure in $O(m \log m)$ time, resulting in a total of $O(m|A|^3 r^3 + m \log m)$ time, which is polynomial.

3.5.2 Adjustable uncertainty set for the maximum model

We define the adjustable uncertainty set for the maximum model as $\mathcal{U}_M^{\Lambda} = \{c \in \mathbb{R}_+^{|A|} : \exists \alpha_j \in \mathbb{R}^2,\ \forall j \in [m];\ \lambda_{ajk} \in \{0,1\},\ \forall a \in A, j \in [m], k \in [r_j];\ \rho_j \in \{0,1\},\ \forall j \in [m]$ s.t. Eqs. (3.71a)- (3.71f) hold$\}$, where

$$c_a = c_a^0 + \sum_{j\in[m]} \sum_{k\in[r_j]} d_{jk}\lambda_{ajk} \qquad \forall a \in A \tag{3.71a}$$

$$\sum_{j\in[m]} \sum_{k\in[r_j]} \lambda_{ajk} \leq 1 \qquad \forall a \in A \tag{3.71b}$$

$$\sum_{k\in[r_j]} \lambda_{ajk} \leq \rho_j \qquad \forall a \in A,\ j \in [m] \tag{3.71c}$$

$$\sum_{j\in[m]} \rho_j \leq \Lambda \tag{3.71d}$$

$$D(p_a, \alpha_j) - M_a(1 - \lambda_{ajk}) \leq q_{jk}\ \forall a \in A,\ j \in [m], k \in [r_j] \tag{3.71e}$$

$$L^{(i)} \leq \alpha_j^{(i)} \leq U^{(i)} \qquad \forall j \in [m], i = 1, 2. \tag{3.71f}$$

The corresponding adversarial problem for the budgeted uncertainty set is given by

$$v_M^{\Lambda}(y) = \max\{c^\top y : c \in \mathcal{U}_M^{\Lambda}\}. \tag{3.72}$$

Similarly to the adversarial problem for the maximum model without the budget constraint, problem $v_M^{\Lambda}(y)$ is NP-Hard (as $v_M(y)$ is a special case with $\Lambda = m$). Our big-M free reformulation (3.39) is still valid for problem $v_M^{\Lambda}(y)$ after adding constraints (3.55) and (3.56), and the results presented in propositions (6) to (8) are still valid following some minor modifications to the arguments.

3.6 Numerical experiments

We study the computational performance of Algorithm 1 using the different approaches proposed to solve $v_L(y)$ and $v_M(y)$. We generate a testbed of 80 problem instances based on a grid network structure commonly used in the literature (Israeli and Wood, 2002; Cappanera and Scaparra, 2011; Lozano and Smith, 2017b; Borrero and Lozano, 2020) and we also consider real road networks taken from Karduni et al. (2016). Section 3.6.2 contains details about the selection of parameters and instance generation. For the grid networks (Tables 3.1 and 3.2), we consider problems having $m = 3$ ℓ^1-shaped disruptions with $r_1 = 5$, $r_2 = 7$, and $r_3 = 9$ ripples. Additionally, we perform sensitivity analysis to evaluate the effect of varying the number of disruptions and ripples on the computational performance of the algorithm and how this affects the robust costs (Table 3.3). For the real road networks (Table 3.4), we consider problems having $m = 2$ disruptions with $r_1 = 3$ and $r_2 = 5$ ripples. Moreover, we compare the solution quality for our proposed linear model with two other approaches via a simulation study in Section 3.6.7.

All algorithms are coded in C++ and solved using ILOG CPLEX 12.10.0 on a 164 node server, each with an Intel Skylake 6130 CPU and 96 GB of RAM under a time limit of 1 hour (3600s). All source code and datasets are publicly available in Ansari et al. (2022).

Before going through the results of our experiments, we introduce a preprocessing technique that removes arcs from the network, improving the computation of $v_L(y)$ and $v_M(y)$ for all the methods considered.

3.6.1 Removing inactive arcs

The number of variables and constraints in all the proposed formulations increases with the number of arcs in the network, thus the size of the problem can be significantly reduced by only considering a set of *active arcs* $\mathcal{A}'(y) = \{a \in A : y_a > 0\}$ instead of the complete set of arcs A. Recall that Algoritm 1 solves problem $v^\ell = \max\{c^\top y^\ell \colon c \in \mathcal{U}\}$ at iteration ℓ for a

given flow vector y^ℓ. Note that the coefficient in the objective function for any arc $a \notin \mathcal{A}'(y)$ equals 0 and thus

$$c^\top y^\ell = \sum_{a \in \mathcal{A}'(y^\ell)} y_a^\ell c_a + \sum_{a \in \mathcal{A} \setminus \mathcal{A}'(y^\ell)} 0 \cdot c_a = \sum_{a \in \mathcal{A}'(y^\ell)} y_a^\ell c_a. \tag{3.73}$$

As a result, we remove from our formulations all variables and constraints associated with arcs that do not belong to $A'(y)$, without changing the value of v^ℓ. Then, we compute the cost after disruption of all the inactive arcs $a \in A \setminus A'(y)$ by considering the epicenter of each disruption $j \in [m]$ and adding the corresponding damage(s) to the initial cost of the arc according to the linear or maximum assumptions.

3.6.2 Selection of parameter setting

Here, we determine the common parameters in all experiments:

- **Initial cost** c_a^0 is set by the Euclidean distance between nodes that are incident with arc $a \in A$.
- **The location** p_a is assumed to be the midpoint coordinates of arc $a \in A$, i.e., the average of coordinates of nodes that are incident with a.
- **The damage** d_{jk} is computed by the formula $d_{jk} = \frac{D}{(1.5)^k}$ for one disruption $j \in [m]$ and ripple $k \in [r_j]$, where we set $D = 40$.
- **The region** q_{jk} is computed by the formula $q_{jk} = \frac{kQ}{r_j}$ for one disruption $j \in [m]$ and ripple $k \in [r_j]$ where Q is a constant which determines the largest ripple and other smaller ripples are a portion of this constant. The parameter Q is set to a value that the largest ripple covers all engaged arcs between the source and sink.
- **Big value** M_a is defined as the distance from the furthest point of midpoint of arc a in the area $[L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}]$ to this arc (see (3.41)). The reason of this definition is

that disruptions do not happen out of the network area and we would prefer to define the value M_a as small as the problem remains feasible.

- **A tolerance value** ϵ is set 0.1 for all experiments. As a result, we stop the cutting-plane algorithm whenever the absolute difference between the upper and lower bound is less than 0.1.

- **A time limit** of one hour is set for running each algorithm on datasets. If the solver does not find an ϵ-optimal solution to the problem within one hour, the problem is unsolved and the gap between the last optimal values of master and subproblem is reported.

- **Optimal epicenters** are obtained by the following feasible values (recall the inequalities of (3.36) and (3.37)):

$$\alpha_j^{(1)} = \frac{\max_{a\in A}\{-q_{jk_a} + (p_a^{(1)} + p_a^{(2)})\} + \max_{a\in A}\{-q_{jk_a} + (p_a^{(1)} - p_a^{(2)})\}}{2} \tag{3.74a}$$

$$\alpha_j^{(2)} = \frac{\max_{a\in A}\{-q_{jk_a} + (p_a^{(1)} + p_a^{(2)})\} - \max_{a\in A}\{-q_{jk_a} + (p_a^{(1)} - p_a^{(2)})\}}{2}. \tag{3.74b}$$

- **The grid network structure** in Sections 3.6.3 and 3.6.4 was introduced by Israeli and Wood (2002). The grid network $G = (N, A)$ has these characteristics (see Figure 3.8):

  1. Two nodes s and t are generated as source and sink at left and right of the area, respectively.
  2. $L \geq 1$ layers and $H \geq 1$ nodes per layer, determine the structure of G.
  3. Locations of all $L \times H$ "transshipment nodes", source s, and sink t are randomly generated by uniform distributions on $[L^{(1)}, U^{(1)}]$ and $[L^{(2)}, U^{(2)}]$ for x-coordinate and y-coordinate, respectively.
  4. There are H arcs from the source node s to all nodes in the first layer. Also, there are H arcs head to the sink node t from all nodes in the last layer.

5. There are arcs from the node in grid position (l, h) to the nodes in grid positions $(l, h-1)$, $(l, h+1)$, $(l+1, h-1)$, $(l+1, h)$, and $(l+1, h+1)$ if a node exists in these positions.

Thus, we have: $|N| = LH + 2$ and $|A| = 5LH - 4L - H + 2$.

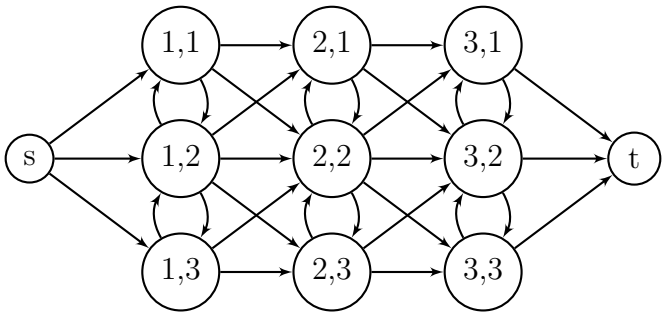


Figure 3.8: The topology of 3×3 grid network

In addition, we consider the following assumptions for our experiments over the real road network dataset:

- We set the node with the maximum degree as source node.
- We apply a breadth-first search algorithm to find the furthest node from the source node and select it as sink.
- We set the value of parameter Q (the radius of the largest ripple) equal to the ℓ^1-norm distance between source and sink.

3.6.3 Linear model for multiple ripple effect disruptions over grid networks

We compare four different methods for solving subproblem $v_L(y)$ for a given flow vector $y \in \mathbb{R}_+^{|A|}$:

1. EF: Extended MIP formulation (section 3.2.2),

2. DF: Decomposed MIP formulations (section 3.3.1),

3. PA: Polynomial Time Algorithm (section 3.3.2),

4. BR: Binary MIP Reformulation (section 3.4.2)

Table 3.1 compares the performance of each method in terms of the solution times, optimality gaps, number of iterations, and number of instances solved within the time limit. The first column presents the configuration of the grid. Columns 2 and 3 show the number of nodes and arcs in the network, respectively. Columns 4–7 present the average CPU time in seconds for each method, computed across 5 randomly generated instances having the same configuration. Columns 8–9 compare the average time required to solve one adversarial problem via EF and BR, respectively. Columns 10–13 show the average number of iterations. Columns 14–17 present the number of instances solved to optimality within the time limit and in parenthesis the average optimality gap across the instances not solved to optimality. Columns 18–20 compare the average optimality gap at the root node of the adversarial problems for EF and BR, computed based on the linear relaxation without including any cuts or preprocessing techniques. In many cases, the cutting-plane algorithm exceeds the time limit while solving a MIP subproblem. In that case, we wait until the MIP subproblem finishes its execution before terminating the algorithm, which results in some computational times reported being larger than the 1 hour time limit.

Table 3.1 shows that DF solves 39 out of 80 instances within the time limit while EF only solves 22. The average solution times for DF are also slightly better than the solution times for EF, showing that the decomposition approach results in a moderate improvement in computational performance when compared to the extended formulation. On the other hand, PA and BR greatly outperform EF and DF, solving every single instance within the time limit and reducing the computation times by an order of magnitude in most cases. The average time to solve the adversarial problems shows that BR solves the reformulated

Network	Nodes	Arcs	Solution Time (s)				Adv. Time (s)		# Iterations				# Solved (gap %)				Root Gap %	
			EF	DF	PA	BR	EF	BR	EF	DF	PA	BR	EF	DF	PA	BR	EF	BR
5×5	27	102	85	20	2	2	2.9	0.1	29	29	35	33	5	5	5	5	27.33%	0.04%
5×10	52	222	623	405	4	6	14.8	0.1	42	42	36	43	5	5	5	5	36.70%	0.13%
5×15	77	342	2921	1657	16	19	68.2	0.3	43	54	57	57	2 (0.34)	5	5	5	40.69%	0.22%
5×20	102	462	3298	2228	19	22	92.1	0.4	36	56	59	57	1 (0.34)	4 (0.05)	5	5	41.84%	0.34%
10×5	52	207	776	379	10	5	21.0	0.1	37	39	42	41	5	5	5	5	33.20%	0.05%
10×10	102	452	*3622*	2328	23	17	127.5	0.4	28	45	52	44	1 (0.69)	4 (0.04)	5	5	46.19%	0.18%
10×15	152	697	*3959*	2991	35	20	164.9	0.4	24	49	60	46	0 (0.56)	3 (0.07)	5	5	46.97%	0.18%
10×20	202	942	*3934*	*3638*	116	99	207.0	1.4	19	39	71	70	0 (0.82)	0 (0.28)	5	5	45.94%	0.20%
15×5	77	312	2627	1931	17	8	97.3	0.2	27	37	39	34	3 (0.71)	5	5	5	40.43%	0.06%
15×10	152	682	*3765*	3576	102	70	257.8	1.1	15	38	63	61	0 (1.42)	1 (0.22)	5	5	45.23%	0.11%
15×15	227	1052	*3819*	*3737*	162	147	289.2	2.0	13	31	76	72	0 (1.58)	0 (0.40)	5	5	45.21%	0.21%
15×20	302	1422	*3948*	*3766*	459	418	340.2	4.4	12	25	97	93	0 (2.19)	0 (0.71)	5	5	45.44%	0.21%
20×5	102	417	*3771*	3316	41	22	232.7	0.5	16	40	46	44	0 (0.59)	2 (0.05)	5	5	39.97%	0.11%
20×10	202	912	*4019*	*3724*	95	109	352.4	1.7	11	30	56	65	0 (1.35)	0 (0.32)	5	5	43.66%	0.20%
20×15	302	1407	*3768*	*3820*	277	240	384.2	3.3	10	25	74	72	0 (2.04)	0 (0.53)	5	5	43.15%	0.19%
20×20	402	1902	*3855*	*3755*	789	562	385.2	5.7	10	21	103	98	0 (2.24)	0 (1.07)	5	5	45.59%	0.29%

Table 3.1: Comparing 4 different methods for the linear model

MIP subproblems dramatically faster than EF, which solves the original big-M formulation. The average root gaps also showcase the strength of the big-M free reformulation, which in many cases solves the subproblem to optimality at the root node of the branch-and-bound tree. Finally, we find that over all the instances in the dataset, the number of active arcs is consistently lower than roughly 10% of the total number of arcs.

3.6.4 Maximum model for multiple ripple effect disruptions over grid networks

Since for the maximum model the decomposition approaches are not applicable, we consider only the extended MIP formulation, EF (section 3.2.3), and the binary MIP reformulation, BR (section 3.4.2), for solving subproblems $v_M(y)$ for a given flow vector $y \in \mathbb{R}_+^{|A|}$. Table 3.2 shows the results of our experiments. Columns are defined in the same way as before. The average solution times and the number of instances solved by both methods suggest that the maximum model instances are considerably more difficult to solve than the linear model instances. EF only solves 30 out of 80 instances while BR solves 53. The optimality gap for the instances not solved is considerably better for BR than for EF, for example over instances 15×10 the average optimality gap for EF is 2.01% while for BR is only 0.12%. Overall, BR is considerably faster than EF, achieving speedups of up to an order of magnitude (e.g., over instances 20×5 BR is roughly 27 times faster than EF) and obtaining roughly an order of

Network	Nodes	Arcs	Solution Time (s)		Adv. Time (s)		# Iterations		# Solved (gap %)		Root Gap %	
			EF	BR	EF	BR	EF	BR	EF	BR	EF	BR
5×5	27	102	2	1	0.2	0.0	11	14	5	5	0.62%	0.13%
5×10	52	222	140	4	5.0	0.1	28	27	5	5	2.62%	0.48%
5×15	77	342	401	11	15.7	0.4	26	27	5	5	2.45%	0.51%
5×20	102	462	1640	131	39.6	2.4	41	55	4 (0.48)	5	2.89%	1.01%
10×5	52	207	56	3	2.7	0.2	21	22	5	5	1.20%	0.41%
10×10	102	452	*3728*	210	83.6	3.3	45	64	0 (0.27)	5	6.47%	1.79%
10×15	152	697	*3751*	1522	117.9	15.2	32	100	0 (1.02)	5	7.14%	2.48%
10×20	202	942	*3893*	*3653*	157.0	36.8	25	99	0 (2.10)	0 (0.27)	7.05%	1.95%
15×5	77	312	1373	39	35.7	0.9	38	43	5	5	4.39%	1.08%
15×10	152	682	*3778*	2515	162.8	24.7	23	102	0 (2.01)	4 (0.12)	7.31%	2.59%
15×15	227	1052	*3788*	3367	197.3	42.5	19	79	0 (2.75)	1 (0.34)	7.79%	1.87%
15×20	302	1422	*3808*	*3667*	232.2	51.9	16	71	0 (3.52)	0 (0.46)	7.49%	1.66%
20×5	102	417	3521	126	92.6	2.5	38	50	1 (0.21)	5	5.16%	1.41%
20×10	202	912	*3840*	2864	199.9	31.1	19	92	0 (2.13)	3 (0.15)	6.95%	2.13%
20×15	302	1407	*4014*	*3676*	254.0	54.2	16	68	0 (2.74)	0 (0.44)	7.56%	1.66%
20×20	402	1902	*3889*	*3692*	270.0	69.9	14	53	0 (3.99)	0 (0.94)	7.72%	1.52%

Table 3.2: Comparing two solution approaches for the maximum model

magnitude gap reductions for those instances not solved within an hour.

3.6.5 Sensitivity analysis on the number of disruptions and ripples

We study the effects of changing the number of disruptions and ripples on the performance of the maximum model (which results in the most time-consuming instances as shown above). We conduct additional experiments on a set of 30 randomly generated 10×10 instances and test our binary MIP reformulation (BR) varying the number of disruptions in $\{2, 3, 4, 5\}$ and the number of ripples in $\{1, 2, 4, 6\}$. We consider homogeneous disruptions (i.e., each disruption has the same number of ripples) for which their impact areas are expanded by increasing the number of ripples (i.e., as the number of ripples is higher, each disruption deals damage to larger areas). Table 3.3 presents the result for this experiment where reported values are computed as the average of the corresponding outputs across all 30 instances.

The average nominal cost (without any disruptions) of moving flow from source to sink in our data set is 84.3. As expected, increasing the number of disruptions and ripples increases the robust cost (after disruptions). As the number of ripples increases there is a point where adding additional ripples no longer increases the robust cost since there is a maximum damage to be achieved for a fixed flow and number of disruptions (e.g., the robust cost for problems

# Disruptions	# Ripples	Robust Cost	Time (s)	Iterations	Constraints	Adv. Time (s)
2	1	246.8	25	67	2519	0.4
	2	337.1	11	42	2608	0.3
	4	342.7	15	39	3013	0.4
	6	342.7	17	39	3088	0.4
3	1	309.5	232	90	4377	2.6
	2	367.9	119	57	5428	2.1
	4	368.3	108	52	5551	2.1
	6	368.3	111	53	5568	2.1
4	1	361.4	461	94	5712	4.9
	2	377.9	62	27	3065	2.3
	4	377.9	84	30	3435	2.8
	6	377.9	70	28	3221	2.5
5	1	378.0	359	82	5172	4.4
	2	378.2	3	20	2522	0.1
	4	378.2	2	16	2366	0.1
	6	378.2	1	14	1956	0.1

Table 3.3: Sensitivity analysis for the maximum model

with 4 disruptions is 377.9 for 2, 4, and 6 ripples).

Regarding computational times, all problems are solved within the one-hour time limit except one problem having 5 disruptions and 1 ripple. The average CPU time to solve problems with 3 disruptions is roughly an order of magnitude larger than the time required for 2 disruptions. For problems with 4 disruptions, there is an increase in CPU time for 1 ripple and a considerable decrease in CPU time for 2 ripples or more. We observe the same behavior for problems having 5 disruptions. These results suggest that for problems having a large number of disruptions and ripples it is easier to achieve the maximum damage for a fixed flow. Similarly, problems with few ripples and disruptions are also easy because of the fewer number of variables and conflict constraints. The most challenging problems seem to be the ones having either an intermediate number of disruptions and ripples or a combination of large number of disruptions with only one ripple. In this regard, we suspect that problems with one ripple are harder because each disruption covers less space. The less space the disruptions cover, the more challenging is the problem of deciding an optimal (worst case) placement of disruptions. This observations is supported by the fact that more iterations are needed and solving each adversarial problem takes more time on average as evidenced in Table 3.3. Finally, we observe that there

is a direct relationship between running times and the number of conflict constraints generated.

3.6.6 Ripple effect disruptions over real road networks

We conduct additional experiments to assess the scalability of our approach over large real road networks. For these experiments, we only consider method BR as it is the best performer among all the methods evaluated before. Table 3.4 shows the total CPU time required to prove optimality, the average time to solve one adversarial problem, the number of iterations, and the number of active arcs for both the linear and maximum models over each of the 12 real road networks considered.

Dataset	Nodes	Arcs	CPU Time (s)		Adv. Time (s)		# Iterations		# Active Arcs	
			Linear	Maximum	Linear	Maximum	Linear	Maximum	Linear	Maximum
Paris	1,290,375	1,291,464	838	1186	3	6	9	3	277	353
Houston	1,264,561	1,261,792	1470	820	37	63	5	2	734	869
San Francisco	1,171,685	1,169,300	1523	644	102	101	7	2	608	629
Chicago	1,127,987	1,121,620	689	556	8	14	7	3	541	584
Los Angeles	1,105,378	1,103,138	959	836	12	59	4	4	519	527
Tokyo	1,031,624	1,035,446	693	622	20	49	5	4	719	918
Madrid	1,023,814	1,023,406	942	887	13	27	6	5	659	693
London	752,871	753,958	613	913	15	38	6	7	527	662
Bogota	399,686	400,696	145	110	12	14	7	4	472	558
Tehran	294,826	295,274	41	35	1	2	3	2	291	345
Shangai	213,734	215,070	23	26	1	3	4	3	238	241
Delhi	152,472	152,726	20	21	1	1	8	7	173	200

Table 3.4: Solving RMCFP under 2 disruptions with 3 and 5 ripples on real road networks

Although these real road networks are considerably larger than the grid networks in terms of the number of arcs and nodes, BR is able to solve every problem to optimality in less than 30 minutes. In contrast to the grid networks, it would seem that the maximum model is slightly easier to solve than the linear model over the real road networks. We conjecture that the problem difficulty is highly dependent on the density of the graph and the number of paths between the source and sink nodes.

The preprocessing technique is critical in the performance of the algorithm for these large networks. On average we were able to remove roughly 99 percent of the total arcs at each iteration of the cutting-plane algorithm. Additionally, the solver is able to solve the

subproblem to optimality at the root node of the branch-and-bound tree for most of our test problems, due to the stronger representation provided by our proposed binary reformulation.

3.6.7 Simulation study

We perform a simulation study to compare the performance of our proposed model, of the budgeted uncertainty model (B Model), and of a nominal solution that ignores the disruptions altogether. The study is based on a hypothetical situation where the decision-maker (DM) does not know the parameters characterizing the disruptions, and thus seeks to select one of the three models to decide. We assume, however, that the DM can make a reasonable guess of the *ranges* of the parameters (e.g., she estimates that there are going to be between 2 to 5 disruptions, see details in Section I below). Based on this limited knowledge, the DM considers different combinations of parameters for the proposed model and for the B model, and runs them to obtain flow solutions. The performance of each flow solution is then evaluated by using simulations. Each replication of the simulation randomly generates a number of disruptions, disruptions' locations, ripples, and damage costs, which yield a *post-disruptions* cost vector. The flow solutions from the three different approaches (our model, the B model, and the nominal min cost flow solution that assumes no disruptions happen) are then evaluated using the resulting post-disruption cost vectors. We report the average, minimum and maximum cost, the standard deviation of the costs, and the optimal robust optimization costs, obtained by the two robust models. The results show that our model outperforms the B model in terms of solution quality, measured by the worst-cost realization from the simulation, while providing a less conservative estimation of the worst-case cost, given by the optimal objective value of the models.

More specifically, assume we are given a fixed *nominal* network $G = (N, A)$ with a fixed cost vector $\boldsymbol{c}^0$. We randomly generated $R = 1000$ different scenarios for the disruptions. The disruptions increase the nominal costs following the linear model of damage (see Section 3.3). The resulting post-disruption cost vector in simulation r is denoted by $\boldsymbol{c}^r$. On the other hand,

let $\mathcal{P}_1$ denote the collection of parameter values for the proposed model and $\mathcal{P}_2$ denote the collection of parameter values for the B model (see details in Sections II and III below). For each $\pi \in \mathcal{P}_1$ we solved the proposed model assuming the parameters are given by π (i.e., we solve problem (3.1) assuming that $\mathcal{U} = \mathcal{U}_L$) and obtain an optimal flow vector that is denoted by y^π. Similarly, for each $\rho \in \mathcal{P}_2$ we solved the B model assuming that the parameters are given by ρ (i.e., we solve problem (3.1) assuming that $\mathcal{U}$ is given by $\mathcal{U}_B$, see Section III below) and obtain an optimal flow vector which is denoted by y^ρ. Then, for each replication $r \in [1000]$ we compute $z^{r,\pi} = (\boldsymbol{c}^r)^\top y^\pi$ and $z^{r,\rho} = (\boldsymbol{c}^r)^\top y^\rho$, for all $\pi \in \mathcal{P}_1$ and $\rho \in \mathcal{P}_2$. In addition, we compute the min-cost flow solution y^n over G under vector $\boldsymbol{c}^0$, which is hereafter referred as the *nominal* (flow) solution, and let $z^{r,n} = (\boldsymbol{c}^r)^\top y^n$ be the cost of using the nominal solution in scenario r.

I. Random generation of instances. For each replication the number of disruptions is generated at random from a uniform discrete distribution between 2 and 5. The number of ripples is generated at random from a uniform discrete distribution between 5 and 9, independently of the number of disruptions. To compute the size of the ripples we first find Q, which is the radius of a ripple that can cover all the arcs in the network. Then, we define the radius of the largest ripple as $0.1Q$ and the radius of ripple i by $0.1iQ/r$, where r is the number of ripples. The damages are computed by first generating a uniform discrete number δ between 20 and 60, and then using $\delta/1.5^{i-1}$ for the damage of the i-th ripple. Finally, the epicenters of the disruptions are generated uniformly (continuous) at random in the rectangular region where the network is located. In order to reduce variation in our experiment, we assume that all disruptions are homogeneous, meaning they have the same number of ripples, and that the sizes and damages of the ripples are defined in the same way for all disruptions. We remark that we assume that the DM has no knowledge of all the parameters above, but only a reasonable guess about the range for the number of disruptions, the number of ripples, and the value of δ.

II. Parameters for the proposed model. For our study, we select variations of three parameters: the number of disruptions m, the number of ripples per disruption r (which is the same for all disruptions), and the value D defining the damages (where the damage of the i-th ripple is given by $D/1.5^{i-1}$). For simplicity, we assume that the radii of the ripples are the same as those of the simulations. For the experiments we take $m \in \{2, 3, 4, 5\}$, $(r, D) \in \{(5, 20), (7, 40), (9, 60)\}$. All the possible combinations between these sets give that there are 12 different values for $\pi = (m, r, D)$ in $\mathcal{P}_1$.

III. Parameters for the B model. The following interpretation of the B model assumes that at most $\Gamma > 0$ of the entries of the post-disruption cost vector are subject to uncertainty (Γ being an user-defined parameter). If the entry corresponding to arc $a \in A$ is subject to uncertainty, then it can take values in the interval $[c_a^0, c_a^0 + D]$, where $D > 0$ is a parameter. Thus, the uncertainty set has the following mixed-integer representation

$$\mathcal{U}_B = \Big\{ c \in \mathbb{R}_+^{|A|} \colon \exists \gamma \in \{0,1\}^{|A|} \text{ s.t. } c_a^0 \leq c_a \leq c_a^0 + \gamma_a D \quad \forall a \in A, \sum_{a \in A} \gamma_a \leq \Gamma \Big\}. \tag{3.75}$$

For the experiments, we let $\Gamma \in \{0.1|A|, 0.3|A|, 0.6|A|, 0.9|A|\}$ and $D \in \{20, 40, 60\}$. We consider all the possible combinations across these two sets, which results in 12 different parameter values $\rho = (\Gamma, D)$ in $\mathcal{P}_2$.

IV. Results and analysis. Following the previous computational experiments, we consider two grid networks G of sizes 5×10 and 10×10. Table 3.5 presents the results for the proposed model and Table 3.6 presents the results for the B model. The information for the nominal solution is presented at the bottom of each table (note that the nominal solution does not depend on the model, we repeat the results in each table for convenience). In each table the column "Parameters" describes the parameter used by the respective model; "Robust" shows the optimal objective value of the corresponding robust optimization problem;

"Mean" has the sample mean of the simulated objectives, i.e., $\bar{z}^{\pi} = (1/1000)\sum_{r=1}^{1000} z^{r,\pi}$ and $\bar{z}^{\rho} = (1/1000)\sum_{r=1}^{1000} z^{r,\rho}$, respectively. "SD" has the corresponding sample standard deviation; "Min" has the minimum simulated value across all 1000 replications; and "Max" has the maximum (worst-case) simulated value across all replications. For instances not solved to optimality within the time limit, we report a dash in the "Robust" column and use the best solution found within the time limit to compute the values for the remaining columns. (We do not report confidence intervals for the sample means, but note that the half-widths of the intervals are of the order of 10^{-2} at a confidence of 95%, which suggest that any difference of *sample means* of one or more is statistically significant).

Parameters $(m, r, D) \in \mathcal{P}_1$	5x10 (one network)					10x10 (one network)				
	Robust	Mean ($\bar{z}^{\pi}$)	SD	Min	Max	Robust	Mean ($\bar{z}^{\pi}$)	SD	Min	Max
(2,5,20)	102	104	11	88	146	119	128	17	101	227
(2,7,40)	112	106	10	92	143	130	129	16	102	229
(2,9,60)	119	109	9	95	147	137	134	16	107	222
(3,5,20)	109	104	10	89	144	128	129	16	102	223
(3,7,40)	122	109	9	96	144	143	133	16	105	222
(3,9,60)	131	109	9	95	149	152	138	17	110	218
(4,5,20)	116	105	10	90	143	137	130	16	103	220
(4,7,40)	132	110	9	96	144	155	138	16	111	217
(4,9,60)	143	112	9	98	152	165	141	16	114	217
(5,5,20)	123	106	10	90	143	145	133	16	105	213
(5,7,40)	141	110	9	96	144	165	138	16	111	217
(5,9,60)	155	112	9	99	152	-	144	16	117	218
Nominal	79	95	16	79	183	97	123	26	97	264

Table 3.5: Simulation results for the proposed model. The results for the nominal case are shown in the last row. Instances not solved to optimality within the 1-hour time limit are denoted by a dash in the "Robust" column.

The results show that when considering *worst-case* performance (the columns "Max" in the tables) the B model solutions are on average 19.15% and 11.60% more expensive than those of the proposed model for the 5×10 and 10×10 networks, respectively, and that the nominal solution is on average 25.42% and 19.90% more expensive than the proposed model solutions for the 5×10 and 10×10 networks, respectively. In other words, the solutions yield by our proposed model perform better against the worst-case scenario on average. The

Parameters $(\Gamma, D) \in \mathcal{P}_2$	5x10 (one network)					10x10 (one network)				
	Robust	Mean ($\bar{z}^\rho$)	SD	Min	Max	Robust	Mean ($\bar{z}^\rho$)	SD	Min	Max
(0.1|A|,20)	139	111	11	96	168	186	149	17	121	247
(0.1|A|,40)	182	113	10	99	167	251	154	17	126	248
(0.1|A|,60)	224	113	10	99	167	313	159	17	131	245
(0.3|A|,20)	181	120	10	106	164	264	167	17	139	246
(0.3|A|,40)	254	124	9	110	162	386	171	17	143	246
(0.3|A|,60)	327	124	9	110	162	509	171	17	143	246
(0.6|A|,20)	199	95	16	79	183	317	123	26	97	264
(0.6|A|,40)	319	95	16	79	183	-	140	18	113	234
(0.6|A|,60)	439	95	16	79	183	-	151	18	123	236
(0.9|A|,20)	199	95	16	79	183	317	123	26	97	264
(0.9|A|,40)	319	95	16	79	183	-	142	17	114	233
(0.9|A|,60)	439	95	16	79	183	-	152	17	124	239
Nominal	79	95	16	79	183	97	123	26	97	264

Table 3.6: Simulation results for the B model. The results for the nominal case are shown in the last row. Instances not solved to optimality within the 1-hour time limit are denoted by a dash in the "Robust" column.

results also show that both robust models are considerably better in terms of worst-case performance than the nominal solution. Given the set-up of the experiment, this behavior is expected as the robust models are designed to optimize worst-case performance.

Regarding *mean* performance, we have that the B model solutions are on average 1.75% cheaper and 11.66% more expensive than those of our proposed model for the 5×10 and 10×10 networks, respectively, and that the nominal solution is on average 12.08% and 8.54% cheaper than our proposed model for the 5×10 and 10×10 networks, respectively. In other words, the nominal solution is cheaper to use on average than the solutions of the robust models, while on average the solutions of both robust models do not provide a definitive insight into which can might be better in terms of average performance. Observe that this type of behavior is reasonable, as the robust models are not geared towards optimizing average performance but rather worst-case performance. On the other hand, it is to be expected that the nominal minimum cost flow is not going to be subjected to disruptions in many realizations (that is why the nominal solutions' values and their corresponding minimum values coincide), which can explain why on average the nominal solution has a better performance.

Regarding the value of the robust optimization problems, we can observe that the B model provides robust costs that are 214% and 290% more expensive than our proposed model, on average. From a pure decision-making point of view, this is rather significant. It means that the B model greatly overestimates the worst-case costs compared to our proposed model. In other words, the B model is far more conservative than the proposed model, which tends to underestimate the worst-case cost in our experiment. Note, of course, that this behavior is expected, as our model specifically addresses how the costs change due to the realization of the uncertain events, while the B model can be though as providing an "educated over conservative guess" of how the uncertainty actually affects the costs. It is important to note that the performance of the B model would be improved by decreasing the values of D; however, it is not clear *a priori* how to calibrate this parameter, even in settings where the actual values defining the disruptions are known.

Finally, consider a decision-maker faced with determining the parameters to use in the proposed model (i.e., the problem of calibrating the parameters of our proposed model in case that their actual values are not known in advance). The results in Table 3.5 show that the average worst-case performance improves as the parameters take larger values for the 10×10 case, while for the 5×5 case they are almost constant. The mean performance tends to be slightly worse as the parameters increase for both cases, while the optimal robust cost increases as the parameters increase (which is of course expected). In general, these results do not provide a clear cut pattern that can be used as a rule-of-thumb; however, the 10×10 results suggest that larger values decrease the actual worst-case performance of the cost and also provide a better estimation of the "real" worst-case cost (in our case, the one obtained via simulation).

3.7 Conclusion

We study a challenging class of robust minimum-cost flow problems in which a decision maker seeks to move flow through a network that is susceptible to disruptions, which increase the

cost of the arcs following a ripple effect. We consider two variants of the problem, the first one assumes that the damage of different disruptions on an arc is cumulative (linear model), while the second one assumes that the damage of different disruptions on a single arc is given by the maximum damage among the disruptions affecting the arc (maximum model).

A standard approach to solve our problem is a cutting-plane algorithm that requires solving a MIP subproblem at each iteration, resulting in poor computational performance. We contribute theoretical results that support the design of a polynomial-time algorithm for solving the subproblems for the linear model. We then provide complexity results that show that the subproblem for the maximum model is NP-hard. Additionally, we propose a reformulation of the subproblem that does not require big-M constants, which are commonly used by previous works from the literature. We show that (under mild assumptions) our reformulation is stronger than the standard big-M formulations. From a computational perspective, our proposed methods achieve compelling improvements in terms of both the average CPU time and the number of problem instances solved to optimality within a time limit.

CHAPTER IV

TWO-STAGE ROBUST OPTIMIZATION APPROACH FOR ENHANCED COMMUNITY RESILIENCE UNDER TORNADO HAZARDS

4.1 Motivation

Tornadoes are vertical columns of rotating air that are spawned from supercell thunderstorms caused by rotating updrafts that form because of the shear in the environmental wind field (NOAA, 2022d). Catastrophic tornadoes are common natural disasters that happen in many populated regions around the world and are particularly concerning in dense high-risk urban areas. On average, more than 1200 tornadoes happen in the US annually which cause around 60 fatalities (NOAA, 2022a). Tornadoes have caused damage costs in a range between $183 million to $9.493 billion per year in the US within the past two decades (NOAA, 2022e). The intensity of tornadoes is usually measured by the Enhanced Fujita (EF) scale which ranks a tornado from 0 (weakest) through 5 (most violent) based on an estimation of their average wind speed. The average wind speed, in turn, is estimated by comparing observed tornado damages with a list of Damage Indicators and Degrees of Damage (NOAA, 2022f).

Fatalities and injuries from tornadoes have decreased in the past few decades thanks to weather forecast-based warning systems (Standohar-Alfano et al., 2017; Koliou and van de Lindt, 2020). For example, the Storm Prediction Center at the National Weather Service keeps a Day 1-8 Convective Outlook for all the US and issues watches or warnings based on the possibility or observation of tornadoes (NOAA, 2022b). Tornado warnings are issued as

Parts of this document are reprinted with permission from submitted paper to Operations Research.

soon as a rotating supercell has been identified by a weather radar; these warnings instruct people in the affected area to take shelter immediately. Even though the warning systems have caused a great reduction in fatalities (FEMA, 2022a), their short lead time (of the order of minutes) do not allow to prevent physical structures from possible destruction. This is further aggravated by the fact that more than 80% of building stocks in the US are wood-frame buildings that are highly vulnerable to wind damage (van de Lindt and Dao, 2009).

Besides weather forecasting, there are other alternatives that can be employed to reduce the impact of tornadoes. In the past decade, several studies have shown that retrofitting strategies with simple and inexpensive actions, e.g., improving the roof cover or enhancing the roof sheathing nailing pattern of a house, can improve building codes to make wood-frame buildings more resistant to tornado damage, particularly from damaging tornadoes of EF2 intensity or less (Simmons et al., 2015; Ripberger et al., 2018; Masoomi et al., 2018; Koliou and van de Lindt, 2020; Wang et al., 2021). In particular, Ripberger et al. (2018) shows that a 30% or more reduction in lifetime damage can be expected by enhancing existing wood frame buildings with simple retrofitting actions. Similarly, recovery strategies can be employed after a tornado has hit. Recent research has considered different restoration strategies and have shown that such strategies have the potential to significantly improve the restoration time and reduce population dislocation. (Masoomi and van de Lindt, 2018; Farokhnia et al., 2020; Koliou and van de Lindt, 2020)

At US federal level, the use of retrofitting has been identified as a valuable strategy to reduce the impact of natural disasters (FEMA, 2021, 2022b; The White House, 2022). At the local level, however, such strategies have not been implemented yet. In fact, there are federal programs that provide general guidelines for local administrators (e.g., the emergency management agencies of cities and towns) to design and fund retrofitting plans FEMA (2021). In this sense, the main goal of the study is to present a model to enhance community resilience in an urban area subject to tornado hazards. Here, community resilience is defined as "the ability

to prepare for anticipated hazards, adapt to changing conditions, and withstand and recover rapidly from disruptions" (McAllister et al., 2015). In our model, resilience is enhanced by efficiently allocating resources to retrofit physical structures before a tornado, and by allocating resources to improve the recovery rate of the damaged structure after a tornado disaster.

Specifically, in this study, we assume that there is a decision-maker (a government agency or a public-private consortium; for instance the emergency management department of a city and home insurance companies) that can invest a limited budget in retrofitting and recovering structures across a geographical area of interest (e.g., a city). The objective of the decision-maker is to allocate the resources in order to maximize the community resilience of an urban area due to an uncertain tornado. We model this decision problem as a two-stage robust optimization model. The first-stage decisions are made before the realization of a tornado and determine what structures have to be retrofitted and to what extent. The second-stage decisions determine recovery strategies for the locations that are damaged by the tornado. Without loss of generality, in this study we consider total *population dislocation*, which is the involuntary movement of people from their residential sites after a severe tornado, as a metric for community resilience to be minimized (therefore, maximizing community resilience would be equivalent to minimizing population dislocation). Our proposed formulation, however, allows for other metrics that assess community resilience.

We formulate the first- and second- stage problems as integer programming problems (IP) whose decision variables select retrofitting strategies and recovery strategies, respectively, for each location. Both stages share the same budget, which models that, at least a priori, the decision-maker does not know what proportion of the budget should be spent on retrofitting and what proportion on recovery. On the other hand, as an accurate prediction of the location/time where a tornado forms, the motion direction, and its magnitude, is out of reach of the current technology (NOAA, 2022c), we approach the uncertainty using a robust lens. That is, we assume that given any retrofitting plan, the corresponding worst-case

tornado damage happens. We model tornado paths as line segments and represent the possible tornado damage with a mixed-integer non-linear uncertainty set. Consequently, the decision-maker's optimization problem is framed as a two-stage robust optimization problem with a mixed-integer non-linear uncertainty set.

In order to solve the problem, we employ an exact column-and-constraint generation (C&CG) algorithm based on the method discussed by Zeng and Zhao (2013). Here, at each iteration a master problem solves a relaxation of the original problem over a subset of possible tornado scenarios. New scenarios are iteratively generated in a subproblem and added to the master until the remaining tornado scenarios do not change the latest robust optimal value. In contrast with the classical setting of the C&CG algorithm, our subproblem is a challenging max-min non-linear integer problem, which we solve by using another decomposition algorithm. Particularly, we derive valid linear constraints for the uncertainty set, which are used as the initial constraints of the subproblem master relaxation. The feasibility of master solutions is checked using a *stabbing line algorithm* and a continuous non-convex optimization problem, as needed, and canonical conflict constraints are added to separate infeasible solutions.

Using geographical and population data from IN-CORE (2022), we use the proposed model to determine optimal retrofitting and recovery actions in Joplin, MO. In this case study, the results show that retrofitting actions should be performed in most locations in the geographical center of the city, which are also the locations with higher population density. They also show that spending budget in retrofitting is prioritized over recovery, and that only when a "critical" set of locations in the center of the city are retrofitted, there should be expenditures in recovery. By performing simulations we also show that the proposed model can significantly outperform other retrofitting policies that make retrofitting decisions at random and that the model is not over-conservative: the optimal worst-case population dislocation can be within 10% of the maximum population dislocation observed in the simulations.

To summarize, in this chapter we make the following contributions:

- We propose a novel two-stage optimization model under uncertainty to aid decision-makers in the allocation of resources to retrofit and recover residential wood-frame buildings in tornado-prone regions.

- We explicitly model tornado paths as arbitrary line segments on the plane and formulate the problem as a two-stage robust optimization problem with integrality requirements in the first and second stages and in the uncertainty set.

- We develop an exact algorithm that embeds a decomposition branch-and-cut algorithm within a column-and-constraint generation method. By exploiting the geometric properties of line segments, we develop initialization and separation procedures to effectively implement the embedded decomposition branch-and-cut algorithm.

- Using real data we provide optimal retrofitting and recovery strategies for Joplin, MO. The results show that there can be up to 20% reductions in worst-case population dislocation by investing $15 million; that our approach outperforms other retrofitting policies, and that the model does not suffer from over-conservativeness.

The remainder of this chapter is organized as follows. Section 4.2 describes the two-stage robust optimization problem. Section 4.3 presents a customized version of the C&CG algorithm. We suggest a decomposition branch-and-cut method to solve the subproblem in C&CG algorithm in Section 4.4. In Section 4.5, we conduct the numerical experiments on Joplin data as our case study. Lastly, Section 4.6 concludes this chapter.

4.2 Model formulation

Next, we define the two-stage robust optimization model to minimize the total population dislocation under an uncertain tornado and formulate the problem using mixed-integer programming.

4.2.1 Two-stage robust optimization formulation

Recall that the first stage decisions seek to determine what locations to retrofit before the realization of a tornado. The second stage decisions, which happen after the uncertainty is revealed, select what recovery strategies should be implemented in the locations affected by the tornado.

Formally, suppose S denotes the set of retrofitting strategies and P denotes the set of recovery plans. We consider a set of locations of interest L, hereafter referred to as locations for simplicity, which are placed in a 2-dimensional plane. Let $w_{\ell s}$ be the population dislocation estimation pre-tornado at location $\ell \in L$ under retrofitting strategy $s \in S$ (in most cases $w_{\ell s} = 0$) and let $g_{\ell sp}$ be the population dislocation post-tornado at location $\ell \in L$ assuming that the retrofitting strategy $s \in S$ is used and that the recovery plan is $p \in P$.

We assume that the decision-maker has a limited budget of $A \geq 0$ to invest on the retrofitting and recovery plans. Let $d_{\ell s}$ be the retrofitting cost of all the buildings in location $\ell \in L$ under strategy $s \in S$, and let $c_{\ell sp}$ denote the cost of using recovery plan $p \in P$ on the buildings in location $\ell \in L$ if the retrofitting strategy $s \in S$ is applied. Define the decision variables $f_{\ell s}$ and $r_{\ell sp}$ as

$$f_{\ell s} = \begin{cases} 1, & \text{if the buildings in location } \ell \in L \text{ are retrofitted using strategy } s \in S \\ 0, & \text{otherwise,} \end{cases}$$

$$r_{\ell sp} = \begin{cases} 1, & \text{if the buildings in } \ell \in L \text{ are retrofitted with strategy } s \in S \\ & \text{and recovered with plan } p \in P \\ 0, & \text{otherwise,} \end{cases}$$

and let z_ℓ, $\ell \in L$, be the binary variables that represent the coverage of a tornado across the

locations, that is:

$$z_\ell = \begin{cases} 1 \text{ if location } \ell \in L \text{ is affected by the tornado} \\ 0 \text{ otherwise.} \end{cases}$$

Then, the two-stage robust optimization model is defined as follows

$$v = \min \sum_{\ell \in L} \sum_{s \in S} w_{\ell s} f_{\ell s} + \max_{z \in \mathcal{U}} Q(z, f) \tag{4.1a}$$

$$\text{s.t.} \sum_{s \in S} f_{\ell s} = 1 \qquad \forall \ell \in L \tag{4.1b}$$

$$f \in \{0,1\}^{|L||S|}, \tag{4.1c}$$

where the second stage problem $Q(z, f)$ for given $z = (z_\ell : \ell \in L)$ and $f = (f_{\ell s} : \ell \in L, s \in S)$ is

$$Q(z, f) = \min \sum_{\ell \in L} z_\ell \sum_{s \in S} \sum_{p \in P} g_{\ell s p} r_{\ell s p} \tag{4.2a}$$

$$\text{s.t.} \sum_{\ell \in L} \sum_{s \in S} \sum_{p \in P} c_{\ell s p} r_{\ell s p} \leq A - \sum_{\ell \in L} \sum_{s \in S} d_{\ell s} f_{\ell s} \tag{4.2b}$$

$$\sum_{p \in P} r_{\ell s p} = f_{\ell s} \qquad \forall \ell \in L, s \in S \tag{4.2c}$$

$$r \in \{0,1\}^{|L||S||P|}. \tag{4.2d}$$

The first term of the objective function (4.1a) measures the pre-tornado dislocation across all locations due to implementing the retrofitting strategies. The second term measures the population dislocation after the worst-case tornado with respect to the retrofitting strategies selected at the first stage. Constraint (4.1b) ensures that each location picks exactly one retrofitting strategy. We make the assumption that there is a "do-nothing" strategy in S

with zero cost.

The second stage optimization problem (4.2) selects the recovery plans that minimize the population dislocation given a retrofitting strategy vector f and the tornado coverage vector z. Budget constraint (4.2b) ensures that the overall cost for the execution of recovery and retrofitting strategies is A. Constraint (4.2c) also forces each location to pick exactly one recovery strategy corresponding to the selected retrofitting strategy. Likewise, we assume the "do-nothing" strategy in P with zero cost.

In the next section, we model tornado paths by ensuring that the decision vector z belongs to the uncertainty set $\mathcal{U} \subseteq \{0,1\}^{|L|}$ which contains all feasible tornado damages over locations in L.

4.2.2 Uncertainty set formulation

We make assumptions in the following to model a tornado path based on observed characteristics of the recorded tornadoes in historical data. For example, Figure 4.1 represents the map of Oklahoma county tornadoes within 70 years in which tornado paths can be fairly seen as line segments on the map. More related data can be found in NWC (2023) where the coverage of tornadoes is characterized by the length and width of segments. Accordingly, we assume that

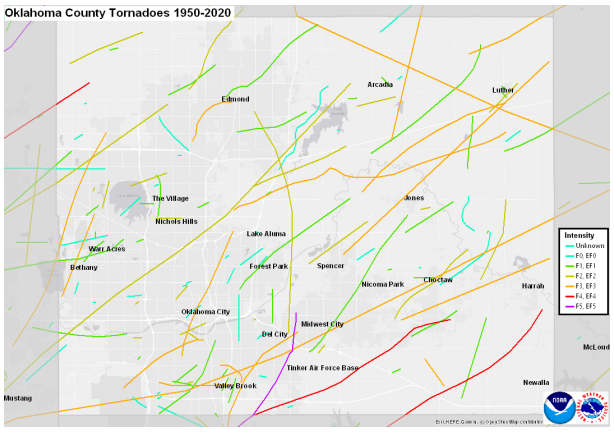


Figure 4.1: Map of Oklahoma county tornadoes between 1950-2020 (NWC, 2023).

a tornado path is specified by a line segment q on a plane and a location $\ell \in L$ is covered if it

is within a distance Δ to some point on q. We assume that the length of tornado q is at most a given parameter E, and that the two endpoints of q are located within a sufficiently large rectangle $R = [R_L^{(1)}, R_U^{(1)}] \times [R_L^{(2)}, R_U^{(2)}]$, in which all locations are located. Let $e_0 = (e_0^x, e_0^y) \in R$ and $e_1 = (e_1^x, e_1^y) \in R$ be the endpoints of line segment q (see Figure 4.2). The uncertainty set $\mathcal{U}$ can be defined by the following mixed-integer nonlinear programming (MINLP) formulation

$$\mathcal{U} = \Bigg\{ z \in \{0,1\}^{|L|} : \exists e_0, e_1 \in R, t \in \mathbb{R}^{|L|}, \tag{4.3a}$$

$$\|e_0 + t_\ell(e_1 - e_0) - (x_\ell, y_\ell)\| \leq \Delta + M_\ell(1 - z_\ell),\ \forall \ell \in L, \tag{4.3b}$$

$$0 \leq t_\ell \leq 1,\ \forall \ell \in L, \tag{4.3c}$$

$$\|e_0 - e_1\| \leq E \Bigg\}, \tag{4.3d}$$

where $\|.\|$ calculates the Euclidean distance between the vector coordinates and where (x_ℓ, y_ℓ) denote the coordinates of location $\ell \in L$ on the plane.

The value of $\|e_0 + t_\ell(e_1 - e_0) - (x_\ell, y_\ell)\|$ in Constraint (4.3b) calculates the distance between a point $e_0 + t_\ell(e_1 - e_0)$ on the line segment q and location ℓ. This constraint defines z-variables such that if $z_\ell = 1$ for some $\ell \in L$, then the location ℓ is covered by tornado since $\|e_0 + t_\ell(e_1 - e_0) - (x_\ell, y_\ell)\| \leq \Delta$. The value of M_ℓ is sufficiently large to ensure the constraint is trivially satisfied if $z_\ell = 0$ (the value of M_ℓ is set to be the largest value among the Euclidean distances from location $\ell \in L$ to the corners of rectangle R). Constraint (4.3c) enforces the range of continuous variable t_ℓ to be between 0 and 1, which indicates that q is a finite segment and not an infinite line. Constraint (4.3d) also ensures that the length of a tornado, which is the Euclidean distance between the two endpoints, does not exceed E. Note that the problem maximizes over the z-variables and that all of them have non-negative coefficients in the objective function. This observation and Constraint (4.3b) imply that $z_\ell = 1$ if and only if location $\ell \in L$ is covered by the tornado path.

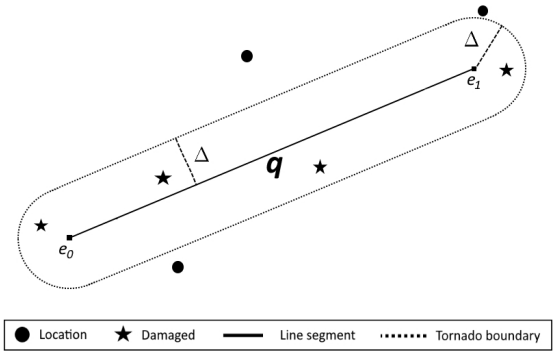


Figure 4.2: The line segment q represents a tornado central line. Locations within Δ distance of the line segment (stars) are covered by the tornado.

4.2.3 Computational complexity

We show that the robust optimization (4.1) belongs to the class of NP-hard problems, even if there is only one tornado scenario in $\mathcal{U}$. We prove the NP-hardness of the problem by a reduction from the well-known *knapsack problem* in Proposition 12. The decision version of the knapsack problem is stated that "given N items with weight $b_i, \forall i \in N$, values $q_i, \forall i \in N$, capacity B, and value Q, is there a subset $N^* \subseteq N$ such that $\sum_{i \in N^*} b_i \leq B$ and $\sum_{i \in N^*} q_i \geq Q$?". The knapsack problem is known in the class of NP-complete problems (Kellerer et al., 2004).

Proposition 12. *The optimization problem* (4.1) *is NP-hard.*

Proof. Consider an arbitrary instance of the knapsack problem with N items, weights $b_i, \forall i \in N$, values $q_i, \forall i \in N$, capacity B, and value Q. We build an instance of problem (4.1) that is equivalent to the knapsack problem. Consider an instance of problem (4.1) with $|S| = 1$ and $|P| = 2$. Observe that because $|S| = 1$, f has to be equal to a vector of ones. Suppose that the coordinates of the locations and the values of E and Δ are such that $z_\ell = 1$ for all $\ell \in L$ is feasible in $\mathcal{U}$. For simplicity, we drop index s from the notation. Also, note that $r_{\ell 2}$ can be replaced by $1 - r_{\ell 1}$ for all $\ell \in L$ because only two recovery strategies are

assumed. The specific case of the problem can be rewritten as

$$v = \min \sum_{\ell \in L} (g_{\ell 1} - g_{\ell 2}) r_{\ell 1} + \sum_{\ell \in L} (w_\ell + g_{\ell 2}) \tag{4.4a}$$

$$\text{s.t.} \ \sum_{\ell \in L} (c_{\ell 1} - c_{\ell 2}) r_{\ell 1} \leq A - \sum_{\ell \in L} (d_\ell + c_{\ell 2}) \tag{4.4b}$$

$$r_{\ell 1} \in \{0, 1\}^{|L|}. \tag{4.4c}$$

Consider the example of problem (4.4) where $L = N$, $c_{\ell 1} - c_{\ell 2} = b_\ell$ and $g_{\ell 2} - g_{\ell 1} = q_\ell$ for each item $\ell \in L$, and $A - \sum_{\ell \in L}(d_\ell + c_{\ell 2}) = B$. The answer of knapsack problem for some instance $L^* \subseteq N$ is YES (i.e., $\sum_{i \in L^*} q_i \geq Q$), if and only if $v \leq \sum_{\ell \in L}(w_\ell + g_{\ell 2}) - Q$. □

4.3 A solution method based on the C&CG framework

We present a method to solve the two-stage robust optimization problem (4.1), based on the C&CG algorithm in Zeng and Zhao (2013). First, we provide a linear one-level reformulation of problem (4.1) by means of an epigraphic reformulation of the worst-case second-stage objective. To this end, note that $\mathcal{U}$ is a finite set and write it as $\mathcal{U} = \{z^1, \ldots, z^n\}$, where $n \geq 1$ is the cardinality of $\mathcal{U}$ and suppose r^i is the vector of second-stage variables corresponding to

scenario z^i. The one-level linear mixed-integer programming (MIP) reformulation of (4.1) is:

$$v = \min \sum_{\ell \in L} \sum_{s \in S} w_{\ell s} f_{\ell s} + \theta \tag{4.5a}$$

$$\text{s.t.} \sum_{s \in S} f_{\ell s} = 1 \qquad \forall \ell \in L \tag{4.5b}$$

$$\theta \geq \sum_{\ell \in L} z^i_\ell \sum_{s \in S} \sum_{p \in P} g_{\ell s p} r^i_{\ell s p} \qquad \forall i \in \mathcal{I} \tag{4.5c}$$

$$\sum_{\ell \in L} \sum_{s \in S} \sum_{p \in P} c_{\ell s p} r^i_{\ell s p} + \sum_{\ell \in L} \sum_{s \in S} d_{\ell s} f_{\ell s} \leq A \qquad \forall i \in \mathcal{I} \tag{4.5d}$$

$$\sum_{p \in P} r^i_{\ell s p} = f_{\ell s} \qquad \forall \ell \in L, s \in S, i \in \mathcal{I} \tag{4.5e}$$

$$f \in \{0,1\}^{|L||S|} \tag{4.5f}$$

$$r^i \in \{0,1\}^{|L||S||P|} \qquad \forall i \in \mathcal{I}, \tag{4.5g}$$

where $\mathcal{I} = [n] := \{1, \ldots, n\}$. Constraint (4.5c) implies that the minimum value of θ is equal to the maximum population dislocation among all tornadoes in $\mathcal{U}$. Constraint (4.5d) and (4.5e) ensure that the recourse vectors r^i corresponding to each scenario, meet the constraints in the second stage problem.

Remark 4. *Solving problem* (4.5) *directly is not practical in general because the uncertainty set* $\mathcal{U}$ *might have exponentially many scenarios in terms of the number of locations. If* $E = \infty$*, that is, if the tornado path is a line and not a segment, there is a polynomial number of scenarios in* $\mathcal{U}$ *which can be constructed based on the stabbing line algorithm in Section 4.4.3 (in this case, nevertheless, using* (4.5) *to directly solve the problem remains impractical because there would be at least a quadratic number of scenarios in terms of the number of locations). If* $E < \infty$*, it remains an open question to determine whether there are polynomially many scenarios in* $\mathcal{U}$*.*

We employ a modified version of the C&CG to solve formulation (4.5). The algorithm is set to iteratively solve a relaxation of (4.5) over a subset $\mathcal{I}^k \subseteq \mathcal{I}$ of scenarios at iteration

k to obtain an optimal solution (θ^k, f^k). If given f^k, there exists a tornado path $z^{i_k} \in \mathcal{U}$ that produces a dislocation greater than θ^k, then we add scenario z^{i_k} to the master problem at the next iteration. Scenario i_k is found by solving the following subproblem, evaluated at f^k:

$$\Phi(f) = \max_{z \in \mathcal{U}} \min \left\{ \sum_{\ell \in L} z_\ell \sum_{s \in S} \sum_{p \in P} g_{\ell sp} r_{\ell sp} : r \in \mathcal{R}(f) \right\}, \tag{4.6}$$

where $\mathcal{R}(f)$ is the set of feasible solutions of problem $Q(z, f)$; see constraints (4.2b)–(4.2d). Specifically, if the optimal solution is such that $\Phi(f^k) > \theta^k$, then z^{i_k} is the tornado path z that attains $\Phi(f^k)$. Note that when the variables and constraints associated with scenario i_k are added at iteration $k+1$, the solution (θ^k, f^k) is no longer feasible in the master relaxation.

Let v^k be the value of the master problem at iteration k, that is, problem (4.5) with $\mathcal{I}$ replaced by $\mathcal{I}^k$, with $\mathcal{I}^k = \{i_0, i_1, \ldots, i_{k-1}\}$ and let $\Phi(f^k)$ be the subproblem for the retrofitting strategy f^k found at iteration k of the algorithm. The C&CG method is presented in Algorithm 2.

Algorithm 2: C&CG algorithm to solve v

Data: Set $\mathcal{U}, L, S$, and P

Result: f^k

Set $k = 0$, $\mathcal{I}^1 = \{i_0\} \subseteq \mathcal{U}$, $LB = -\infty$, and $UB = \infty$;

while $UB - LB > 0$ **do**

  Set $k = k + 1$;

  Solve the master MIP v^k and let f^k be the optimal retrofitting strategy. Update $LB = v^k$;

  Solve the subproblem $\Phi(f^k)$ and let i_k be the index of the z-optimal solution in $\mathcal{I}$. Update $UB = \min\{UB, \sum_{\ell \in L} \sum_{s \in S} w_{\ell s} f^k_{\ell s} + \Phi(f^k)\}$ and $\mathcal{I}^{k+1} = \mathcal{I}^k \cup \{i_k\}$;

Algorithm 2 begins with an arbitrary feasible scenario $i_0 \in \mathcal{I}$. In each iteration k, the relaxed MIP problem v^k is solved. The value of v^k then provides a lower bound for the value of v. We note that the values of v^k, $k \geq 0$, are non-decreasing in k. Then, the subproblem $\Phi(f^k)$ generates a new feasible scenario i_k for the next iteration and updates the upper bound. The algorithm terminates when the subproblem does not find a scenario that violates the

lower bound value, which implies the upper bound is equal to the lower bound. Zeng and Zhao (2013) prove that the C&CG algorithm converges in finite time.

Note that Step 5 in Algorithm 2 is very expensive computationally because it requires solving the MINLP problem $\Phi(f)$ in (4.6). The approach to solve the subproblem in standard applications of the C&CG relies on using duality or the Karush Kuhn Tucker (KKT) conditions on the second-stage problem (Zeng and Zhao, 2013). Notice that $\Phi(f)$ is a bilevel problem with integrality requirements at both levels. Therefore, we cannot use either of these approaches. Alternatively, we propose a decomposition branch-and-cut (DBC) method in the next section to solve $\Phi(f)$.

4.4 Decomposition branch-and-cut algorithm to solve $\Phi(f)$

The DBC method is based on a one-level reformulation of $\Phi(f)$. Here, we replace the min in the objective by its epigraphic reformulation and replace the uncertainty set requirements in terms of linear (although potentially exponentially many in the worst-case) constraints. Next, we discuss how to initialize the constraints in the master problem and how to perform the "two types" of separations.

4.4.1 Overview of the DBC algorithm

Suppose $\mathcal{C}$ is the set of combinations of locations that cannot be covered by a tornado path. A one-level linear reformulation of $\Phi(f)$ is given by:

$$\Phi(f) = \max \eta \tag{4.7a}$$

$$\text{s.t. } \eta \leq \sum_{\ell \in L} z_\ell \sum_{s \in S} \sum_{p \in P} g_{\ell sp} r_{\ell sp} \qquad \forall r \in \mathcal{R}(f) \tag{4.7b}$$

$$\sum_{\ell \in C} z_\ell \leq |C| - 1 \qquad \forall C \in \mathcal{C} \tag{4.7c}$$

$$z \in \{0,1\}^{|L|}. \tag{4.7d}$$

The correctness of the reformulation follows from the maximization sense in (4.7).

Observe that $\mathcal{R}(f)$ and $\mathcal{C}$ are finite sets with potentially exponentially many elements, thus $\Phi(f)$ cannot be solved directly by a commercial solver. The DBC starts with a relaxation of $\Phi(f)$ by replacing $\mathcal{R}(f)$ and $\mathcal{C}$ with subsets $\mathcal{R}^0(f)$ and $\mathcal{C}^0$, respectively. The resulting problem is referred as the *master relaxation*. The standard branch-and-cut (BC) algorithm then starts solving the master and prunes nodes by bound and infeasibility. When at some node h an integral solution z^h is found (i.e., a solution that is feasible in the master), its feasibility with respect to the original formulation in (4.7) must be verified.

To this end, Algorithm 3 takes the master feasible solution (η^h, z^h) at node h of the BC tree and verifies its feasibility: first, it checks whether z^h satisfies all $\mathcal{C}$*-constraints* (4.7c), i.e., it checks whether $z^h \in \mathcal{U}$. If z^h does not pass the first check, define $C^h = \{\ell \in L \colon z^h_\ell = 1\}$ as the *active locations* in z^h. Observe that C^h is a combination of locations that cannot be covered by a tornado path, thus a cut (4.7c) with $C = C^h$ is added on the fly. If z^h passes the first check, then the algorithm checks whether z^h satisfies all $\mathcal{R}(f)$-constraints (4.7b). If z^h also passes the second check, then z^h is feasible in $\Phi(f)$; if not, then a cut (4.7b) with $r = r^h$ is added on the fly, r^h being the optimal solution of the second-stage problem associated

with f and z^h. The performance of Algorithm 3 depends on the quality of $\mathcal{C}^0$ and $\mathcal{R}^0(f)$ and on having fast separation routines. We discuss these issues next.

Algorithm 3: Separation Procedure

Data: Master feasible solution (η^h, z^h)

Result: Generating cuts on the fly, if (η^h, z^h) is infeasible

if $z^h \notin \mathcal{U}$ **then**
  Define $C^h = \{\ell \in L : z^h_\ell = 1\}$;
  Add constraint (4.7c) with $C = C^h$ on the fly;
else if $\eta^h > Q(z^h, f)$ **then**
  Define $r^h = \arg\min\{\sum_{\ell \in L} z^h_\ell \sum_{s \in S} \sum_{p \in P} g_{\ell sp} r_{\ell sp} : r \in \mathcal{R}(f)\}$;
  Add constraint (4.7b) with $r = r^h$ on the fly;
else
  (η^h, z^h) is feasible.

4.4.2 Definition of $\mathcal{C}^0$

The set $\mathcal{C}^0$ is constructed by identifying pairs and triples of locations that cannot be covered by a single tornado path. It is worth mentioning that the valid constraints associated with these infeasible cases can also be used to tighten the relaxation of $\mathcal{U}$ by the elimination of potentially many infeasible fractional solutions for z-variables; we evaluate the performance of DBC with $\mathcal{U}$ enhanced by these valid cuts in Section 4.5.2.

Cuts from infeasible pairs:

Because the length of a tornado is at most E and only locations within a distance of Δ from a tornado line segment are covered, a tornado cannot cover two locations $\ell_1, \ell_2 \in L$ if they are at a distance of more than $E + 2\Delta$. Consequently we have the following result:

Proposition 13. *Let* $\Omega = \{(\ell_1, \ell_2) \in L^2 : \|(x_{\ell_1}, y_{\ell_1}) - (x_{\ell_2}, y_{\ell_2})\| > 2\Delta + E\}$ *be the set of*

infeasible pairs. Then any $z \in \mathcal{U}$ satisfies that

$$z_{\ell_1} + z_{\ell_2} \leq 1 \qquad \forall (\ell_1, \ell_2) \in \Omega. \tag{4.8}$$

Proof. Suppose $(\ell_1, \ell_2) \in \Omega$, and for the sake of contradiction that $z_{\ell_1} = z_{\ell_2} = 1$. For the constraint in (4.3b), we have

$$\|(1 - t_{\ell_1})e_0 + t_{\ell_1} e_1 - (x_{\ell_1}, y_{\ell_1})\| \leq \Delta \tag{4.9a}$$

$$\|(1 - t_{\ell_2})e_0 + t_{\ell_2} e_1 - (x_{\ell_2}, y_{\ell_2})\| \leq \Delta. \tag{4.9b}$$

The sum of two above inequalities results

$$\|(1 - t_{\ell_1})e_0 + t_{\ell_1} e_1 - (x_{\ell_1}, y_{\ell_1})\| + \|(x_{\ell_2}, y_{\ell_2}) - (1 - t_{\ell_2})e_0 - t_{\ell_2} e_1\| \leq 2\Delta, \tag{4.10}$$

and by applying the triangle inequality $\|X + Y\| \leq \|X\| + \|Y\|$, we conclude that

$$\|(x_{\ell_2} - x_{\ell_1}, y_{\ell_2} - y_{\ell_1}) + (t_{\ell_2} - t_{\ell_1})(e_0 - e_1)\| \leq 2\Delta. \tag{4.11}$$

Finally, we employ the fact that $\|X\| - \|Y\| \leq \|X - Y\|$ to reach

$$\|(x_{\ell_2} - x_{\ell_1}, y_{\ell_2} - y_{\ell_1})\| - \|(t_{\ell_1} - t_{\ell_2})(e_0 - e_1)\| \leq 2\Delta. \tag{4.12}$$

Now, we show (4.12) cannot be true because $(\ell_1, \ell_2) \in \Omega$, i.e.,

$$\|(x_{\ell_2} - x_{\ell_1}, y_{\ell_2} - y_{\ell_1})\| > 2\Delta + E. \tag{4.13}$$

By constraint (4.3d) and using $\|aY\| = |a|\|Y\|$, we have

$$\|(t_{\ell_1} - t_{\ell_2})(e_0 - e_1)\| = |t_{\ell_1} - t_{\ell_2}|\|e_0 - e_1\| \leq |t_1 - t_2|E \leq E. \tag{4.14}$$

Note that $|t_1 - t_2| \leq 1$. So, the two above inequalities imply that

$$\begin{cases} \|(x_{\ell_2} - x_{\ell_1}, y_{\ell_2} - y_{\ell_1})\| > 2\Delta + E \\ -\|(t_1 - t_2)(e_0 - e_1)\|_1 \geq -E \end{cases} \tag{4.15}$$
$$\implies \|(x_{\ell_2} - x_{\ell_1}, y_{\ell_2} - y_{\ell_1})\| - \|(t_{\ell_1} - t_{\ell_2})(e_0 - e_1)\| > 2\Delta.$$

Observe that (4.12) violates the true inequality (4.15). Therefore, the cut $z_{\ell_1} + z_{\ell_2} \leq 1$ is valid because $z_{\ell_1} = z_{\ell_2} = 1$ is an infeasible solution under the assumption $(\ell_1, \ell_2) \in \Omega$. □

Infeasible triple cuts:

We also identify infeasible triples of locations that cannot be covered by a tornado path. To introduce the infeasible set of triples, we assume $E = \infty$ which implies a line representation for a tornado path. Observe that if a line does not cover a subset of locations $Z \subseteq L$, then there is no segment of the line to cover the locations in Z either. For any locations ℓ_1 and ℓ_2, the analysis below partitions the plane ($\mathbb{R}^2$) in two regions, denoted by R'_{ℓ_1,ℓ_2} and R''_{ℓ_1,ℓ_2}. Region R'_{ℓ_1,ℓ_2} is the set of points $\ell \in \mathbb{R}^2$ for which there exist a line that covers ℓ_1, ℓ_2 and ℓ, and $R''_{\ell_1,\ell_2} = \mathbb{R}^2 \setminus R'_{\ell_1,\ell_2}$. For simplicity, in the next discussion we interchangeably refer to a point in $\mathbb{R}^2$ by its name, as $\ell \in \mathbb{R}^2$, or by its two-dimensional coordinates, as in $(x_\ell, y_\ell) \in \mathbb{R}^2$.

Observe that a line $\lambda \subset \mathbb{R}^2$ covers both ℓ_1 and ℓ_2 if and only if $\lambda \cap B_1 \neq \emptyset$ and $\lambda \cap B_2 \neq \emptyset$, where B_i is the ball with center ℓ_i and radius Δ, $i = 1, 2$. Let $\mathcal{L}_{\ell_1,\ell_2}$ be the collection of these lines, and let P_{ℓ_1,ℓ_2} be the set of points that belong to any such line:

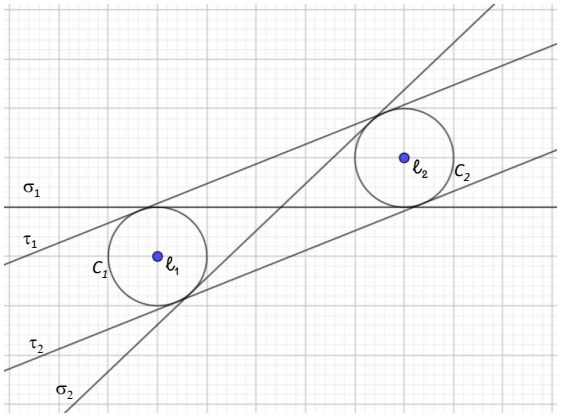
$$P_{\ell_1,\ell_2} = \{\ell \in \mathbb{R}^2 \colon \exists \lambda \in \mathcal{L}_{\ell_1,\ell_2} \text{ s.t. } \ell \in \lambda\}. \tag{4.16}$$

For any set $T \subseteq \mathbb{R}^2$ let $T(\Delta)$ be the set of points that are at a distance at most Δ from T, that is

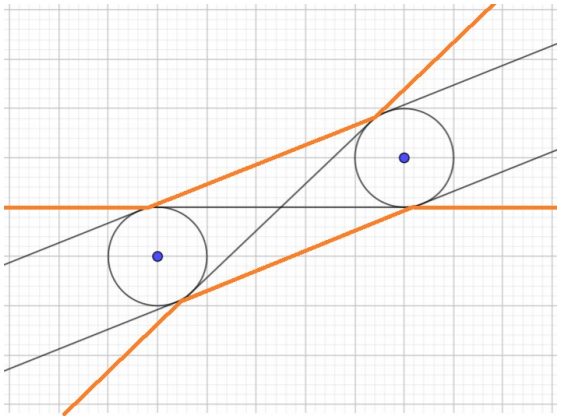
$$T(\Delta) = \{(x_\ell, y_\ell) \in \mathbb{R}^2 \colon \exists (x_t, y_t) \in T \text{ s.t. } ||(x_\ell, y_\ell) - (x_t, y_t)|| \leq \Delta\}. \tag{4.17}$$

It is readily seen that $R'_{\ell_1,\ell_2} = P_{\ell_1,\ell_2}(\Delta)$.

Now, the set P_{ℓ_1,ℓ_2} can be characterized by the four lines τ_1, τ_2, σ_1 and σ_2 that are tangent to both C_1 and C_2, where C_1 and C_2 are the circles enclosing B_1 and B_2, see Figure 4.3a. Specifically, let $\tilde{P}_{\ell_1,\ell_2}$ be the set enclosed by these tangents, see Figure 4.3b, we next show that $\tilde{P}_{\ell_1,\ell_2} = P_{\ell_1,\ell_2}$.



(a) C_1 and C_2 are circles centered at ℓ_1 and ℓ_2, respectively, with radius Δ; τ_1, τ_2, σ_1, and σ_2 are the (only) lines that are tangent to both C_1 and C_2.



(b) The region $\tilde{P}_{\ell_1,\ell_2}$ is enclosed by the orange lines.

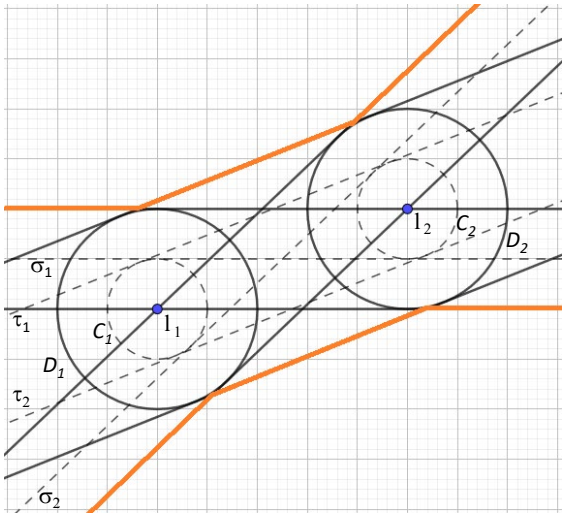
Figure 4.3: Tangent lines to circles centered at ℓ_1 and ℓ_2 with radius Δ

Indeed, let λ_0 be the line that passes through ℓ_1 and ℓ_2 and suppose that a point ℓ is in between τ_1 and τ_2. Then, the line λ that passes through ℓ and is parallel to λ_0 intersects both B_1 and B_2. Suppose that ℓ belongs to the region left of ℓ_1 between τ_2 and σ_2. Then, the line λ that passes through ℓ and the point $\sigma_2 \cap C_2$ intersects both B_1 and B_2. Notice that this argument can be repeated for the remaining regions of $\tilde{P}_{\ell_1,\ell_2}$, and thus we can conclude that $\tilde{P}_{\ell_1,\ell_2} \subseteq P_{\ell_1,\ell_2}$.

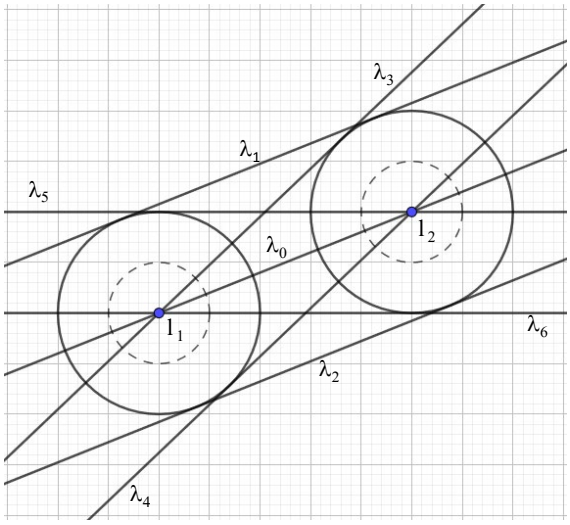
Conversely, let $\ell \in P_{\ell_1,\ell_2}$ and let $\lambda \in \mathcal{L}_{\ell_1,\ell_2}$ be the line intersecting B_1 and B_2 such that $\ell \in$

λ. If λ is one of the tangents, then it is clear that $\lambda \subseteq \tilde{P}_{\ell_1,\ell_2}$ and it follows that $\ell \in \tilde{P}_{\ell_1,\ell_2}$. Thus, assume that λ is not one of the tangents and let p_1 and p_2 be two arbitrary elements of $\lambda \cap B_1$ and $\lambda \cap B_2$. Observe that the segment s of λ between p_1 and p_2 must be completely contained in between τ_1 and τ_2. Moreover, as λ is not one of the tangents, s intersects both σ_1 and σ_2. Because two straight lines can intersect at most once, the previous observations imply that the λ cannot intersect σ_1 nor σ_2 at any point left of p_1 and right of p_2; in other words, it must be the case that λ is completely contained in $\tilde{P}_{\ell_1,\ell_2}$, which implies that $\ell \in \tilde{P}_{\ell_1,\ell_2}$, as desired.

Given the previous considerations, in order to characterize R'_{ℓ_1,ℓ_2} it is necessary to characterize $\tilde{P}_{\ell_1,\ell_2}(\Delta)$. From the shape of $\tilde{P}_{\ell_1,\ell_2}$, it is clear that $\tilde{P}_{\ell_1,\ell_2}(\Delta)$ looks very similar to $\tilde{P}_{\ell_1,\ell_2}$, with the difference that its border starts Δ units up and Δ units down, see Figure 4.4a



(a) The region $\tilde{P}_{\ell_1,\ell_2}(\Delta)$ is enclosed by the orange lines. The tangents τ_1, τ_2, σ_1, and σ_2, as well as the circles C_1 and C_2 are shown in dashed lines. The circles D_1 and D_2 centered at ℓ_1 and ℓ_2, respectively, have radius 2Δ.



(b) The lines λ_i, $i = 0, \ldots, 6$, that define the region $\tilde{P}_{\ell_1,\ell_2}(\Delta)$. The smaller circles are C_1 and C_2, the larger circles are D_1 and D_2.

Figure 4.4: The region $\tilde{P}_{\ell_1,\ell_2}(\Delta)$.

Next, we use geometric facts to derive an explicit equation for lines defining the boundaries of $\tilde{P}_{\ell_1,\ell_2}(\Delta)$ for a given ℓ_1 and ℓ_2. For ease of exposition hereafter we assume that $x_{\ell_1} \leq x_{\ell_2}$ and $y_{\ell_1} \leq y_{\ell_2}$. Define $\theta = \arctan \frac{y_{\ell_2} - y_{\ell_1}}{x_{\ell_2} - x_{\ell_1}}$ to be the angle of slope of the line λ_0 which connects

ℓ_1 and ℓ_2 with respect to the horizontal axis and let $\alpha = \arcsin \frac{2\Delta}{D(\ell_1,\ell_2)}$, where $D(\ell_1, \ell_2)$ is the Euclidean distance between ℓ_1 and ℓ_2. Then we have that $\widetilde{P}_{\ell_1,\ell_2}(\Delta) = \{(x, y) \in \mathbb{R}^2 :$ Eqs (4.18a)–(4.18f) hold$\}$, where

$$y \leq \tan\theta \cdot (x - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos\theta} \qquad (\lambda_1) \qquad (4.18a)$$

$$y \geq \tan\theta \cdot (x - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos\theta} \qquad (\lambda_2) \qquad (4.18b)$$

$$y \leq \tan(\theta + \alpha) \cdot (x - x_{\ell_1}) + y_{\ell_1} \qquad (\lambda_3) \qquad (4.18c)$$

$$y \geq \tan(\theta + \alpha) \cdot (x - x_{\ell_2}) + y_{\ell_2} \qquad (\lambda_4) \qquad (4.18d)$$

$$y \leq \tan(\theta - \alpha) \cdot (x - x_{\ell_2}) + y_{\ell_2} \qquad (\lambda_5) \qquad (4.18e)$$

$$y \geq \tan(\theta - \alpha) \cdot (x - x_{\ell_1}) + y_{\ell_1} \qquad (\lambda_6). \qquad (4.18f)$$

The lines λ_i, $i = 0, \ldots, 6$ defined in Equations (4.18a)– (4.18f) are depicted graphically in Figure 4.4b.

Lemma 3. *Lines* $\lambda_1, \ldots, \lambda_6$ *in* (4.18a)-(4.18f) *determine boundaries of* $\widetilde{P}_{\ell_1,\ell_2}(\Delta)$.

Proof. We find feasible slope m and y-intercept q of lines in the form of $y = mx + q$ with a distance Δ from both locations ℓ_1 and ℓ_2 which means the following equalities must simultaneously hold,

$$\frac{|y_{\ell_1} - mx_{\ell_1} - q|}{\sqrt{1 + m^2}} = \Delta, \qquad (4.19a)$$

$$\frac{|y_{\ell_2} - mx_{\ell_2} - q|}{\sqrt{1 + m^2}} = \Delta. \qquad (4.19b)$$

The above equalities imply two possible cases:

$$\frac{|y_{\ell_1} - mx_{\ell_1} - q|}{\sqrt{1+m^2}} = \frac{|y_{\ell_2} - mx_{\ell_2} - q|}{\sqrt{1+m^2}}$$
$$\implies \begin{cases} y_{\ell_1} - mx_{\ell_1} = y_{\ell_2} - mx_{\ell_2} \text{ (\textbf{Case 1})} \\ y_{\ell_1} - mx_{\ell_1} - q = mx_{\ell_2} + q - y_{\ell_2} \text{ (\textbf{Case 2})} \end{cases} . \tag{4.20}$$

Case 1.

$$m = \frac{y_{\ell_2} - y_{\ell_1}}{x_{\ell_2} - x_{\ell_1}} = \tan(\theta) \text{ (by the definition).} \tag{4.21}$$

By replacing $m = \tan(\theta)$ in (4.19a), we will have

$$\frac{|y_{\ell_1} - \tan(\theta)x_{\ell_1} - q|}{\sqrt{1+\tan^2(\theta)}} = \Delta \tag{4.22a}$$
$$\implies |y_{\ell_1} - \tan(\theta)x_{\ell_1} - q||\cos(\theta)| = \Delta \qquad (\sqrt{1+\tan^2(\theta)} = \frac{1}{|\cos(\theta)|}) \tag{4.22b}$$
$$\implies |y_{\ell_1} - \tan(\theta)x_{\ell_1} - q| = \frac{\Delta}{\cos(\theta)} \qquad (x_{\ell_1} \le x_{\ell_2} \implies \cos(\theta) \ge 0) \tag{4.22c}$$
$$\implies \begin{cases} q' = y_{\ell_1} - \tan(\theta)x_{\ell_1} - \frac{\Delta}{\cos(\theta)} \\ q'' = y_{\ell_1} - \tan(\theta)x_{\ell_1} + \frac{\Delta}{\cos(\theta)} \end{cases} \tag{4.22d}$$

So, the lines with formulations

$$y = \tan(\theta)(x - x_{\ell_1}) + y_{\ell_1} - \frac{\Delta}{\cos(\theta)} \qquad \text{(line } \tau_2\text{)}, \tag{4.23a}$$
$$y = \tan(\theta)(x - x_{\ell_1}) + y_{\ell_1} + \frac{\Delta}{\cos(\theta)} \qquad \text{(line } \tau_1\text{)}, \tag{4.23b}$$

are two boundaries for $\widetilde{P}_{\ell_1,\ell_2}$ that cover both locations (observe that any other q between q' and q'' ($q' \le q \le q''$) corresponding to the slope $m = \tan\theta$ forms a line path that has a distance at most Δ to the locations ℓ_1 and ℓ_2). The line in (4.23a) covers all locations in distance Δ between the lines $y = \tan\theta(x - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos\theta}$ (see (4.18b) for

line λ_2) and $y = \tan\theta(x - x_{\ell_1}) + y_{\ell_1}$. Similarly, the line (4.23b) covers all points between $y = \tan\theta(x - x_{\ell_1}) + y_{\ell_1}$ and $y = \tan\theta(x - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos\theta}$ (see (4.18a) for line λ_1).

Replacing $m = \tan\theta$ in (4.19b) will give the same boundaries as (4.19a).

Case 2.

$$q = \frac{y_{\ell_1} + y_{\ell_2} - m(x_{\ell_1} + x_{\ell_2})}{2}. \tag{4.24}$$

By replacing $q = \frac{y_{\ell_1}+y_{\ell_2}-m(x_{\ell_1}+x_{\ell_2})}{2}$ in (4.19a), we have

$$\begin{aligned} &\frac{|y_{\ell_1} - mx_{\ell_1} - \frac{y_{\ell_1}+y_{\ell_2}-m(x_{\ell_1}+x_{\ell_2})}{2}|}{\sqrt{1+m^2}} = \Delta \\ \Longrightarrow\ &\frac{|y_{\ell_2} - y_{\ell_1} - m(x_{\ell_2} - x_{\ell_1})|}{\sqrt{1+m^2}} = 2\Delta \end{aligned} \tag{4.25}$$

Instead of solving (4.25) for m which leads to a complicated quadratic equation, we exploit the form of equality to compute possible values of m. Observe that (4.25) can be interpreted as a line passing through the origin and has a distance 2Δ from the point $(x_{\ell_2} - x_{\ell_1}, y_{\ell_2} - y_{\ell_1})$ (see Figure 4.5). Now, we discuss that the angles corresponding to the possible slopes of m must be

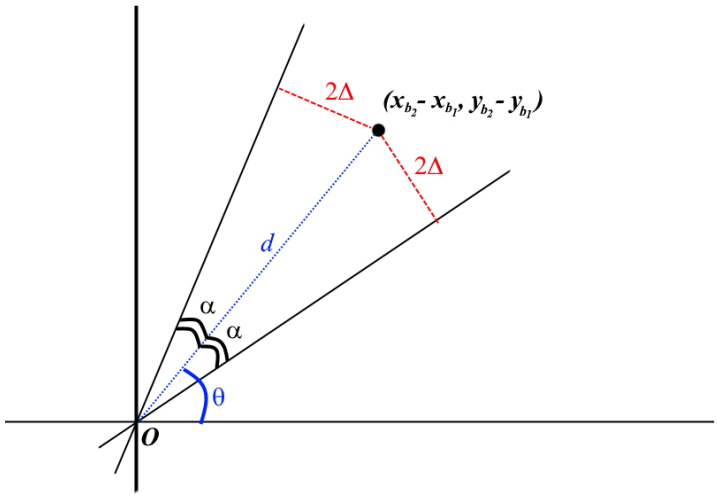


Figure 4.5: Two lines passing the origin and have a distance 2Δ from the point $(x_{\ell_2} - x_{\ell_1}, y_{\ell_2} - y_{\ell_1})$

$$m' = \tan\left(\arctan\frac{y_{\ell_2}-y_{\ell_1}}{x_{\ell_2}-x_{\ell_1}} + \arcsin\frac{2\Delta}{d}\right), \tag{4.26}$$

$$m'' = \tan\left(\arctan\frac{y_{\ell_2}-y_{\ell_1}}{x_{\ell_2}-x_{\ell_1}} - \arcsin\frac{2\Delta}{d}\right), \tag{4.27}$$

where $\arctan\frac{y_{\ell_2}-y_{\ell_1}}{x_{\ell_2}-x_{\ell_1}} = \theta$ and $\arcsin\frac{2\Delta}{D(\ell_1,\ell_2)} = \alpha$ by the definition. Therefore,

$$m' = \tan(\theta+\alpha), \tag{4.28}$$

$$m'' = \tan(\theta-\alpha). \tag{4.29}$$

So, the resulting lines are

$$\begin{aligned} y &= \tan(\theta+\alpha)x + \frac{y_{\ell_1}+y_{\ell_2}-\tan(\theta+\alpha)(x_{\ell_1}+x_{\ell_2})}{2} \\ &= \tan(\theta+\alpha)(x-\frac{x_{\ell_1}+x_{\ell_2}}{2}) + \frac{y_{\ell_1}+y_{\ell_2}}{2} \qquad \text{(line } \sigma_2\text{)}, \end{aligned} \tag{4.30}$$

$$\begin{aligned} y &= \tan(\theta-\alpha)x + \frac{y_{\ell_1}+y_{\ell_2}-\tan(\theta-\alpha)(x_{\ell_1}+x_{\ell_2})}{2} \\ &= \tan(\theta-\alpha)(x-\frac{x_{\ell_1}+x_{\ell_2}}{2}) + \frac{y_{\ell_1}+y_{\ell_2}}{2} \qquad \text{(line } \sigma_1\text{)}, \end{aligned} \tag{4.31}$$

Observe that these lines pass through the midpoint $(\frac{x_{\ell_1}+x_{\ell_2}}{2}, \frac{y_{\ell_1}+y_{\ell_2}}{2})$, see Figure 4.4a. So, two edges of the line in (4.30) with distance Δ will be $y = \tan(\theta+\alpha)(x-x_{\ell_1}) + y_{\ell_1}$ and $y = \tan(\theta+\alpha)(x-x_{\ell_2}) + y_{\ell_2}$ (see lines λ_3 and λ_4 in (4.18c) and (4.18d)). Likewise, the Δ edges of (4.31) are $y = \tan(\theta-\alpha)(x-x_{\ell_1}) + y_{\ell_1}$ and $y = \tan(\theta-\alpha)(x-x_{\ell_2}) + y_{\ell_2}$ (see lines λ_5 and λ_6 in (4.18e) and (4.18f)).

Note that replacing q in (4.19b) leads to the same equality in (4.25), and so the same conclusions. □

Now, let

$$\Gamma_{\ell_1,\ell_2} = \big\{\ell \in L \setminus \{\ell_1,\ell_2\} : (x_\ell, y_\ell) \in \mathbb{R}^2 \setminus \tilde{P}_{\ell_1,\ell_2}(\Delta)\big\}. \tag{4.32}$$

The previous discussion is summarized in the following result, which states that if a location belongs to the infeasible set Γ_{ℓ_1,ℓ_2} then at most two locations among ℓ_1, ℓ_2, and ℓ_3 can be chosen to be covered by a tornado.

Proposition 14. *Let ℓ_1 and ℓ_2 be two elements of L and suppose that $D(\ell_1, \ell_2) > 2\Delta$. Then any $z \in \mathcal{U}$ satisfies that*

$$z_{\ell_1} + z_{\ell_2} + z_{\ell} \leq 2 \qquad \forall \ell \in \Gamma_{\ell_1,\ell_2}. \tag{4.33}$$

Proof. We are interested to identify the division of space R into feasible (R'_{ℓ_1,ℓ_2}) and infeasible (R''_{ℓ_1,ℓ_2}) subspaces for two locations ℓ_1 and ℓ_2 with coordinates (x_{ℓ_1}, y_{ℓ_1}) and (x_{ℓ_2}, y_{ℓ_2}), respectively. For the sake of simplicity, we assume that $x_{\ell_1} \leq x_{\ell_2}$ throughout the proof; the results of the following mathematical work can be easily extended to the case $x_{\ell_1} > x_{\ell_2}$.

In the following, we determine the feasible pair $(\beta, q(\beta))$ for the line $y = \tan(\beta)x + q(\beta)$ to cover both locations ℓ_1 and ℓ_2 within the distance Δ, where $\beta = \arctan(m)$ is the angle of the slope of the line and $q(\beta)$ is the y-intercept of the line given angle β. It can be shown that any feasible slope angle β must be inside the interval $[\theta - \alpha, \theta + \alpha]$ as follows.

$$\frac{|y_{\ell_1} - \tan(\beta)x_{\ell_1} - q(\beta)|}{\sqrt{1 + \tan^2 \beta}} \leq \Delta, \tag{4.34a}$$

$$\frac{|y_{\ell_2} - \tan(\beta)x_{\ell_2} - q(\beta)|}{\sqrt{1 + \tan^2 \beta}} \leq \Delta. \tag{4.34b}$$

The summation of (4.34a) and (4.34b) results in

$$\frac{|-y_{\ell_1} + \tan(\beta)x_{\ell_1} + q(\beta)| + |y_{\ell_2} - \tan(\beta)x_{\ell_2} - q(\beta)|}{\sqrt{1 + \tan^2 \beta}} \leq 2\Delta \tag{4.35}$$

By the triangle inequality $\|X + Y\| \leq \|X\| + \|Y\|$, we conclude

$$\frac{|y_{\ell_2} - y_{\ell_1} - \tan(\beta)(x_{\ell_2} - x_{\ell_1})|}{\sqrt{1 + \tan^2 \beta}} \leq 2\Delta. \tag{4.36}$$

Based on Case 2 in the proof of Lemma 3, (4.36) implies that $\beta \in [\theta - \alpha, \theta + \alpha]$.

Now, we compute the feasible $q(\beta)$ for given $\beta \in [\theta - \alpha, \theta + \alpha]$. From (4.34a) and (4.34b), we have

$$y_{\ell_1} - \tan(\beta)x_{\ell_1} - \frac{\Delta}{|\cos(\beta)|} \leq q(\beta) \leq y_{\ell_1} - \tan(\beta)x_{\ell_1} + \frac{\Delta}{|\cos(\beta)|} \tag{4.37a}$$

$$y_{\ell_2} - \tan(\beta)x_{\ell_2} - \frac{\Delta}{|\cos(\beta)|} \leq q(\beta) \leq y_{\ell_2} - \tan(\beta)x_{\ell_2} + \frac{\Delta}{|\cos(\beta)|}. \tag{4.37b}$$

Note that $\sqrt{1 + \tan^2 \beta} = 1/|\cos(\beta)|$. For the sake of simplicity, we assume hereafter that $\theta + \alpha < \pi/2$ and $\theta - \alpha > -\pi/2$ (which implies $\cos(\beta) > 0$) and then discuss two cases for a feasible $q(\beta)$ followed by the inequalities (4.37a) and (4.37b):

1. $\theta \leq \beta \leq \theta + \alpha \implies \tan(\theta) \leq \tan(\beta) \implies \frac{y_{\ell_2} - y_{\ell_1}}{x_{\ell_2} - x_{\ell_1}} \leq \tan(\beta)$:

$$y_{\ell_1} - \tan(\beta)x_{\ell_1} - \frac{\Delta}{\cos(\beta)} \leq q(\beta) \leq y_{\ell_2} - \tan(\beta)x_{\ell_2} + \frac{\Delta}{\cos(\beta)}. \tag{4.38}$$

2. $\theta - \alpha \leq \beta \leq \theta \implies \tan(\theta) \geq \tan(\beta) \implies \frac{y_{\ell_2} - y_{\ell_1}}{x_{\ell_2} - x_{\ell_1}} \geq \tan(\beta)$:

$$y_{\ell_2} - \tan(\beta)x_{\ell_2} - \frac{\Delta}{\cos(\beta)} \leq q(\beta) \leq y_{\ell_1} - \tan(\beta)x_{\ell_1} + \frac{\Delta}{\cos(\beta)}. \tag{4.39}$$

Figure 4.6 visualizes an example of the feasible ranges (4.38) and (4.39) for $q(\beta)$ given angle $\beta \in [\theta - \alpha, \theta + \alpha]$.

Inequalities (4.38) and (4.39) imply line $y = \tan(\beta)x + q(\beta)$ for given $\beta \in [\theta - \alpha, \theta + \alpha]$ is *feasible* (covers ℓ_1 and ℓ_2 within a distance of Δ), if and only if a point $(\bar{x}, \bar{y})$ on the line meets

$$\begin{cases} \beta \in [\theta, \theta + \alpha] : \bar{y} \in [\tan(\beta)(\bar{x} - x_{\ell_1}) + y_{\ell_1} - \frac{\Delta}{\cos(\beta)}, \tan(\beta)(\bar{x} - x_{\ell_2}) + y_{\ell_2} + \frac{\Delta}{\cos(\beta)}], \\ \beta \in [\theta - \alpha, \theta] : \bar{y} \in [\tan(\beta)(\bar{x} - x_{\ell_2}) + y_{\ell_2} - \frac{\Delta}{\cos(\beta)}, \tan(\beta)(\bar{x} - x_{\ell_1}) + y_{\ell_1} + \frac{\Delta}{\cos(\beta)}]. \end{cases} \tag{4.40}$$

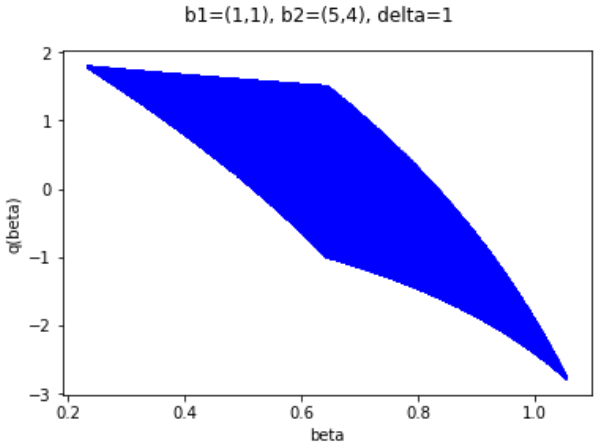


Figure 4.6: The admissible y-intercept $q(\beta)$ according to slope angle β for two locations located at $(1,1)$ and $(5,4)$ with given $\Delta = 1$. Note that $\theta = \arctan(3/4)$ and $\alpha = \arcsin(2/5)$.

Remark 5. *Given the angle $\beta \in [\theta - \alpha, \theta + \alpha]$, a feasible line $y = \tan(\beta)x + q(\beta)$ covers a point (x_0, y_0) within the distance Δ, if and only if*

$$\begin{cases} \beta \in [\theta, \theta + \alpha] : y_0 \in [\tan(\beta)(x_0 - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos(\beta)}, \tan(\beta)(x_0 - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\beta)}], \\ \beta \in [\theta - \alpha, \theta] : y_0 \in [\tan(\beta)(x_0 - x_{\ell_2}) + y_{\ell_2} - \frac{2\Delta}{\cos(\beta)}, \tan(\beta)(x_0 - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos(\beta)}]. \end{cases} \tag{4.41}$$

Regarding Remark 5, to obtain an inclusive set of feasible coverage for any line $y = \tan(\beta)x + q(\beta)$ where $\beta \in [\theta - \alpha, \theta + \alpha]$, define

$$I(x) = \bigcup_{\theta \le \beta \le \theta + \alpha} [\tan(\beta)(x - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos(\beta)}, \tan(\beta)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\beta)}],$$

and

$$J(x) = \bigcup_{\theta - \alpha \le \beta \le \theta} [\tan(\beta)(x - x_{\ell_2}) + y_{\ell_2} - \frac{2\Delta}{\cos(\beta)}, \tan(\beta)(x - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos(\beta)}].$$

A point (x, y) is covered within distance Δ of *some* feasible line if and only if $y \in I(x) \cup J(x)$.

In the following, we find the upper and lower bounds of the set $I(x) \cup J(x)$ to create the set of infeasible Γ_{ℓ_1,ℓ_2} for some location $b_3 \in L$ which $y_{b_3} \notin I(x_{b_3}) \cup J(x_{b_3}), \forall \beta \in [\theta - \alpha, \theta + \alpha]$. First, the upper bound of set $I(x)$ is discussed as follows.

Define

$$x_u^I = x_{\ell_2} - 2\Delta \left(\frac{\sqrt{1+\tan^2(\theta+\alpha)} - \sqrt{1+\tan^2(\theta)}}{\tan(\theta+\alpha) - \tan(\theta)} \right),$$

which is the intersection of two extreme lines λ_1 and λ_3 (see Lemma 3 and note that $y = \tan\theta(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos\theta}$ and $y = \tan(\theta+\alpha)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta+\alpha)}$ are other representations of λ_1 and λ_3). We claim that line λ_1 for $x \leq x_u^I$ and line λ_3 for $x \geq x_u^I$ are the upper bounds of $I(x)$.

Lemma 4. *Function* $f(z) = \frac{\sqrt{1+\tan^2 z} - \sqrt{1+\tan^2 a}}{\tan z - \tan a}$ *is strictly increasing.*

Proof. We show the derivative of $f(z)$ is strictly positive:

$$\begin{aligned}
f'(z) =& \frac{\frac{\tan z \sec^2 z}{\sqrt{1+\tan^2 z}}(\tan z - \tan a) - \sec^2 z(\sqrt{1+\tan^2 z} - \sqrt{1+\tan^2 a})}{(\tan z - \tan a)^2} \\
=& \frac{\sec^2 z}{(\sqrt{1+\tan^2 z})(\tan z - \tan a)^2} \\
& \times (\tan^2 z - \tan z \tan a - 1 - \tan^2 z + \sqrt{1 + \tan^2 z + \tan^2 a + \tan^2 z \tan^2 a}) \\
=& \frac{\sec^2 z}{(\sqrt{1+\tan^2 z})(\tan z - \tan a)^2} \\
& \times (\sqrt{(1+\tan z \tan a)^2 + \tan^2 z + \tan^2 a - 2\tan z \tan a} - \tan z \tan a - 1) \\
=& \frac{\sec^2 z}{(\sqrt{1+\tan^2 z})(\tan z - \tan a)^2} \\
& \times (\sqrt{(1+\tan z \tan a)^2 + (\tan z - \tan a)^2} - (\tan z \tan a + 1)) > 0.
\end{aligned} \tag{4.42}$$

□

Using Lemma 4, we conclude:

1. $x \leq x_u^I$:

$$\begin{aligned} x &\leq x_{\ell_2} - 2\Delta \left(\frac{\sqrt{1+\tan^2(\theta+\alpha)} - \sqrt{1+\tan^2(\theta)}}{\tan(\theta+\alpha) - \tan(\theta)} \right) \\ &\leq x_{\ell_2} - 2\Delta \left(\frac{\sqrt{1+\tan^2(\beta)} - \sqrt{1+\tan^2(\theta)}}{\tan(\beta) - \tan(\theta)} \right) \qquad \forall \beta \in [\theta, \theta+\alpha] \\ \implies \tan(\theta)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta)} &\geq \tan(\beta)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\beta)} \qquad \forall \beta \in [\theta, \theta+\alpha] \end{aligned} \tag{4.43}$$

2. $x \geq x_u^I$:

$$\begin{aligned} x &\geq x_{\ell_2} - 2\Delta \left(\frac{\sqrt{1+\tan^2(\theta+\alpha)} - \sqrt{1+\tan^2(\theta)}}{\tan(\theta+\alpha) - \tan(\theta)} \right) \\ &\geq x_{\ell_2} - 2\Delta \left(\frac{\sqrt{1+\tan^2(\theta+\alpha)} - \sqrt{1+\tan^2(\beta)}}{\tan(\theta+\alpha) - \tan(\beta)} \right) \qquad \forall \beta \in [\theta, \theta+\alpha] \\ \implies \tan(\theta+\alpha)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta+\alpha)} &\geq \tan(\beta)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\beta)} \qquad \forall \beta \in [\theta, \theta+\alpha] \end{aligned} \tag{4.44}$$

Hence, the upper bound of $I(x)$ will be (see Figure 4.7)

$$\begin{cases} \tan(\theta)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta)} \text{ (line } \lambda_1) & x \leq x_u^I \\ \tan(\theta+\alpha)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta+\alpha)} \text{ (line } \lambda_3) & x \geq x_u^I \end{cases} \tag{4.45}$$

Likewise, upper bound of $J(x)$ can be determined by defining:

$$x_u^J = x_{\ell_1} - 2\Delta \left(\frac{\sqrt{1+\tan^2(\theta)} - \sqrt{1+\tan^2(\theta-\alpha)}}{\tan(\theta) - \tan(\theta-\alpha)} \right),$$

as the intersection of lines λ_1 and λ_5. Employing Lemma 4, similarly it is concluded that

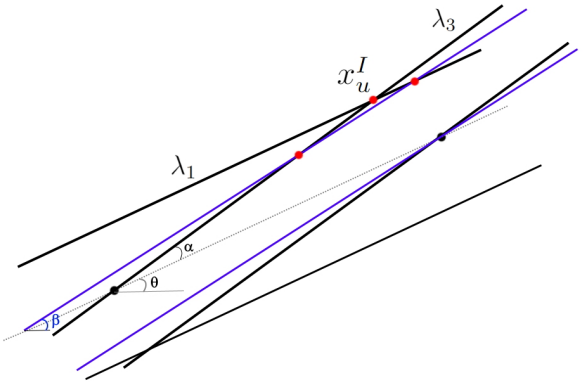


Figure 4.7: For $x \leq x_u^I$, the boundary line λ_1 and for $x \leq x_u^I$, the boundary line λ_3 restrict any given line $\tan(\beta)(x - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\beta)}, \forall \beta \in [\theta - \alpha, \theta + \alpha]$.

the upper bound of $J(x)$ will be

$$\begin{cases} \tan(\theta)(x - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos(\theta)} \text{ (line } \lambda_1) & x \geq x_u^J \\ \tan(\theta - \alpha)(x - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos(\theta - \alpha)} \text{ (line } \lambda_5) & x \leq x_u^J \end{cases} \tag{4.46}$$

The lower bounds of $I(x)$ and $J(x)$ can be determined with a similar approach. Define

$$x_l^I = x_{\ell_1} + 2\Delta \left(\frac{\sqrt{1 + \tan^2(\theta + \alpha)} - \sqrt{1 + \tan^2(\theta)}}{\tan(\theta + \alpha) - \tan(\theta)} \right),$$

and

$$x_l^J = x_{\ell_2} + 2\Delta \left(\frac{\sqrt{1 + \tan^2(\theta)} - \sqrt{1 + \tan^2(\theta - \alpha)}}{\tan(\theta) - \tan(\theta - \alpha)} \right).$$

Then, the lower bound of $I(x)$ will be

$$\begin{cases} \tan(\theta)(x - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos(\theta)} \text{ (line } \lambda_2) & x \geq x_l^I \\ \tan(\theta + \alpha)(x - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos(\theta + \alpha)} \text{ (line } \lambda_4) & x \leq x_l^I, \end{cases} \tag{4.47}$$

and for the lower bound of $J(x)$, we have

$$\begin{cases} \tan(\theta)(x - x_{\ell_2}) + y_{\ell_2} - \frac{2\Delta}{\cos(\theta)} \text{ (line } \lambda_2) & x \leq x_l^J \\ \tan(\theta - \alpha)(x - x_{\ell_2}) + y_{\ell_2} - \frac{2\Delta}{\cos(\theta-\alpha)} \text{ (line } \lambda_6) & x \geq x_l^J. \end{cases} \tag{4.48}$$

In conclusion, for given location (x_{b_3}, y_{b_3}) if it is located above the upper bounds in (4.45) and (4.46) which means

$$\begin{cases} y_{b_3} > \tan(\theta)(x_{b_3} - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos(\theta)} (= \tan(\theta)(x_{b_3} - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta)}), \\ y_{b_3} > \tan(\theta + \alpha)(x_{b_3} - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta+\alpha)} (= \tan(\theta + \alpha)(x_{b_3} - x_{\ell_1}) + y_{\ell_1}), \\ y_{b_3} > \tan(\theta - \alpha)(x_{b_3} - x_{\ell_1}) + y_{\ell_1} + \frac{2\Delta}{\cos(\theta-\alpha)} (= \tan(\theta - \alpha)(x_{b_3} - x_{\ell_2}) + y_{\ell_2}), \end{cases} \tag{4.49}$$

OR below the lower bounds in (4.47) and (4.48) which implies

$$\begin{cases} y_{b_3} < \tan(\theta)(x_{b_3} - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos(\theta)} (= \tan(\theta)(x_{b_3} - x_{\ell_2}) + y_{\ell_2} - \frac{2\Delta}{\cos(\theta)}), \\ y_{b_3} < \tan(\theta + \alpha)(x_{b_3} - x_{\ell_1}) + y_{\ell_1} - \frac{2\Delta}{\cos(\theta+\alpha)} (= \tan(\theta + \alpha)(x_{b_3} - x_{\ell_2}) + y_{\ell_2}), \\ y_{b_3} < \tan(\theta - \alpha)(x_{b_3} - x_{\ell_2}) + y_{\ell_2} + \frac{2\Delta}{\cos(\theta-\alpha)} (= \tan(\theta - \alpha)(x_{b_3} - x_{\ell_1}) + y_{\ell_1}), \end{cases} \tag{4.50}$$

then $y_{b_3} \notin I(x) \cup J(x)$. The set of infeasible locations for ℓ_1 and ℓ_2, Γ_{ℓ_1,ℓ_2}, includes locations in L that meet (4.49) or (4.50), i.e.,

$$\begin{aligned} \Gamma_{\ell_1,\ell_2} &= \{b_i \in L \setminus \{\ell_1, \ell_2\} : y_{b_i} \notin I(x_{b_i}) \cup J(x_{b_j})\} \\ &= \{b_i \in L \setminus \{\ell_1, \ell_2\} : (x_{b_i}, y_{b_i}) \in R''_{\ell_1,\ell_2}\}. \end{aligned} \tag{4.51}$$

Therefore, locations in Γ_{ℓ_1,ℓ_2} cannot be covered by a line that has a distance at most Δ from two locations ℓ_1 and ℓ_2, and so $z_{\ell_1} + z_{\ell_2} + z_{b_3} \leq 1$ for $b_3 \in \Gamma_{\ell_1,\ell_2}$ is a valid cut.

Remark 6. *The boundaries of the cases $\theta+\alpha \geq \pi/2$ or $\theta-\alpha \leq -\pi/2$ can be easily obtained by rotating coordinates to get θ' such that $\theta'+\alpha < \pi/2$ and $\theta'-\alpha > -\pi/2$ the same as the assumptions in the proof.*

□

It is worth mentioning that we do not need to consider all possible combinations of locations to construct the infeasible triple cuts because if $\ell_3 \in \Gamma_{\ell_1,\ell_2}$, then $\ell_1 \in \Gamma_{\ell_2,\ell_3}$ and $\ell_2 \in \Gamma_{\ell_1,\ell_3}$. Also, we note that the pair-wise and triple valid constraints introduced so far do not completely characterize $\mathcal{U}$, as shown in the next remark. However, they can be used to define the initial subset $\mathcal{C}^0 \subseteq \mathcal{C}$ of the master relaxation in the DBC as follows:

$$\mathcal{C}^0 = \big\{(\ell_1,\ell_2) \in \Omega\big\} \cup \big\{(\ell_1,\ell_2,\ell_3) : \ell_3 \in \Gamma_{\ell_1,\ell_2}, \ell_1,\ell_2 \in L, D(\ell_1,\ell_2) > 2\Delta\big\}. \tag{4.52}$$

Remark 7. *Define $\mathcal{V} = \{z \in \{0,1\}^{|L|} : \sum_{\ell \in C} z_\ell \leq |C|-1, \forall C \in \mathcal{C}^0\}$. The set $\mathcal{V}$ does not necessarily determine all the feasible solutions in the uncertainty set $\mathcal{U}$; specifically $\mathcal{U} \subset \mathcal{V}$ and in general there exist points in $\mathcal{V}$ that do not belong to $\mathcal{U}$. For example, consider a case of three locations ℓ_1, ℓ_2, and ℓ_3 with coordinates $(0,0), (4,0)$, and $(2,1.1)$ on a plane under a tornado with the maximum length of $E = 2$ and the width of $\Delta = 1$ (See Figure 4.8). It can be easily observed that there exists no tornado path that covers all three locations because the only tornado path covering ℓ_1 and ℓ_2 is the line segment with endpoints $(1,0)$ and $(3,0)$. Obviously, the location $(2,1.1)$ is not covered by this path. However, $z_{\ell_1} = z_{\ell_2} = z_{\ell_3} = 1$ is a feasible solution in $\mathcal{V}$ because it satisfies both the infeasible pair and triple cuts.*

4.4.3 Feasibility check in $\mathcal{C}$

Next, we present a method to determine whether a master feasible solution z^h belongs to $\mathcal{U}$. We start our analysis for the case when $E = \infty$ and then present a method for the case when $E < \infty$.

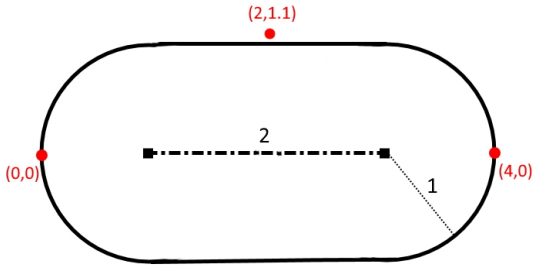


Figure 4.8: The only possible tornado path covering buildings $(0, 0)$ and $(4, 0)$.

Feasibility check for $E = \infty$ using the SLA:

The stabbing line problem consists of a collection of (disjoint) circles $\mathcal{O} = \{O_1, O_2, \ldots, O_n\}$ and the objective is to find the line on the plane that intersects the most circles. The SLA is an algorithm that solves this problem and reports $c(\mathcal{O})$: the maximum number of circles in $\mathcal{O}$ that can be intersected with a line. Consider the collection of circles $\mathcal{O}^h$ with centers at the locations in C^h and radius Δ. Observe that a tornado path with $E = \infty$ covers all locations in C^h if and only if the $c(\mathcal{O}^h) = |C^h|$ (see Figure 4.9). Therefore, if $c(\mathcal{O}^h) < |C^h|$, then $z^h \notin \mathcal{U}$ whereas if $c(\mathcal{O}^h) = |C^h|$ then $z^h \in \mathcal{U}$.

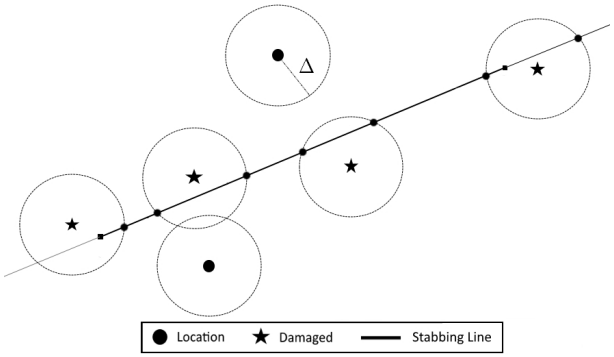


Figure 4.9: Finding the locations that are covered by a tornado within distance Δ is equivalent to finding a line that stabs the circles centered at the locations with radius Δ.

We explain next the SLA following Saba (2013). Assume first a set of disjoint circles $\mathcal{O} = \{O_1, O_2, \ldots, O_n\}$. Given any two disjoint circles O_a and O_b, find the two external and two internal tangent lines to each pair of circles (O_a, O_b) and record the intersection points on both circles; see Figure 4.10. Let p^E_{ab}, q^E_{ab} and p^I_{ab}, q^I_{ab} be the intersection points of these

external and internal tangent lines, respectively, with O_a. Note that a tangent line to O_a that touches any point p in the arc $[p^E_{ab}, p^I_{ab}]$ (or $[q^I_{ab}, q^E_{ab}]$) intersects O_b, see Figure 4.10.

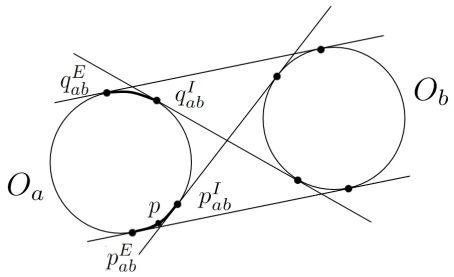


Figure 4.10: Four tangent lines and the intersection points for a pair of circles (O_a, O_b) are shown. A tangent line at any point p on the arc $[p^E_{ab}, p^I_{ab}]$ also intersects O_b.

Given O_a, the SLA computes the arc-intervals $[p^E_{ab}, p^I_{ab}]$ and $[q^I_{ab}, q^E_{ab}]$ for all $b \neq a$; note there exist exactly $2(n-1)$ such intervals for O_a. Let π_a be an endpoint of any such arc-intervals that belong to the maximum amount of arc-intervals. Then, the tangent line that touches O_a at π_a is a line intersecting O_a that intersects the most circles in $\mathcal{O}$. Iterating this process starting from each $a = 1, \ldots, n$, will determine a line that intersects the most circles. The SLA can be implemented $O(n^2 \log n)$ time; see Saba (2013) for further details on the proof of correctness and complexity of the algorithm.

Note that in our case the circles in $\mathcal{O}^h$ might be non-intersecting, as the locations in C^h might be at a distance less than 2Δ. We thus adapt the SLA, noting that if $O_a \cap O_b \neq \emptyset$ then internal tangents are not necessary and that doing the analysis with the arc-interval $[p^E_{ab}, q^E_{ab}]$ is sufficient for this case.

Feasibility check if $E < \infty$:

In this case, if $c(\mathcal{O}^h) < |C^h|$, then we can again conclude that there is no finite line segment that covers C^h and thus $z^h \notin \mathcal{U}$. Otherwise, if $c(\mathcal{O}^h) = |C^h|$, then there is an infinite line that covers all locations in C^h, but there is no guarantee that a line segment of *length at most* E covers all locations of C^h. Thus, we complete the feasibility check in two steps: first, we use the

line resulting from the SLA and projection ideas in the following Lemmas 5 and 6 to determine whether the line can be trimmed to be of length E without compromising its coverage. If even after this step the feasibility of z^h cannot be guaranteed, then we check the feasibility of z^h algebraically using a simplified version of $\mathcal{U}$. Specifically, consider the set $\tilde{\mathcal{U}}^h$ defined as

$$\tilde{\mathcal{U}}^h = \Big\{(e_0, e_1) \in \mathbb{R}^2 : \exists t \in [0,1]^{|C^h|}, \|e_0 - e_1\| \leq E \tag{4.53a}$$

$$\|e_0 + t_\ell(e_1 - e_0) - (x_\ell, y_\ell)\| \leq \Delta,\ \forall \ell \in C^h\Big\}. \tag{4.53b}$$

Observe that $z^h \in \mathcal{U}$ if and only if $\tilde{\mathcal{U}}^h \neq \emptyset$ and that (4.53) is far simpler than $\mathcal{U}$ because it has fewer variables and does not involve binary variables nor 'big-M' values. Checking whether $\tilde{\mathcal{U}}^h$ is non-empty can be done using a quadratic (non-convex) continuous optimization solver.

To elaborate on the feasibility check procedure in case $E < \infty$, we first provide the following definitions. Let $C \subseteq L$ be a subset of locations in L and let λ be a line in $\mathbb{R}^2$. We say that λ is a *covering line* for C if the distance from λ to any $\ell \in C$ is at most Δ. Similarly, we will refer to a finite segment ξ that covers all locations in C a *covering segment* for C; the length of segment ξ will be denoted by u_ξ. For any $a, b \in L$, let λ_a and λ_b the projections of a and b in λ, respectively.

Lemma 5. *Let $C \subseteq L$ be given and let λ be a covering line for C. If $D(\lambda_a, \lambda_b) \leq E$ for all $a, b \in C$, then there exists a covering segment ξ of C such that $u_\xi \leq E$.*

Proof. Let $w, w' \in \operatorname{argmax}\{D(v, v') \colon v = \lambda_a, v' = \lambda_b, a, b \in C\}$. That is, w and w' are the projections of the points in C on λ that attain the maximum distance. If $\xi = [w, w']$, then clearly ξ covers C and $u_\xi \leq E$. □

Let $C \subseteq L$ be given and let λ be a covering line of C. We denote by $C_{\lambda,E} \subseteq C$ the set

of building pairs in C whose projections in λ are at a distance of more than E, i.e.,

$$C_{\lambda,E} = \big\{(a,b) \in C \times C \colon D(\lambda_a, \lambda_b) > E\big\}. \tag{4.54}$$

For any $\ell \in L$ let $\delta_\ell = D(\ell, \lambda_\ell)$ and let $\xi_\ell = \sqrt{\Delta^2 - \delta_\ell}$. For any $(a,b) \in C_{\lambda,E}$ let $\lambda'_a = \lambda_a + \xi_a(\lambda_b - \lambda_a)$ and $\lambda'_b = \lambda_b + \xi_b(\lambda_a - \lambda_b)$. Note that both λ'_a and λ'_b are points in λ.

Lemma 6. *Let $C \subseteq B$ be given and let λ be a covering line for C. If $D(\lambda'_a, \lambda'_b) \leq E$ for all $a, b \in C_{\lambda,E}$ then there exist a covering segment ξ of C such that $u_\xi \leq E$.*

Proof. By construction, for any $(a,b) \in C_{\lambda,E}$, both λ'_a and λ'_b are elements of λ and $D(\lambda'_a, a) = D(\lambda'_b, b) = \Delta$, thus the segment $[\lambda'_a, \lambda'_b]$ covers both a and b. Let $w, w' \in \operatorname{argmax}\{D(\lambda'_a, \lambda'_b) \colon a, b \in C_{\lambda,E}\}$, then by definition $D(\lambda'_a, \lambda'_b) \leq D(w, w')$ for all $a, b \in C_{\lambda,E}$ and, moreover, $[w, w']$ covers C. Because $D(w, w') \leq E$ then $\xi = [w, w']$ is a covering segment of C with $u_\xi \leq E$. □

Step 1 in Algorithm 3 which checks the feasibility of solution z^h in $\mathcal{U}$, can be reframed as finding two endpoints e_0 and e_1 of some covering segment ξ for C^h with $u_\xi \leq E$. To accelerate the search for two endpoints in step 1, we implement SLA for C^h along with two lemmas 5 and 6 as shown in Algorithm 4.

Algorithm 4 executes SLA for the set of circles with centers in C^h and radius Δ (step 2). If the maximal stabbing line λ_{MAX} does not cover locations in C^h, then no segment produces the exiting solution which means z^h is infeasible. Else, using the discussion in Lemmas 5 and 6, we check whether the shortest covering segment of λ_{MAX}, denoted as ξ_{MAX}, has a length at most E (step 4). If so, we recognize z^h feasible by ξ_{MAX}. If the length of the segment is greater than E, we directly place z^h in $\mathcal{U}$ that gives set $\tilde{\mathcal{U}}^h$. We solve the quadratic set $\tilde{\mathcal{U}}^h$ for a pair (e_0, e_1). If set $\tilde{\mathcal{U}}^h$ is not empty, then there are feasible endpoints for a segment covering C^h (step 7). Furthermore, we enhance the performance of step 7 by restricting the

Algorithm 4: Feasibility check for solution z^h

Data: Set $\tilde{\mathcal{U}}^h, C^h, E$
Result: $feasible$

```
set feasible = FALSE;
run SLA for locations in C^h. Let λ_MAX be the line intersecting maximal number of
  circles and ξ_MAX be the shortest covering segment;
if maximum coverage with λ_MAX = |C^h| then
    if u_{ξ_MAX} ≤ E then
        feasible = TRUE
    else
        if Ũ^h ≠ ∅ then
            feasible = TRUE
```

searching ranges for feasible endpoints as follows.

Let ℓ_i and ℓ_j be two extreme locations that have the maximum distance among all pairs of locations in C^h. Then, the endpoints of the line segment e_0 and e_1 must be within distance Δ from each ℓ_i and ℓ_j. Particularly, points e_0 and e_1 should satisfy that (see Figure 4.11):

$$x_{\ell_i} - \Delta \leq e_0^x \leq x_{\ell_i} + \Delta \tag{4.55a}$$

$$y_{\ell_i} - \Delta \leq e_0^y \leq y_{\ell_i} + \Delta \tag{4.55b}$$

$$x_{\ell_j} - \Delta \leq e_1^x \leq x_{\ell_j} + \Delta \tag{4.55c}$$

$$y_{\ell_j} - \Delta \leq e_1^y \leq y_{\ell_j} + \Delta. \tag{4.55d}$$

4.4.4 Definition of $\mathcal{R}^0(f)$ and feasibility check for $\mathcal{R}(f)$

To initialize the master problem we assume that $\mathcal{R}^0(f) = \{r^0\}$ where r^0 is an optimal solution for the second-stage problem $Q(z, f)$ assuming no location is hit, i.e., r^0 is a solution of $Q(\mathbf{0}, f)$. On the other hand, once it has been checked that $z^h \in \mathcal{U}$, we check that z^h satisfies all $\mathcal{R}(f)$-constraints in (4.7b) by solving the second-stage problem evaluated at z^h, i.e., by

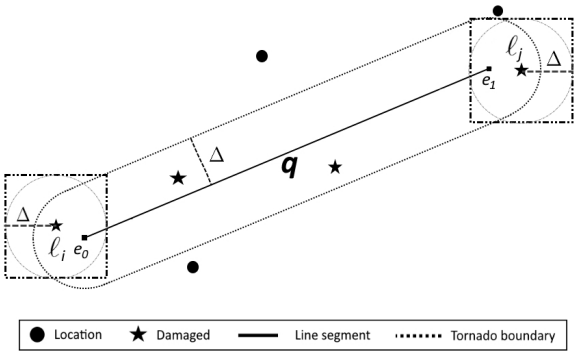


Figure 4.11: Two endpoints e_0 and e_1 are within circles centered at locations ℓ_i and ℓ_j with radius Δ. These two circles are inside squares which are added to restrict the search ranges for endpoints by linear constraints.

solving $Q(z^h, f)$. Let r^h be an optimal solution of $Q(z^h, f)$. If $\eta^h > Q(z^h, f)$ then (η^h, z^h) is not feasible in (4.7b) and we add the constraint $\eta \leq \sum_{\ell \in L} z_\ell \sum_{s \in S} \sum_{p \in P} g_{\ell sp} r^h_{\ell sp}$ at all active nodes of the BC tree. Conversely, if $\eta^h \leq Q(z^f, f)$, then (η^h, z^h) is a feasible solution of (4.7) and node h is pruned by feasibility after updating the lower bound of the BC. Observe that the initialization and separation procedure involve solving the NP-hard problem $Q(z, f)$; however, this problem is relatively simple and can be solved quickly using commercial MIP solvers, see our computational results in Section 4.5.

4.5 Case study: retrofitting/recovery residential buildings in Joplin, MO

Next, we perform several numerical experiments with the proposed model. All algorithms are coded in Python 3.9 and solved using Gurobi Optimizer 9.1.1 on a 64-bit Windows operating system with Intel(R) Core(TM) i7-8550U CPU and 8.00 GB of RAM under a time limit of 3600 seconds. Our objectives are to study the performance of the two-stage optimization problem, to analyze how the optimal solutions change with key parameters, to gather high-level insights into the optimal retrofitting and recovery activities, and to compare the optimal policies with other benchmark policies. The experiments are based on location and cost data from Joplin, MO. Joplin was the location of one of the most severe tornadoes in US history that occurred

on May 22, 2011. The tornado was rated EF-5 with an average speed of more than 200 mph and crossed 22.1 miles on the ground with a width of one mile. This catastrophe caused the loss of 161 lives and more than a thousand injuries. The total damage was approximated at 3 billion dollars as around 7500 residential structures were destroyed (Kuligowski et al., 2014).

4.5.1 Definition of parameters

The geographic and demographic data are based on information from Joplin and extracted from the online platform IN-CORE (2022). Particularly, this dataset divides buildings in about 1500 "block IDs" (which are contiguous blocks of the city) and includes over 20,000 buildings. For our experiments, we use only the single or multi-story residential wood-frame building archetypes.

We assume an EF-2 tornado with a wind speed of 134 mph, a width of 0.75 miles (thus Δ=0.75/2), and a length of 5 miles. We assume that the first stage population dislocation parameter is zero for all locations and strategies, that is, $w_{\ell s} = 0$ for all $\ell \in L, s \in S$. Following Wang et al. (2021), we consider the retrofitting strategies presented in Table 4.1. Masoomi et al. (2018) analyze the effect of the strategies in Table 4.1 on the component fragility of wood-frame residential building archetypes.

Strategy s	Description
R_1	Roof covering with asphalt shingles, using 8d nails to attach roof deck sheathing panel to rafters spaced at 6/12 inch, and the selection of two 16d toenails for roof-to-wall
R_2	Roof covering with asphalt shingles, using 8d nails to attach roof deck sheathing panel to rafters spaced at 6/6 inch, and the selection of two H2.5 clips for roof-to-wall
R_3	Roof covering with clay tiles, using 8d nails to attach roof deck sheathing panel to rafters spaced at 6/6 inch, and the selection of two H2.5 clips for roof-to-wall.

Table 4.1: Retrofitting strategies for the experiments.

The cost of implementing these strategies was computed from the data of IN-CORE (2022). As mentioned in Section 4.2, a 'do-nothing' strategy with zero cost is also included in the retrofitting strategy set S, that is, S={do-nothing, R1, R2, R3}. On the other hand,

only two recovery strategies are assumed in our experiment, namely, recover ($p = 1$) and do-nothing ($p = 0$); thus $P = \{0, 1\}$.

The second stage population dislocation parameters, $g_{\ell sp}$, are defined as the expected population dislocation at location $\ell \in L$, 60 days after the tornado. The value of $g_{\ell s1}$ is evaluated based on the state of damage $d \in D$ immediately after the tornado hits, where D is the set of damage states. Let X_ℓ be the random variable denoting the functionality of ℓ after 60 days; where $X_\ell = 0$, only if ℓ reaches 100% functionality after 60 days (no dislocation) and $X_\ell = 1$ otherwise (dislocation). Also, let $Y_\ell \in D$ be the random variable representing the damage at ℓ immediately after the tornado, and let σ_ℓ denote the retrofitting strategy. Then, the expected population dislocation after recovery is

$$g_{\ell s1} = N_\ell \times E[X_\ell | \sigma_\ell = s], \tag{4.56}$$

where N_ℓ is the population of location $\ell \in L$. Since the value of X_ℓ depends on Y_ℓ, using conditional probability and the (reasonable) assumption that X_ℓ is independent of σ_ℓ given the knowledge of Y_ℓ, we have that

$$E[X_\ell | \sigma_\ell = s] = \sum_{d \in D} P[X_\ell = 1 | Y_\ell = d] P[Y_\ell = d | \sigma_\ell = s]. \tag{4.57}$$

Following Koliou and van de Lindt (2020), the damage states are given by $D = \{$Minor, Moderate, Extensive, Complete$\}$. We compute $P[X_\ell = u | Y_\ell = d]$ using the repair time distributions described in Hazus (2003). These distributions are lognormal and their parameters can be found in Koliou and van de Lindt (2020). The probabilities $P[Y_\ell = d | \sigma_\ell = s]$ are computed from the building-level tornado fragility-curves for wood-frame residential buildings in Masoomi et al. (2018); Koliou and van de Lindt (2020); Wen et al. (2021).

The cost of recovery $c_{\ell s1}$ is also the expected percentage of the total location replace-

ment cost: Suppose R_ℓ is the area of location $\ell \in L$ and α is the cost of replacement per m^2. For the wood residential archetype, we set $\alpha = \$862.0/m^2$ and, the percentage of the location replacement cost to be $r_{minor} = 0.5\%$, $r_{moderate} = 2.3\%$, $r_{extensive} = 11.7\%$, and $r_{complete} = 23.4\%$, see Koliou and van de Lindt (2020). The following formula is used to compute the cost of recovery for location ℓ if it is retrofitted by strategy s:

$$c_{\ell s1} = \alpha \times R_\ell \times \sum_{d \in D} r_d P[Y = d | \sigma = s]. \tag{4.58}$$

We assume the decision-maker takes charge of all recovery expenses.

On the other hand, if a location does not receive resources from the decision-maker, then the population dislocation will be higher and depends on the resident's actions. We assume the population dislocation for the do-nothing strategy is a factor of the population dislocation resulting from the recovery actions of the decision-maker, i.e.,

$$g_{\ell s0} = \mu_{\ell s} g_{\ell s1} \quad \forall \ell \in L, s \in S. \tag{4.59}$$

The factor $\mu_{\ell s}$ can vary in a range such that $g_{\ell s0} \in [g_{\ell s1}, N_\ell]$: $g_{\ell s0} = g_{\ell s1}$ implies that the residents pay a full amount of expenses by themselves to recover their buildings as quick as the decision-maker, and $g_{\ell s0} = N_\ell$ means that the residents take no recovery action on their own and therefore the whole population is dislocated from location ℓ. For our experiments, we fix $g_{\ell s0} = (g_{\ell s1} + N_\ell)/2$ which is the midpoint in the range. Note that the cost of the do-nothing strategy is $c_{\ell s0} = 0, \forall \ell \in L, s \in S$ from the decision-maker's point of view.

4.5.2 Performance of subproblem methods

We compare three different methods to address the one-level subproblem $\Phi(f)$ in Algorithm 1:

1. ORG: solves the subproblem with *original* set $\mathcal{U}$ (instead of using (4.7c)) by DBC and

only updates $\mathcal{R}(f)$ on the fly,

2. AVC: *adds valid cuts* for infeasible combinations in $\mathcal{C}^0$ to $\mathcal{U}$ to enhance ORG performance while updates $\mathcal{R}(f)$ on the fly,

3. DEC: replace $\mathcal{U}$ by conflict constraints (4.7c) and update $\mathcal{C}$ and $\mathcal{R}(f)$ on the fly as described in Section 4.4.1.

Table 4.2 compares the three approaches in terms of optimality gap, solution time, number of iterations, subproblem run time, and its callback function run time. We generate a testbed of 5 sample problems in which 10 blocks in Joplin are randomly selected as locations. Column 2 shows the method that is implemented to solve the subproblem. Columns 3, 4, and 5 present the best objective value (expected population dislocation), the best lower bound, and the optimality gap found within the time limit of one hour, respectively. Columns 6 and 7 show the CPU run time of C&CG Algorithm 1 in seconds and its number of iterations, respectively. Column 8 compares the CPU run time of the DBC method in seconds for each selection of subproblems. Column 9 also presents the total amount of time that the DBC spends in the callback function to generate cuts on the fly with Algorithm 3; note that the callback run time for ORG and AVC methods does not include the time of checking feasibility in $\mathcal{U}$ as we directly use the original feasible solution.

Sample #	Subproblem method	Best objective	Best bound	Gap	CCG run time (sec.)	CCG iteration	DBC run time (sec.)	Callbacks run time (sec.)
1	ORG	365	-	-	*3600*	1	3600	42
	AVC	185	185	0%	6	3	6	0
	DEC	185	185	0%	1	3	1	1
2	ORG	161	153	5%	*4012*	2	4012	1
	AVC	153	153	0%	26	3	26	1
	DEC	153	153	0%	1	3	1	0
3	ORG	178	178	0%	*4484*	3	4484	1
	AVC	178	178	0%	9	3	9	0
	DEC	178	178	0%	2	3	1	1
4	ORG	56	-	-	*3600*	1	3600	0
	AVC	75	75	0%	3549	3	3549	1
	DEC	75	75	0%	1	3	1	1
5	ORG	103	-	-	*3600*	1	3600	0
	AVC	65	65	0%	36	3	36	1
	DEC	65	65	0%	1	3	1	0

Table 4.2: Comparing C&CG performance for 3 different methods to solve the subproblem

The results in Table 4.2 show that the ORG method is not able to solve any sample with

only 10 locations within the one-hour time limit. Also, based on the results, solving the master problem and the second stage recovery problem in the callback function is very fast and almost all the run time in ORG is spent on solving the MINLP subproblem (4.6). For samples 1, 4, and 5 the subproblem was not solved to optimality in one hour. For some samples, the ORG method exceeds the time limit while solving the MINLP subproblem. In these cases, we wait until the subproblem finishes its execution before terminating the algorithm, which results in some computational times reported being larger than the one-hour time limit (*italic values* in the table).

Adding infeasible pairs and triples cuts in AVC significantly reduces the computation times by as much as two orders of magnitude, resulting in all the samples being solved in less than one hour. However, AVC can be potentially time-consuming for problems such as sample 4 in the experiment. The DEC method greatly outperforms the other two methods as it solved all samples within a few seconds. The subproblem runtime for DEC shows how fast the subproblem can lead to finding optimal tornado paths. The callbacks' runtime also emphasizes that the feasibility check with Algorithm 4 in Section 4.4.3 which is equipped with the polynomial stabbing line algorithm is a quick task.

4.5.3 Robust retrofitting and recovery strategies in Joplin, MO

We implement the proposed two-stage optimization model to find robust retrofitting and recovery strategies for locations in Joplin. We use Algorithm 2, along with the DBC method described in Section 4.4 with the DEC setup that yields the best performance regarding comparative experiments in Table 4.2.

In order to define the locations in L, we group Joplin "block IDs" in 100 center locations using the k-means method (MacQueen, 1967) and aggregate the data accordingly. Particularly, we assume that when the tornado hits a location then all of its buildings are hit, and that the tornado hits a location only if the location's centroid is within Δ units of the path. Table 4.3 compares the population dislocation for when the tornado length is $E = 5$ miles or ∞ and the

budgets are $A = 0, 15, 30$ million USD. Columns 1 and 2 in Table 4.3 show E and A. Column 3 has the optimal population dislocation under the worst-case tornado for each set of parameters. Columns 4 and 5 report the CPU run time of C&CG Algorithm 2 in seconds and its number of iterations, respectively. Column 6 shows the CPU runtime of the DBC method in seconds to solve subproblem $\Phi(f)$. Column 7 compares the CPU time that the DBC algorithm spends in a callback function to verify the feasibility of integral solutions. Column 8 shows the total CPU time to investigate set $\tilde{\mathcal{U}}^h$ for a feasible solution. Columns 9 and 10 present the proportion of the total budget that is spent for retrofitting and recovery purposes, respectively.

Length	Budget ($)	Population dislocation	CCG run time (sec.)	CCG iteration	DBC run time (sec.)	Callback run time (sec.)	$\tilde{\mathcal{U}}^h$ feasibility run time (sec.)	Retrofitting cost /budget	Recovery cost /budget
5	0 M	16318	127	2	126	3	1	-	-
	15 M	13408	307	3	306	11	6	0.6	0.4
	30 M	13091	300	3	300	11	5	0.3	0.7
∞	0 M	17293	568	2	568	3	0	-	-
	15 M	14236	716	2	716	5	0	0.6	0.4
	30 M	13839	986	3	985	5	0	0.4	0.6

Table 4.3: Solving the two-stage robust optimization model with different parameters of tornado length and available budget for 100 locations group in Joplin, MO.

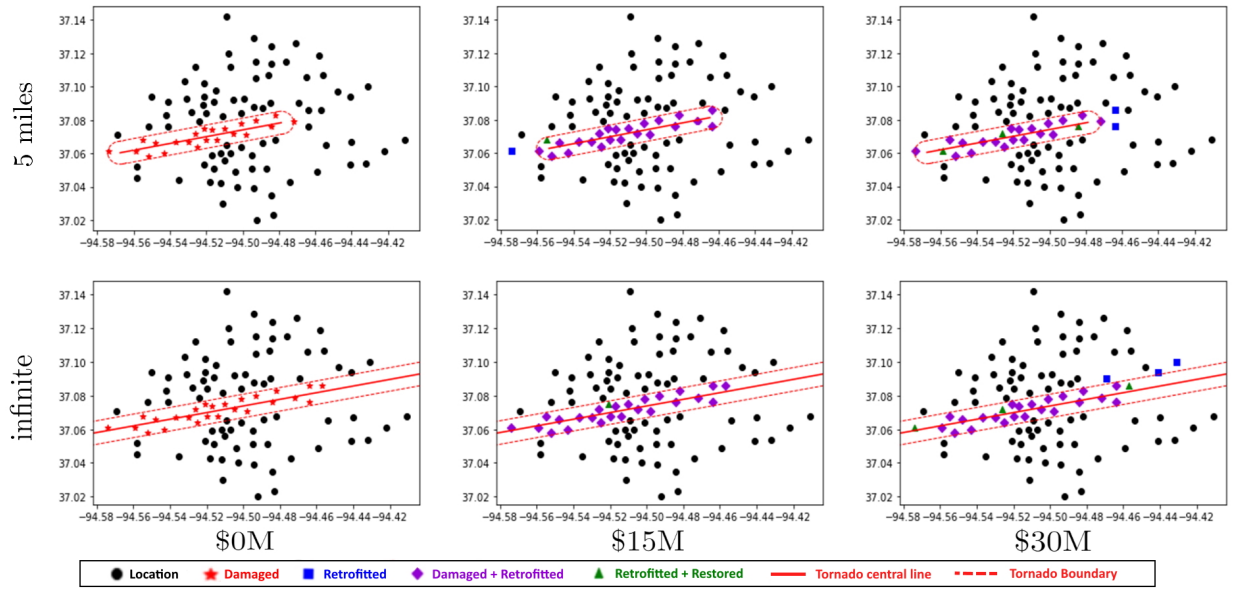


Figure 4.12: Maps of 100 locations and their retrofitting and recovery strategies under the worst-case tornado scenario for six parameter settings in Table 4.3.

Table 4.3 shows that the solver is able to find optimal solutions for every experiment in

at most 986 seconds. It turns out that solving the subproblem with DBC method is the most time-consuming step. In general, the algorithm solves the subproblem for 5 miles line segment faster than the infinite line; this fact can be justified by the effect of the conflict constraints (4.8) which are not present when $E = \infty$. In addition, for all experiments, the callback function performs very fast in checking the feasibility and solving the second stage problem on the fly. As expected, for the infinite line, the results show zero seconds to check the feasibility directly using $\tilde{\mathcal{U}}^h$, because in this case, the SLA alone can verify the feasibility.

For both tornado lengths, there are decreases in population dislocation by increasing the amount of available budget because decision-makers are able to retrofit and recover more locations. For example, increasing the budget from \$0 to \$15 million results in around 3000 fewer dislocations in damaged neighborhoods. However, adding another \$15 million will reduce dislocations only by around 500 people because of the great cost of recovery. Note that this translates in reductions of around 20% in population dislocation for both tornado lengths.

More details on the optimal strategies can be found in Figure 4.12, which maps the locations, their optimal strategies, and the worst-case tornado scenarios for various budgets. It shows that in all experiments the decision-maker uses the extra amount of budget from \$15M to \$30M to recover a few locations rather than in retrofit locations that are not in the path of the worst-case tornado. Notice that recovery costs are generally more expensive than retrofitting costs. For example, the recovered location in the ∞-*\$15M* map in Figure 4.12 has 470 residents; 382 of which are estimated to be dislocated by spending \$194K for retrofitting. However, the decision-maker has to spend \$5.5M in recovery to further reduce the estimated dislocation to 295 people for this location. Columns 9-10 in Table 4.3 show that the proportion of the budget that is assigned to recovery increases by having more amount of the budget. In addition, we can observe in Figure 4.12 that the path of worst-case tornadoes does not change by much in different experiments, suggesting that the location of the tornado path is largely driven by the population density in the city center area.

4.5.4 Advantages of robust retrofitting

In this section we compare the optimal two-stage robust retrofitting and recovery strategies against a policy that makes retrofitting and recovery decisions at random. We assume a $E = 5$ miles tornado path and a total budget of A =\$15 million. The decision-maker is allowed to spend \$0, \$3, \$9, and \$15 million of the budget in each experiment to randomly retrofit locations with random selections in set S; the remaining budget is reserved for the recovery actions. Once a retrofitting strategy f is determined, then the corresponding worst-case second-stage happens (i.e., the value of $\Phi(f)$ is determined, see Equation (4.6)). Table 4.4 compares the average, maximum, minimum, and standard deviation of population dislocation for 10 random replications. Note that for the \$0M case, we do not need multiple replications.

Budget to randomly retrofit	Population Dislocation			
	Average	Maximum	Minimum	Standard deviation
\$0M (0%)	16318			-
\$3M (20%)	16057	16263	15717	163
\$9M (60%)	15596	15905	15306	196
\$15M (100%)	15143	15451	14764	189
Robust optimal retrofitting	**13408**			-

Table 4.4: Comparison of population dislocation statistics for 10 random retrofitting plans with spending a certain amount of budget out of a total \$15M.

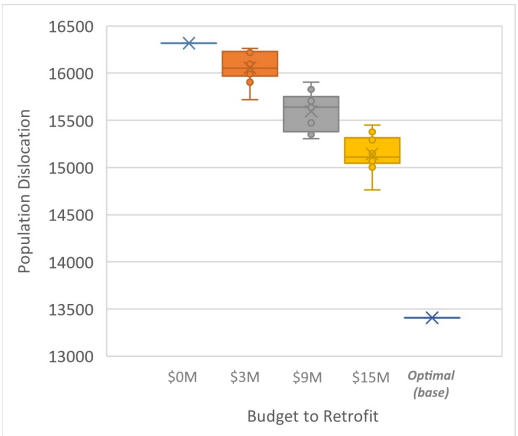


Figure 4.13: Box and Whisker plots for Table 4.4 experiments. It shows the reduction trend for population dislocation by considering more budget for retrofitting.

The results in Table 4.4 show that with random policies the average, maximum, and minimum dislocation after the corresponding worst-case tornado decrease with the percentage of the budget spent in retrofitting. These results are depicted graphically in the Box plots in Figure 4.13. Note that the proposed robust optimization model results in 13408 dislocations, which is noticeable better than the random policies. To further emphasize the value of optimization, recall that the optimal solution with our proposed method in Table 4.3 spends \$9M out of \$15M (60% of budget) in retrofitting. By contrast, even the minimum population dislocation with random retrofitting using the same budget proportion is 15306 (a 14% increase).

We conduct additional simulations to randomly retrofit locations with the full amount of \$15M and \$30M budget as before, but this time the worst-case tornado found in Section 4.5.3 is used. The results across 10 replications are shown in Table 4.5, which shows that for both cases the robust optimal population dislocation is significantly less than the minimum of population dislocation with random retrofitting. The plot in Figure 4.14 provides a visualization of the significant gap between robust optimal value (stars) and the distribution of values for random retrofitting plans. We can also observe that population dislocation reduces by increasing the amount of budget.

Budget	Population dislocation				
	Robust optimal retrofitting	Average	Maximum	Minimum	Standard deviation
\$15M	**13408**	15314	15941	14545	331
\$30M	**13091**	14795	15248	14145	294

Table 4.5: Comparison of population dislocation statistics using ten different random retrofitting plans and budgets of \$15M and \$30M.

The results of this section show that deciding using the proposed two-stage robust method yields significant reductions in population dislocation against random policies when evaluating in terms of worst-case outcomes. In the next section we complement these experiments by analyzing how the optimal policy behaves when random tornadoes, instead of the worst-case scenario, happen.

Figure 4.14: Box and Whisker plots for Table 4.5 experiments. It shows the gap between the robust optimal plan (stars) and the population dislocation of the random retrofitting strategies for the base worst-case tornado.

4.5.5 Robust strategies in random scenarios

In this section we fix the optimal retrofitting-recovery plan and compare the robust population dislocation values resulting from having 10 simulated 5-miles tornadoes, see Table 4.6 and Figure 4.15.

Budget	Population dislocation				
	Worst-case tornado	Average	Maximum	Minimum	Standard deviation
$0M	**16318**	10324	13201	5233	2835
$15M	**13408**	9634	12720	5031	2679
$30M	**13091**	9416	12480	4900	2607

Table 4.6: Comparison of population dislocation statistics for the worst-case tornado and simulated tornadoes by using the optimal retrofitting plan

The results for $0M in Table 4.6 and Figure 4.15 show that there is a large gap between the worst-case tornado 16318 (blue star) and the maximum value among simulation results which is 13201. As expected, by having more budget, the average population dislocations in the simulations reduce from 10324 to 9643 for $15M and to 9416 for $30M. In addition, the gap between the robust and maximum simulation values for $15 and $30M are noticeable smaller than the gap for no-budget experiment.

The observations in this section show a few things about the optimal retrofitting-recovery

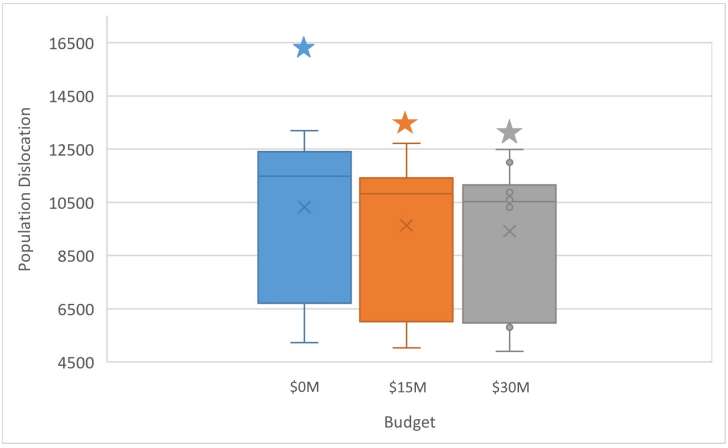


Figure 4.15: Box and Whisker plots for Table 4.6 experiments. It shows the distribution of dislocation values for simulated tornadoes and the gap to the worst-case tornado.

policy: first, that there can be a large variability for population dislocation assuming any tornado is possible. Second, that the variability seems to decrease as more budget is available, and third, that the worst-case scenario is not that far way from simulated scenarios, particularly when there are investments in retrofitting and recovery. This last point is further emphasized by the fact that only 10 tornadoes are simulated; one should expect that with more simulated tornadoes the gap between the worst-case and the maximum dislocation observed reduces even more. Further, this last point also shows that the worst-case is not as rare as one might anticipate, and thus that the proposed robust optimization model does not suffer from 'over-conservatism', which is a common drawback of deciding in a robust manner.

4.6 Conclusion

We consider a two-stage robust optimization model to determine retrofitting and recovery strategies before and after the realization of an uncertain tornado disaster. We assume that the decision-maker (a combination of public-private agencies) has to allocate a budget in retrofitting and recovery actions, to minimize population dislocation. We propose a two-stage optimization model where the first-stage variables determine the retrofitting strategies and the second-stage variables determine the recovery strategy, assuming that for any retrofitting

action, the corresponding worst-case tornado happens. Given that the uncertainty corresponds to the possible tornado paths, the resulting problem is a two-stage mixed-integer robust optimization problem with a mixed-integer non-linear uncertainty set.

We show that the proposed problem setting is NP-hard. A standard approach to solve the problem is a column-and-constraint generation algorithm that needs to solve a max-min subproblem at each iteration. We embed a decomposition branch-and-cut algorithm in the column-and-constraint solution method to address the non-convex structure of the bilevel subproblem. We propose two sets of valid inequalities and a high-performance separation procedure to implement the decomposition branch-and-cut algorithm. Using data from INCORE, we use our model to provide the optimal retrofitting and recovery strategies for the city of Joplin, Missouri against the worst-case scenario.

CHAPTER V

CONCLUSION AND FUTURE WORK

In this dissertation, we studied optimization models under uncertainty, with a focus on finding robust optimal solutions that are effective under any selection of uncertain parameters within a mixed-integer set. We have focused on two specific classes of robust optimization problems in which the uncertainty can be represented using mixed-integer sets. In this section, we summarize our contributions and identify possible future works.

5.1 Contributions

We first explored a class of adversarial minimum-cost flow problems that are subject to multiple ripple effect disruptions that increase arc usage cost. We have addressed the challenge of the uncertain locations of the disruptions' epicenters by seeking a flow that minimizes cost assuming the worst-case realization of the disruptions. We evaluate the damage to each arc using a linear model, where the damage is the cumulative damage of all disruptions affecting the arc; and a maximum model, where the damage is given by the most destructive disruption affecting the arc. For both models, the arcs' costs post-disruptions are represented with a mixed-integer feasible region, resulting in a robust optimization problem with a mixed-integer uncertainty set. We have shown that the subproblem evaluating the worst-case cost for a given flow plan is polynomial-time solvable for the linear model, but NP-hard for the maximum model. To effectively solve the mixed-integer subproblem, we have proposed a big-M free binary reformulation of the uncertainty set based on conflict constraints, resulting

in a significantly tighter linear programming relaxation. We have also extended our models to consider a less conservative approach where only a subset of the disruptions can occur, and shown that the properties of the linear and maximum models also hold in this case. Our proposed approaches have been tested on real road networks and synthetic instances, showing orders of magnitude improvements over a standard approach from the literature. These findings contribute to the development of efficient and effective solutions for adversarial minimum-cost flow problems with uncertain ripple effect disruptions.

Furthermore, this dissertation proposes a novel problem that addresses the critical issue of mitigating the effects of catastrophic tornadoes on human well-being by efficiently allocating resources in retrofitting and recovery strategies. We use a two-stage robust optimization approach with a mixed-integer nonlinear uncertainty set that models tornado damage and explicitly represents tornado paths as line segments. Our proposed algorithm utilizes the column-and-constraint generation method from existing literature, equipped with a decomposition branch-and-cut algorithm. With this approach, we can effectively solve two-stage optimization problems with mixed-integer formulations in both stages. To demonstrate the applicability of our approach, we present numerical results for a case study based on Joplin, Missouri. Our results show that by investing \$15 million in retrofitting and recovery, up to 20% reductions in worst-case population dislocation can be achieved. Moreover, our approach outperforms other retrofitting policies by margins close to 20%, while avoiding over-conservativeness. Our proposed methodology can serve as a valuable tool for decision-makers, such as government agencies and public-private consortia, to allocate resources effectively in minimizing population dislocation due to uncertain tornadoes.

5.2 Future work

5.2.1 Hedging against rippled disruptions

For robust minimum-cost flow problems, future work includes considering more broad effects of the disruptions not only on the cost of the arcs but also on their capacity and the demand/supply at the nodes. One important aspect to consider is the potential for a ripple to cause leakage of flow at the nodes it covers. It is reasonable to assume that nodes closer to the epicenter of a disruption may experience more leakage of flow than nodes farther away. To address this, we propose defining a dummy sink that can account for this leakage. The dummy sink would have ingoing arcs from the nodes, which would be activated based on the extent of the ripple damage, thereby reducing the amount of flow delivered to the actual sink. In this setting, one important consideration is the desire to minimize the amount of leakage. By incorporating this objective into the model, decision-makers could take a more comprehensive approach to managing the impact of ripple effect disruptions.

Also, alternative modeling approaches for the interactions between the disruptions can be considered for further study. To elaborate on this, one possible approach is to use continuous models to compute the damage caused by disruptions in a location. In this approach, the exact distance between the location and the epicenters of the disruptions is taken into account to evaluate the damage. It is worth noting that even with this continuous modeling approach, the linear and maximum cost functions described in the current work can still be applied.

Another line of future research may study combinatorial network problems (e.g., traveling salesman problem, vehicle routing, network design) subject to ripple effect disruptions, which we anticipate to be even more challenging than minimum-cost flow problems and could span an even wider array of applications. The vehicle routing problem, for example, involves determining the optimal routes for a fleet of vehicles to deliver goods or services to various locations. Disruptions can impact the capacity of the vehicles and the demand/supply at the

delivery locations, and a robust optimal strategy could prevent further expenses to constantly re-optimize the routes in real-time.

Moreover, future research may consider more flexible models in which the flow plans could be (partially) updated after observing the realization of the uncertainty. The latter work could be analyzed using the framework of two-stage robust problems, which was introduced in Chapter IV. For example, the decision-maker can reserve a portion of flow to release after observing the cost vector following a disruption. It is possible to analyze the additional cost that the decision-maker must pay to reserve a portion of flow versus the amount of budget saved by observing the disruption locations. This analysis can provide valuable insights into the effectiveness of this approach and can help the decision-maker to make less conservative decisions.

5.2.2 Tornado preparedness

As a future direction for the study of retrofitting and recovery planning in tornado-prone areas, it would be beneficial to investigate modeling tornado damages under the assumption that there could be multiple levels of disruptions. This would involve considering ripple effect disruptions, as described in Chapter III. Specifically, it could be assumed that locations closer to the central line of a tornado would experience greater damage compared to areas farther away. Additionally, the direction and intensity of a tornado could be assumed to change over time, which would allow for a more accurate representation of its behavior. To achieve this, it would be necessary to define time periods (T) in which the motion of the tornado changes to result in worst-case levels of damage. This would provide a more comprehensive understanding of the potential impact of tornadoes on different areas and enable more effective planning for retrofitting and recovery efforts.

Another promising avenue of research could involve applying our proposed framework to investigate community resilience under other natural disasters, such as hurricanes. This could be achieved with minor modifications to the uncertainty model and the set of retrofitting

and recovery strategies. Overall, this approach would provide a robust tool for improving community resilience under a range of natural disasters.

REFERENCES

E. Álvarez-Miranda and M. Sinnl. An exact solution framework for the multiple gradual cover location problem. *Computers & Operations Research*, 108:82–96, 2019.

Y. An and B. Zeng. Exploring the modeling capacity of two-stage robust optimization: Variants of robust unit commitment model. *IEEE Transactions on Power Systems*, 30 (1):109–122, 2015. doi: 10.1109/TPWRS.2014.2320880.

Y. An, B. Zeng, Y. Zhang, and L. Zhao. Reliable p-median facility location problem: two-stage robust models and algorithms. *Transportation Research Part B: Methodological*, 64:54–72, 2014.

M. Ansari, J. S. Borrero, and L. Lozano. Robust minimum-cost flow problems under multiple-ripple effect disruptions online repository. C++ codes online at: `https://github.com/mehdi-ansari/Robust_minimum_cost_flow_ripple_disruptions`, 2022. Accessed: August 16, 2022.

M. Ansari, J. S. Borrero, and L. Lozano. Robust minimum-cost flow problems under multiple ripple effect disruptions. *INFORMS Journal on Computing*, 35(1):83–103, 2023.

A. Atamtürk and M. Zhang. Two-stage robust network flow and design under demand uncertainty. *Operations Research*, 55(4):662–673, 2007.

J. Bagherinejad, M. Bashiri, and H. Nikzad. General form of a cooperative gradual maximal covering location problem. *Journal of Industrial Engineering International*, 14(2):241–253, 2018.

A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of operations research*, 23(4):769–805, 1998.

A. Ben-Tal and A. Nemirovski. Robust solutions to uncertain linear programs. *Operations Research Letters*, 25:1–13, 1999.

A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88(3):411–424, 2000.

A. Ben-Tal and A. Nemirovski. Robust optimization – methodology and applications. *Mathematical Programming*, 92(3):453–480, 2002.

A. Ben-Tal and A. Nemirovski. Selected topics in robust convex optimization. *Mathematical Programming*, 112(1):125–158, 2008.

A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.

A. Ben-Tal, B. Golany, A. Nemirovski, and J.-P. Vial. Retailer-supplier flexible commitments contracts: A robust optimization approach. *Manufacturing & Service Operations Management*, 7(3):248–271, 2005.

A. Ben-Tal, S. Boyd, and A. Nemirovski. Extending scope of robust optimization: Comprehensive robust counterparts of uncertain problems. *Mathematical Programming*, 107(1-2):63–89, 2006.

A. Ben-Tal, L. El-Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, Princeton, NJ, 2009.

A. Ben-Tal, B. Do Chung, S. R. Mandala, and T. Yao. Robust optimization for emergency logistics planning: Risk mitigation in humanitarian relief supply chains. *Transportation research part B: methodological*, 45(8):1177–1189, 2011.

A. Ben-Tal, E. Hazan, T. Koren, and S. Mannor. Oracle-based robust optimization via online learning. *Operations Research*, 63(3):628–638, 2015.

O. Berman and D. Krass. The generalized maximal covering location problem. *Computers & Operations Research*, 29(6):563–581, 2002.

O. Berman, D. Krass, and Z. Drezner. The gradual covering decay location problem on a network. *European Journal of Operational Research*, 151(3):474–480, 2003.

O. Berman, Z. Drezner, and D. Krass. Cooperative cover location problems: the planar case. *IIE Transactions*, 42(3):232–246, 2009.

O. Berman, Z. Drezner, and D. Krass. Generalized coverage: New developments in covering location models. *Computers & Operations Research*, 37(10):1675–1687, 2010.

O. Berman, Z. Drezner, and D. Krass. The multiple gradual cover location problem. *Journal of the Operational Research Society*, 70(6):931–940, 2019.

D. Bertsimas and D. B. Brown. Constructing uncertainty sets for robust linear optimization. *Operations research*, 57(6):1483–1495, 2009.

D. Bertsimas and M. Sim. The price of robustness. *Operations research*, 52(1):35–53, 2004.

D. Bertsimas and A. Thiele. A robust optimization approach to inventory theory. *Operations research*, 54(1):150–168, 2006.

D. Bertsimas, E. Litvinov, X. A. Sun, J. Zhao, and T. Zheng. Adaptive robust optimization for the security constrained unit commitment problem. *IEEE transactions on power systems*, 28(1):52–63, 2012.

D. Bertsimas, I. Dunning, and M. Lubin. Reformulation versus cutting-planes for robust optimization. *Computational Management Science*, 13(2):195–217, 2016.

D. Bertsimas, M. Sim, and M. Zhang. Adaptive distributionally robust optimization. *Management Science*, 2018.

I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo. The maximum clique problem. In *Handbook of combinatorial optimization*, pages 1–74. Springer, 1999.

J. S. Borrero and L. Lozano. Modeling defender–attacker problems as robust linear programs with mixed–integer uncertainty sets. *INFORMS Journal on Computing (forthcoming)*, 2020.

J. S. Borrero, O. A. Prokopyev, and D. Sauré. Sequential shortest path interdiction with incomplete information. *Decision Analysis*, 13(1):68–98, 2016.

P. Cappanera and M. P. Scaparra. Optimal allocation of protective resources in shortest-path networks. *Transportation Science*, 45(1):64–80, 2011.

B. M. Chazelle and D.-T. Lee. On a circle placement problem. *Computing*, 36(1-2):1–16, 1986.

C. Cheng, Y. Adulyasak, and L.-M. Rousseau. Robust facility location under demand uncertainty and facility disruptions. *Omega*, 103:102429, 2021.

R. Church and C. ReVelle. The maximal covering location problem. In *Papers of the Regional Science Association*, volume 32, pages 101–118. Springer-Verlag, 1974.

R. L. Church. Symposium on location problems: in memory of leon cooper: the planar maximal covering location problem. *Journal of Regional Science*, 24(2):185–201, 1984.

K. Cross, O. Dullum, N. Jenzen-Jones, and M. Garlasco. Explosive weapons in populated areas: technical considerations relevant to their use and effects. *Armament Research Services (ARES), Perth*, 2016.

M. De, S. C. Nandy, and S. Roy. In-place algorithms for computing a largest clique in geometric intersection graphs. *Discrete Applied Mathematics*, 178:58–70, 2014.

S. Dempe, V. Kalashnikov, G. A. Pérez-Valdés, and N. Kalashnykova. *Bilevel Programming Problems*. Springer, Heidelberg, 2015.

S. T. DeNegre and T. K. Ralphs. A branch-and-cut algorithm for integer bilevel linear programs. In J. W. Chinneck, B. Kristjansson, and M. J. Saltzman, editors, *Operations Research and Cyber-Infrastructure*, pages 65–78. Springer, New York, 2009.

T. Ding, C. Li, Y. Yang, J. Jiang, Z. Bie, and F. Blaabjerg. A two-stage robust optimization for centralized-optimal dispatch of photovoltaic inverters in active distribution networks. *IEEE Transactions on Sustainable Energy*, 8(2):744–754, 2017. doi: 10.1109/TSTE.2016.2605926.

M. C. Dourado, F. Protti, and J. L. Szwarcfiter. Complexity aspects of the helly property: Graphs and hypergraphs. *The Electronic Journal of Combinatorics*, pages DS17–Jun, 2012.

Z. Drezner, G. O. Wesolowsky, and T. Drezner. The gradual covering problem. *Naval Research Logistics (NRL)*, 51(6):841–855, 2004.

F. Fan and W. Pang. Stochastic track model for tornado risk assessment in the us. *Frontiers in built environment*, 5:37, 2019.

K. Farokhnia, J. W. van de Lindt, and M. Koliou. Selection of residential building design requirements to achieve community functionality goals under tornado loading. *Practice Periodical on Structural Design and Construction*, 25(1):04019035, 2020.

FEMA. Natural hazard retrofit program toolkit. `https://www.fema.gov/sites/default/files/documents/fema\_natural-hazards-retrofit-program-tookit.pdf`, Accessed January 2023, February 2021.

FEMA. Tornado–alerts and warnings. Online at `https://community.fema.gov/ProtectiveActions`, Accessed November 2022, November 2022a.

FEMA. Resources for repairing, retrofitting and rebuilding after a tornado. https://www.fema.gov/fact-sheet/resources-repairing-retrofitting-and-rebuilding-after-tornado-0, Accessed January 2023, March 2022b.

M. Fischetti, I. Ljubić, M. Monaci, and M. Sinnl. A new general-purpose algorithm for mixed-integer bilevel linear programs. *Operations Research*, 65(6):1615–1637, 2017.

R. J. Fowler, M. S. Paterson, and S. L. Tanimoto. Optimal packing and covering in the plane are np-complete. *Information processing letters*, 12(3):133–137, 1981.

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

B. L. Gorissen, İ. Yanıkoğlu, and D. den Hertog. A practical guide to robust optimization. *Omega*, 53:124–137, 2015.

C. Gregory, K. Darby-Dowman, and G. Mitra. Robust optimization and portfolio selection: The cost of robustness. *European Journal of Operational Research*, 212(2):417–428, 2011.

P. Hansen, B. Jaumard, and G. Savard. New branch-and-bound rules for linear bilevel programming. *SIAM Journal on scientific and Statistical Computing*, 13(5):1194–1217, 1992.

M. Hazus. Multi-hazard loss estimation methodology. *Earthquake Model (HAZUS MH. MR4) Technical Manual. Department of Homeland Security, Emergency Preparedness and Response Directorate, FEMA*, 2003.

N. Ho-Nguyen and F. Kılınç-Karzan. Online first-order framework for robust convex optimization. *Operations Research*, 66(6):1670–1692, 2018.

H. Imai and T. Asano. Finding the connected components and a maximum clique of an intersection graph of rectangles in the plane. *Journal of algorithms*, 4(4):310–323, 1983.

IN-CORE. Interdependent networked community resilience modeling environment. https://incore.ncsa.illinois.edu/, Accessed October 2022, October 2022.

E. Israeli and R. K. Wood. Shortest-path network interdiction. *Networks: An International Journal*, 40(2):97–111, 2002.

R. A. Jabr, I. Džafić, and B. C. Pal. Robust optimization of storage investment on transmission networks. *IEEE Transactions on Power Systems*, 30(1):531–539, 2015. doi: 10.1109/TPWRS.2014.2326557.

A. Jalilvand-Nejad, R. Shafaei, and H. Shahriari. Robust optimization under correlated polyhedral uncertainty set. *Computers & Industrial Engineering*, 92:82–94, 2016.

R. Jiang, M. Zhang, G. Li, and Y. Guan. Benders' decomposition for the two-stage security constrained robust unit commitment problem. In *IIE Annual Conference. Proceedings*, page 1. Institute of Industrial and Systems Engineers (IISE), 2012.

M. Karatas and A. Dasci. A two-level facility location and sizing problem for maximal coverage. *Computers & Industrial Engineering*, 139:106204, 2020.

A. Karduni, A. Kermanshah, and S. Derrible. A protocol to convert spatial polyline data to network formats and applications to world urban road networks. *Scientific data*, 3(1):1–7, 2016.

H. Kellerer, U. Pferschy, and D. Pisinger. Introduction to np-completeness of knapsack problems. In *Knapsack problems*, pages 483–493. Springer, 2004.

M. Koliou and J. W. van de Lindt. Development of building restoration functions for use in community recovery planning to tornadoes. *Natural Hazards Review*, 21(2):04020004, 2020.

E. D. Kuligowski, F. T. Lombardo, L. Phan, M. L. Levitan, and D. P. Jorgensen. Final report, national institute of standards and technology (nist) technical investigation of the may 22, 2011, tornado in joplin, missouri. 2014.

G. Laporte and F. V. Louveaux. The integer l-shaped method for stochastic integer programs with complete recourse. *Operations research letters*, 13(3):133–142, 1993.

Y. Li, F. Zhang, Y. Li, and Y. Wang. An improved two-stage robust optimization model for cchp-p2g microgrid system considering multi-energy operation under wind power outputs uncertainties. *Energy*, 223:120048, 2021.

Z. Li, R. Ding, and C. A. Floudas. A comparative theoretical and computational study on robust counterpart optimization: I. robust linear optimization and robust mixed integer linear optimization. *Industrial & engineering chemistry research*, 50(18):10567–10603, 2011.

X. Lin, S. L. Janak, and C. A. Floudas. A new robust optimization approach for scheduling under uncertainty: I. bounded uncertainty. *Computers & chemical engineering*, 28(6-7): 1069–1085, 2004.

L. Lozano and J. C. Smith. A value-function-based exact approach for the bilevel mixed-integer programming problem. *Operations Research*, 65(3):768–786, 2017a.

L. Lozano and J. C. Smith. A backward sampling framework for interdiction problems with fortification. *INFORMS Journal on Computing*, 29(1):123–139, 2017b.

S. Ma, B. Chen, and Z. Wang. Resilience enhancement strategy for distribution systems under extreme weather events. *IEEE Transactions on Smart Grid*, 9(2):1442–1451, 2018.

J. MacQueen. Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, pages 281–297, 1967.

H. Masoomi and J. W. van de Lindt. Restoration and functionality assessment of a community subjected to tornado hazard. *Structure and Infrastructure Engineering*, 14(3):275–291, 2018.

H. Masoomi, M. R. Ameri, and J. W. van de Lindt. Wind performance enhancement strategies for residential wood-frame buildings. *Journal of Performance of Constructed Facilities*, 32(3):04018024, 2018.

L. R. Matthews, C. E. Gounaris, and I. G. Kevrekidis. Designing networks with resiliency to edge failures using two-stage robust optimization. *European Journal of Operational Research*, 279(3):704–720, 2019.

T. P. McAllister et al. Community resilience planning guide for buildings and infrastructure systems, volume i. 2015.

A. Mitsos. Global solution of nonlinear mixed-integer bilevel programs. *Journal of Global Optimization*, 47(4):557–582, 2010.

Y. Moon and T. Yao. A robust mean absolute deviation model for portfolio optimization. *Computers & Operations Research*, 38(9):1251–1258, 2011.

A. Mutapcic and S. Boyd. Cutting-set methods for robust convex optimization with pessimizing oracles. *Optimization Methods & Software*, 24(3):381–406, 2009.

K. Natarajan, D. Pachamanova, and M. Sim. Constructing risk measures from uncertainty sets. *Operations research*, 57(5):1129–1141, 2009.

S. Neyshabouri and B. P. Berg. Two-stage robust optimization approach to elective surgery and downstream capacity planning. *European Journal of Operational Research*, 260(1): 21–40, 2017.

D. T. Nguyen, H. T. Nguyen, N. Trieu, and V. K. Bhargava. Two-stage robust edge service placement and sizing under demand uncertainty. *IEEE Internet of Things Journal*, 9(2): 1560–1574, 2021.

NOAA. National centers for environmental information, tornadoes statistics. Online at https://www.ncei.noaa.gov, Accessed September 2022, September 2022a.

NOAA. Storm predication center - national weather service. Online at https://www.spc.noaa.gov, Accessed October 2022, October 2022b.

NOAA. Tornadoes frequently asked questions. Online at https://www.weather.gov/lmk/tornadoesfaq, Accessed November 2022, November 2022c.

NOAA. Supercells. Online at https://www.spc.noaa.gov/misc/AbtDerechos/supercells.htm, Accessed November 2022, November 2022d.

NOAA. Weather related fatality and injury statistics. Online at https://www.weather.gov/hazstat/, Accessed November 2022, November 2022e.

NOAA. National weather service, the enhanced fujita scale (ef scale). https://www.weather.gov, Accessed September 2022, September 2022f.

O. Nohadani and K. Sharma. Optimization under decision-dependent uncertainty. *SIAM Journal on Optimization*, 28(2):1773–1795, 2018.

NWC. Historical data of oklahoma county tornadoes. Online at https://www.weather.gov/oun/tornadodata-county-ok-oklahoma, Accessed January 2023, January 2023.

M. H. Patel and A. J. Horowitz. Optimal routing of hazardous materials considering risk of spill. *Transportation Research Part A: Policy and Practice*, 28(2):119–132, 1994.

M. Poss. Robust combinatorial optimization with variable cost uncertainty. *European Journal of Operational Research*, 237(3):836–845, 2014.

S. Pu and S. Zhan. Two-stage robust railway line-planning approach with passenger demand uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, 152: 102372, 2021.

R. Rahmati, M. Bashiri, E. Nikzad, and A. Siadat. A two-stage robust hub location problem with accelerated benders decomposition algorithm. *International Journal of Production Research*, 60(17):5235–5257, 2022.

J. T. Ripberger, H. C. Jenkins-Smith, C. L. Silva, J. Czajkowski, H. Kunreuther, and K. M. Simmons. Tornado damage mitigation: Homeowner support for enhanced building codes in oklahoma. *Risk analysis*, 38(11):2300–2317, 2018.

J. O. Royset and R. K. Wood. Solving the bi-objective maximum-flow network-interdiction problem. *INFORMS Journal on Computing*, 19(2):175–184, 2007.

S. Saba. Line intersecting maximal number of circles (circle "stabbing" problem). `https://sahandsaba.com/line-intersecting-maximal-number-of-circles.html`, Accessed October 2022, October 2013.

H. Sengul, N. Santella, L. J. Steinberg, and A. M. Cruz. Analysis of hazardous material releases due to natural hazards in the united states. *Disasters*, 36(4):723–743, 2012.

M. H. Shams, M. Shahabi, M. MansourLakouraj, M. Shafie-khah, and J. P. Catalão. Adjustable robust optimization approach for two-stage operation of energy hub-based microgrids. *Energy*, 222:119894, 2021.

K. M. Simmons, P. Kovacs, and G. A. Kopp. Tornado damage mitigation: Benefit–cost analysis of enhanced building codes in oklahoma. *Weather, climate, and society*, 7(2):169–178, 2015.

A. Sinha, P. Malo, and K. Deb. A review on bilevel optimization: from classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2017.

A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21:1154–1157, 1973.

C. D. Standohar-Alfano, J. W. van de Lindt, and B. R. Ellingwood. Vertical load path failure risk analysis of residential wood-frame construction in tornadoes. *Journal of Structural Engineering*, 143(7):04017045, 2017.

M. Stoner and W. Pang. Tornado hazard assessment of residential structures built using cross-laminated timber and light-frame wood construction in the us. *Natural Hazards Review*, 22(4):04021032, 2021.

S. M. Strader, T. J. Pingel, and W. S. Ashley. A monte carlo model for estimating tornado impacts. *Meteorological Applications*, 23(2):269–281, 2016.

S. Tahernejad, T. K. Ralphs, and S. T. DeNegre. A branch-and-cut algorithm for mixed integer bilevel linear optimization problems and its implementation. *Mathematical Programming Computation*, 12(4):529–568, 2020.

A. Takeda, S. Taguchi, and R. Tütüncü. Adjustable robust optimization models for a nonlinear two-period system. *Journal of Optimization Theory and Applications*, 136(2):275–295, 2008.

The White House. Fact sheet: Biden–harris administration launches initiative to modernize building codes, improve climate resilience, and reduce energy costs. `https://www.whitehouse.gov`, June 2022. Accessed: 2023-01.

A. Thiele, T. Terry, and M. Epelman. Robust linear optimization with recourse. *Rapport technique*, pages 4–37, 2009. URL `http://www.optimization-online.org/DB_FILE/2009/03/2263.pdf`.

J. W. van de Lindt and T. N. Dao. Performance-based wind engineering for wood-frame buildings. *Journal of Structural Engineering*, 135(2):169–177, 2009.

G. A. Velasquez, M. E. Mayorga, and O. Y. Özaltın. Prepositioning disaster relief supplies using robust optimization. *IISE Transactions*, 52(10):1122–1140, 2020.

M. Verma and V. Verter. Railroad transportation of dangerous goods: Population exposure to airborne toxins. *Computers & operations research*, 34(5):1287–1303, 2007.

L. Vicente, G. Savard, and J. Júdice. Descent approaches for quadratic bilevel programming. *Journal of optimization theory and applications*, 81(2):379–399, 1994.

J. Wang, S. Cao, W. Pang, and J. Cao. Experimental study on effects of ground roughness on flow characteristics of tornado-like vortices. *Boundary-Layer Meteorology*, 162:319–339, 2017.

W. Wang, J. W. Van De Lindt, N. Rosenheim, H. Cutler, B. Hartman, J. Sung Lee, and D. Calderon. Effect of residential building wind retrofits on social and economic community-level resilience metrics. *Journal of Infrastructure Systems*, 27(4):04021034, 2021.

Y. Wen. *Development of Multi-Objective Optimization Model of Community Resilience on Mitigation Planning*. PhD thesis, University of Oklahoma, 2021.

Y. Wen, C. Nicholson, and A. D. González. Spatio-temporal structural intervention optimization framework for hazard mitigation. pages 1–16, 2021.

P. Xiong, P. Jirutitijaroen, and C. Singh. A distributionally robust optimization model for unit commitment considering uncertain wind power generation. *IEEE Transactions on Power Systems*, 32(1):39–49, 2017.

P. Xu and L. Wang. An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. *Computers and Operations Research*, 41 (1):309–318, 2014.

T. Yao, S. R. Mandala, and B. Do Chung. Evacuation transportation planning under uncertainty: a robust optimization approach. *Networks and Spatial Economics*, 9(2):171, 2009.

W. Yuan, J. Wang, F. Qiu, C. Chen, C. Kang, and B. Zeng. Robust optimization-based resilient distribution network planning against natural disasters. *IEEE Transactions on Smart Grid*, 7(6):2817–2826, 2016.

B. Zeng and L. Zhao. Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters*, 41(5):457–461, 2013.

J. Zhang, J. Hodgson, and E. Erkut. Using gis to assess the risks of hazardous materials transport in networks. *European Journal of Operational Research*, 121(2):316–329, 2000.

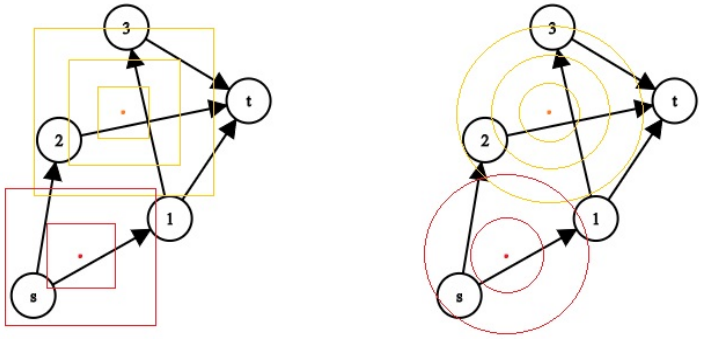
L. Zhao and B. Zeng. Robust unit commitment problem with demand response and wind energy. In *2012 IEEE power and energy society general meeting*, pages 1–8. IEEE, 2012.

CHAPTER VI

APPENDIX

6.1 MIP formulations for the ℓ^∞ and ℓ^2 norms

In this appendix, we present the basic MIP formulations for the uncertainty sets if the distances are induced by the ℓ^∞ and ℓ^2 norm, see Figure 6.1.



(a) ℓ^∞-norm (b) ℓ^2-norm

Figure 6.1: Disruption shapes under the ℓ^∞ and the ℓ^2 norm.

- **ℓ^∞-norm:** The ripples have a square shaped geometry, see Figure 6.1a. Constraint (3.5d) determines the following distance from arcs to epicenters:

$$\max\left\{|p_a^{(1)} - \alpha_j^{(1)}|, |p_a^{(2)} - \alpha_j^{(2)}|\right\} - M_a(1 - \lambda_{ajk}) \leq q_{jk} \qquad \forall a \in A, k \in [r_j], j \in [m]. \tag{6.1}$$

In this case, the maximum in (6.1) can be rewritten as two constraints such that the maximum value of its elements determines the lower bound of the inequality. Hence, we

can replace constraint (3.5d) by the following constraints to obtain a linear formulation for the ℓ^∞-norm:

$$p_a^{(1)} - \alpha_j^{(1)} = \beta_{1aj}^+ - \beta_{1aj}^- \qquad \forall a \in A,\ j \in [m] \tag{6.2a}$$

$$p_a^{(2)} - \alpha_j^{(2)} = \beta_{2aj}^+ - \beta_{2aj}^- \qquad \forall a \in A,\ j \in [m] \tag{6.2b}$$

$$\beta_{1aj}^+ + \beta_{1aj}^- - M_a(1 - \lambda_{ajk}) \leq q_{jk} \qquad \forall a \in A, k \in [r_j]\ j \in [m] \tag{6.2c}$$

$$\beta_{2aj}^+ + \beta_{2aj}^- - M_a(1 - \lambda_{ajk}) \leq q_{jk} \qquad \forall a \in A, k \in [r_j]\ j \in [m] \tag{6.2d}$$

$$\beta_{1aj}^+, \beta_{1aj}^-, \beta_{2aj}^+, \beta_{2aj}^- \in \mathbb{R}_+ \qquad \forall a \in A,\ j \in [m]. \tag{6.2e}$$

- **ℓ^2-norm:** The ripples have a circle shaped geometry, see Figure 6.1b. Constraint (3.5d) determines the Euclidean distance from arcs to epicenters:

$$\sqrt{(p_a^{(1)} - \alpha_j^{(1)})^2 + (p_a^{(2)} - \alpha_j^{(2)})^2} - M_a(1 - \lambda_{ajk}) \leq q_{jk} \qquad \forall a \in A, k \in [r_j], j \in [m]. \tag{6.3}$$

  Even though constraint (6.3) cannot be written as a linear function, it can be replaced by the following convex quadratic constraint:

$$(p_a^{(1)} - \alpha_j^{(1)})^2 + (p_a^{(2)} - \alpha_j^{(2)})^2 - M_a'(1 - \lambda_{ajk}) \leq q_{jk}^2 \qquad \forall a \in A, k \in [r_j], j \in [m]. \tag{6.4}$$

  where M_a' plays a similar role as M_a but with respect to the squared distance.

The sizes of the MIP formulations for the ℓ^1, ℓ^2, and ℓ^∞ induced norms are shown in Table 6.1. Observe that the ℓ^1-norm and ℓ^∞-norm have $4|A|m$ more variables than the ℓ^2-norm because of the variables added to linearize the formulations. However, as seen in the second row, the total number of integer variables for all cases is the same. The ℓ^∞-norm formulation has $|A| \sum_{j \in [m]} r_j$ more constraints than the ℓ^1-norm formulation because the linearization of the max function requires constraints for both the horizontal and the vertical

#	ℓ^1-norm	ℓ^∞-norm	ℓ^2-norm
Variables	$\lvert A\rvert(1+\sum_{j\in[m]} r_j + 4m) + 2m$	$\lvert A\rvert(1+\sum_{j\in[m]} r_j + 4m) + 2m$	$\lvert A\rvert(1+\sum_{j\in[m]} r_j) + 2m$
Integer variables	$\lvert A\rvert \sum_{j\in[m]} r_j$	$\lvert A\rvert \sum_{j\in[m]} r_j$	$\lvert A\rvert \sum_{j\in[m]} r_j$
Constraints	$\lvert A\rvert(1+3m+\sum_{j\in[m]} r_j)$	$\lvert A\rvert(1+3m+2\sum_{j\in[m]} r_j)$	$\lvert A\rvert(1+m+\sum_{j\in[m]} r_j)$
Nonzeros	$\lvert A\rvert(1+7\sum_{j\in[m]} r_j + 6m)$	$\lvert A\rvert(1+8\sum_{j\in[m]} r_j + 6m)$	-

Table 6.1: Sizes of the formulations for the different notions of distance.

coordinates. Also, the ℓ^2-norm formulation has the fewest number of constraints because the constraint (6.3) has been transformed to (6.4) without any additional variable or constraint. Finally, the total number of nonzero elements in the coefficient matrices of formulations are also shown in the table. Because the formulation for the ℓ^2-norm is quadratic and not directly comparable, its column has been left empty.

6.2 Results for the ℓ^∞ norm

In this appendix, we show that the results that hold for the ℓ^1 norm also hold true for the ℓ^∞ norm.

Proposition 15. *For a disruption $j \in [m]$ under the distance measure induced by the ℓ^∞ norm, if $\lambda = (\lambda^1, \ldots, \lambda^m) \in \mathcal{C}$ and $\lambda^j \in \mathcal{B}_j$, then $\lambda^j \in \mathcal{M}_j$.*

Proof. Using the terminology of the proof of Proposition 7, we must show there exist $\alpha_j \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}]$ such that

$$\max\left\{|p_a^{(1)} - \alpha_j^{(1)}|, |p_a^{(2)} - \alpha_j^{(2)}|\right\} \leq q_{jk_a} \qquad \forall a \in A_j. \tag{6.5a}$$

Observe that (6.5a) holds if and only if the following holds:

$$-q_{jk_a} + p_a^{(1)} \leq \alpha_j^{(1)} \leq q_{jk_a} + p_a^{(1)} \qquad \forall a \in A_j \tag{6.6a}$$

$$-q_{jk_a} + p_a^{(2)} \leq \alpha_j^{(2)} \leq q_{jk_a} + p_a^{(2)} \qquad \forall a \in A_j. \tag{6.6b}$$

Equations (6.6) hold if and only if the following holds:

$$\max_{a\in A_j}\{-q_{jk_a}+p_a^{(1)}\}\leq \min_{a\in A_j}\{q_{jk_a}+p_a^{(1)}\} \tag{6.7a}$$

$$\max_{a\in A_j}\{-q_{jk_a}+p_a^{(2)}\}\leq \min_{a\in A_j}\{q_{jk_a}+p_a^{(2)}\}. \tag{6.7b}$$

To see why, note that its clear that (6.6) implies that (6.7) holds. On the other hand, suppose that (6.7) holds. Define $\alpha_j^{(1)}=\max_{a\in A_j}\{-q_{jk_a}+p_a^{(1)}\}$ and $\alpha_j^{(2)}=\max_{a\in A_j}\{-q_{jk_a}+p_a^{(2)}\}$, then it is clear that (6.6) holds.

Let $\max_{a\in A_j}\{-q_{jk_a}+p_a^{(1)}\}=-q_{jk_e}+p_e^{(1)}$, $\min_{a\in A_j}\{q_{jk_a}+p_a^{(1)}\}=q_{jk_f}+p_f^{(1)}$, $\max_{a\in A_j}\{-q_{jk_a}+p_a^{(2)}\}=-q_{jk_g}+p_g^{(2)}$, and $\min_{a\in A_j}\{q_{jk_a}+p_a^{(2)}\}$. So, we rewrite (6.7a) and (6.7b) as follows

$$p_e^{(1)}-p_f^{(1)}\leq q_{jk_e}+q_{jk_f} \tag{6.8a}$$

$$p_g^{(2)}-p_h^{(2)}\leq q_{jk_g}+q_{jk_h}. \tag{6.8b}$$

Since $(k_e,k_f)\notin\Gamma_{e,f}^j$ and $(k_g,k_h)\notin\Gamma_{g,h}^j$, then the inequalities (6.8a) and (6.8b) hold. □

Now we show to obtain the epicenters of the disruptions for the ℓ^∞ case once formulation (3.39) is solved. Define

$$m_1^*=\max_{a\in A_j}\{-q_{jk_{aj}}+p_a^{(1)}\},\quad M_1^*=\min_{a\in A_j}\{q_{jk_{aj}}+p_a^{(1)}\}$$

$$m_2^*=\max_{a\in A_j}\{-q_{jk_{aj}}+p_a^{(2)}\},\quad M_2^*=\min_{a\in A_j}\{q_{jk_{aj}}+p_a^{(2)}\}.$$

In this case, it follows from the proof of Proposition 15 that any point that belongs to the segment S^j, where

$$S^j=\Big\{(x,y)\in\mathbb{R}^2\colon \exists t\in[0,1]\text{ s.t. }(x,y)=t(m_1,m_2)+(1-t)(M_1,M_2)\Big\}, \tag{6.9}$$

yields an optimal epicenter for disruption j in $v_M(y)$.

Proposition 16. *Suppose that the distance measure is induced by the ℓ^∞ norm. Let polyhedrons $\mathcal{P}$ and $\hat{\mathcal{P}}$ be the LP relaxations of formulations* (3.9) *and* (3.39)*, respectively:*

i. For any $a \in A$, if the corresponding big-M coefficient M_a satisfies that

$$M_a \geq \max_{p \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}], j \in [m]} (\|p_a - p\|_\infty - q_{j1}), \tag{6.10}$$

then $\hat{\mathcal{P}} \subseteq \mathcal{P}$.

ii. In addition, if there is a disruption $j \in [m]$ for which there exist arcs $b, c \in A$ such that $\Gamma^j_{b,c} \neq \emptyset$ and such that for any $a \in A$ the coefficient M_a also satisfies that

$$M_a > 2 \left(\max_{p \in [L^{(1)}, U^{(1)}] \times [L^{(2)}, U^{(2)}], j' \in [m], k \in [r_{j'}]} \left| \|p_a - p\|_\infty - q_{j'k} \right| \right), \tag{6.11}$$

then $\hat{\mathcal{P}} \subsetneq \mathcal{P}$.

Proof. The proof of part (ii) is the same as the proof of part (ii) of Proposition 8. On the other hand, for part (i) let $\hat{\lambda} \in \hat{\mathcal{P}}$. We seek to find $\alpha_j^{(1)}$ and $\alpha_j^{(2)}$ such that

$$-q_{jk_a} - M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(1)} \leq \alpha_j^{(1)} \leq q_{jk_a} + M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(1)} \qquad \forall a \in A \tag{6.12a}$$

$$-q_{jk_a} - M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(2)} \leq \alpha_j^{(2)} \leq q_{jk_a} + M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(2)} \qquad \forall a \in A. \tag{6.12b}$$

Let $\max_{a \in A}\{-q_{jk_a} - M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(1)}\} = -q_{jk_e} - M_e(1 - \hat{\lambda}_{ejk_e}) + p_e^{(1)}$, $\min_{a \in A}\{q_{jk_a} + M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(1)}\} = q_{jk_f} + M_f(1 - \hat{\lambda}_{fjk_f}) + p_f^{(1)}$, $\max_{a \in A}\{-q_{jk_a} - M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(2)}\} = -q_{jk_g} - M_g(1 - \hat{\lambda}_{gjk_g}) + p_g^{(2)}$, and $\min_{a \in A}\{q_{jk_a} + M_a(1 - \hat{\lambda}_{ajk_a}) + p_a^{(2)}\} = q_{jk_h} + M_h(1 - \hat{\lambda}_{hjk_h}) + p_h^{(2)}$.

Therefore, we must have

$$(p_e^{(1)} - p_f^{(1)}) - M_e(1 - \hat{\lambda}_{ejk_e}) - M_f(1 - \hat{\lambda}_{fjk_f}) \leq q_{jk_e} + q_{jk_f} \tag{6.13a}$$

$$(p_h^{(2)} - p_g^{(2)}) - M_g(1 - \hat{\lambda}_{gjk_g}) - M_h(1 - \hat{\lambda}_{hjk_h}) \leq q_{jk_g} + q_{jk_h}. \tag{6.13b}$$

If $(k_e, k_f) \notin \Gamma^j_{e,f}$ and $(k_g, k_h) \notin \Gamma^j_{g,h}$, then (6.13a) and (6.13b) trivially hold. Otherwise, suppose that $(k_e, k_f) \in \Gamma^j_{e,f}$ and thus that $\hat{\lambda}_{ejk_e} + \hat{\lambda}_{fjk_f} \leq 1$. Let $M = \min\{M_e, M_f\}$, then we have

$$\begin{aligned}(p_e^{(1)} - p_f^{(1)}) - M_e(1 - \hat{\lambda}_{ejk_e}) - M_f(1 - \hat{\lambda}_{fjk_f}) &\leq (p_e^{(1)} - p_f^{(1)}) - M(2 - \hat{\lambda}_{ejk_e} - \hat{\lambda}_{fjk_f}) \quad (6.14)\\ &\leq (p_e^{(1)} - p_f^{(1)}) - M \leq 0 \leq q_{jk_e} + q_{jk_f}. \quad (6.15)\end{aligned}$$

where the second to last inequality holds from Equation (6.10). Thus, the inequality (6.13a) is true; the validity of (6.13b) follows by similar arguments assuming $(k_g, k_h) \in \Gamma^j_{g,h}$. □

6.3 Strong formulation for the linear model

Here we show that an analogous version of the strong formulation for $v_M(y)$ is equivalent to $v_L(y)$. This assertion follows by noting that the only difference between the formulations of $v_M(y)$ and $v_L(y)$ is that the coupling constraints (3.9c) are more restrictive in the maximum model than constraints (3.5c) in the linear model. This difference, however, does not invalidate the proofs of any of the results that were discussed in Section 3.4. For convenience, we summarize this observation in the following proposition.

Proposition 17. *Let a flow $y \in \mathbb{R}_+^{|A|}$ and a disruption $j \in [m]$ be given. If the distance measure is induced by the ℓ^1 or ℓ^∞ norm then $v_L(y)$ can be computed by solving the following MIP:*

$$
\begin{aligned}
v_L(y) = \max\ & c^\top y && && && \text{(6.16a)}\\
s.t.\ & c_a = c_a^0 + \sum_{j\in[m]} \sum_{k\in[r_j]} d_{jk}\lambda_{ajk} && && \forall a \in A && \text{(6.16b)}\\
& \sum_{k\in[r_j]} \lambda_{ajk} \le 1 && && \forall a \in A,\ j \in [m] && \text{(6.16c)}\\
& \lambda_{ajk_1} + \lambda_{bjk_2} \le 1 && \forall (k_1,k_2) \in \Gamma^j_{a,b}, a, b \in A, a \neq b, && j \in [m] && \text{(6.16d)}\\
& \lambda_{ajk} \in \{0,1\} && && \forall a \in A,\ j \in [m], k \in [r_j] && \text{(6.16e)}\\
& c_a \in \mathbb{R} && && \forall a \in A. && \text{(6.16f)}
\end{aligned}
$$

Moreover, under the assumptions in equations (3.41) *and* (3.42) *for the ℓ^1-induced distance (or equations* (6.10) *and* (6.11) *for the ℓ^∞-induced distance), formulation* (6.16) *is stronger and strictly stronger, respectively, than formulation* (3.5).

Clearly, formulation (6.16) is also valid with only one disruption, which means that the decomposition method to solve $v_L(y)$ can solve MIP (6.16) for each disruption rather than the polynomial time algorithm. In our numerical experiments we show that this approach can sometimes be slightly faster than the polynomial time algorithm.

VITA

Mehdi Ansari Hadipour

Candidate for the Degree of

Doctor of Philosophy

Dissertation: ROBUST OPTIMIZATION MODELS WITH MIXED-INTEGER UNCERTAINTY SETS

Major Field: Industrial Engineering and Management

Biographical:

Education:

Completed the requirements for the Doctor of Philosophy in Industrial Engineering and Management at Oklahoma State University, Stillwater, Oklahoma in May 2023.

Completed the requirements for the Master of Science in Industrial Engineering at Sharif University of Technology, Tehran, Iran in 2017.

Completed the requirements for the Bachelor of Science in Industrial Engineering at Sharif University of Technology, Tehran, Iran in 2015.