Submitted to *INFORMS Journal on Computing*
manuscript (number XXXX)

A General Model and Efficient Algorithms for Reliable Facility Location Problem under Uncertain Disruptions

Yongzhen Li
Department of Management Science and Engineering, School of Economics and Management, Southeast University, Nanjing, Jiangsu 210096, China, yongzhen_li@hotmail.com,

Xueping Li
Department of Industrial and Systems Engineering, The University of Tennessee at Knoxville, Knoxville, TN 37996, US, Xueping.Li@utk.edu,

Jia Shu
Department of Management Science and Engineering, School of Economics and Management, Southeast University, Nanjing, Jiangsu 210096, China, jshu@seu.edu.cn,

Miao Song
Department of Logistics and Maritime Studies, Faculty of Business, The Hong Kong Polytechnic University, Hong Kong, China, miao.song@polyu.edu.hk,

Kaike Zhang
Department of Industrial and Systems Engineering, The University of Tennessee at Knoxville, Knoxville, TN 37996, US, kaikezhang@gmail.com,

This paper studies the reliable uncapacitated facility location problem in which facilities are subject to uncertain disruptions. A two-stage distributionally robust model is formulated, which optimizes the facility location decisions so as to minimize the fixed facility location cost and the expected transportation cost of serving customers under the worst-case disruption distribution. The model is formulated in a general form, where the uncertain joint distribution of disruptions is partially characterized and is allowed to have any pre-specified dependency structure. This model extends several related models in the literature, including the stochastic one with explicitly given disruption distribution and the robust one with moment information on disruptions. An efficient cutting plane algorithm is proposed to solve this model, where the separation problem is solved respectively by a polynomial-time algorithm in the stochastic case and by a column generation approach in the robust case. Extensive numerical study shows that the proposed cutting plane algorithm not only outperforms the best-known algorithm in the literature for the stochastic problem under independent disruptions but also efficiently solves the robust problem under correlated disruptions. The practical performance of the robust models is verified in a simulation based on historical typhoon data in China. The numerical results further indicate that the robust model with even a small amount of information on disruption correlation can mitigate the conservativeness and improve the location decision significantly.

Key words: Uncapacitated facility location, Uncertain facility disruptions, Stochastic and distributionally robust optimizations, Cutting plane, Column generation
History:

1. Introduction

The uncapacitated facility location problem (UFLP) is one of the most well-studied location problems. It is concerned with how to locate facilities among a given set of potential locations and how to serve a given set of customers with known demand rate using these facilities. The objective usually minimizes the total cost, including the fixed cost of setting up facilities and the transportation cost of serving customers. In the classical UFLP, facilities are assumed to be always available once constructed. However, in reality, facilities may fail from time to time due to disruptive events, such as natural disasters, terrorist attacks, or labor strikes. Though facility disruption is rare, once it occurs, serious interruptions and huge losses can occur. For example, in 2005, Hurricane Katrina shut down the oil refineries in the Gulf of Mexico, US. About 2 million barrels per day of refining capacity was lost initially, and it took months before oil production and refining were fully restored (Bamberger and Kumins 2005, Cashell and Labonte 2005). One example of facility disruption in the manufacturing industry was a fire in March 2000 at the Philips microchip factory in Albuquerque, New Mexico, US, which supplied Ericsson with microchips for its mobile phones. The fire left Ericsson millions of chips short, which interrupted the production of phones and eventually brought a potential revenue loss of at least \$400 million (Latour 2001). More examples in supply chain management can be found in Christopher and Peck (2004), Sheffi (2001), and Tang (2006). In view of the damage and losses resulted from facility disruptions, it is crucial to locate facilities wisely and design a reliable supply chain network that can reliably withstand facility disruptions.

In recent decades, the reliable facility location problem under uncertain facility disruptions has been studied extensively. The two categories of existing works are based on differing assumptions about facility disruptions. Some works assume the facility disruptions occur independently, which is a reasonable assumption for disruptions caused by, for example, facility contamination or plant fire. Stochastic approaches are generally applied in these works, which sometimes suffer from computational efficiency. The other category of works considers natural disaster disruptions such as floods or earthquakes, which simultaneously affect multiple facilities in the area. These works assume the facilities disrupt with correlations and solve this problem by either scenario-based stochastic approach or robust optimization, the latter of which is sometimes criticized as overly conservative. In this paper, we propose a general two-stage distributionally robust model, which bridges the stochastic problem under independent disruptions and the robust problem under correlated disruptions. Efficient algorithms are proposed to solve this model. Moreover, we also analyze the value of explicit correlation information in mitigating the conservativeness of robust approaches.

This paper examines the reliable uncapacitated facility location problem (RUFLP) with the consideration of uncertain facility disruptions. The disruptions are described as a random binary vector with uncertain joint distribution belonging to a preset ambiguity set. The ambiguity set is constructed according to the features of disruptive events considered in reliable facility location problem. Note that geographically close locations are more likely to be affected by the same disruptive event, so are the adjacent facilities in the supply chain. Therefore, the ambiguity set is characterized by given pieces of information, where each piece of information corresponds to the probability of a disruptive event and the associated sets of affected/unaffected facilities. For instance, consider the RUFLP under the threat of typhoons. The affected areas by a typhoon are closely related to the landing place, the landing intensity, and the typhoon track. Thus, the ambiguity set can be defined based on the frequency of typhoons landed in certain locations and the set of affected locations along the typhoon tracks, which are available from the historical data. More importantly, as can be seen in Section 3, the proposed ambiguity set generalizes several ambiguity sets in the literature and can capture various dependency structures for disruptions.

We also notice that, in addition to uncertain facility disruptions, some relevant works on disruptive events also consider other kinds of uncertainties, including demand (An et al. 2014), facility coverage (Lutter et al. 2017, Santos et al. 2019), facility capacity (Ahmadi-Javid and Seddighi 2013), and transportation connection (Azad et al. 2013). This paper focuses on uncertain facility disruptions, assuming that all the other parameters are deterministic. Particularly, demand is assumed to be deterministic. We claim that this paper is readily extended to the case when the uncertain demand is independent of the facility disruption. This independent demand case applies for many supply chain networks for general goods where the demand is determined exogenously by the market and quite irrelevant to the facility disruptions. For instance, Thailand suffered from severe floods during the 2011 monsoon season. As approximately 43% of the world's hard disk drives were produced in Thailand, the floods significantly reduced the hard disk drive shipment in Thailand and severely affected the electronic companies all around the world (Haraguchi and Lall 2015).

However, according to Chongvilaivan (2012), "the global demand remains robust." More recently, Typhoon Mangkhut, the largest storm in the world in 2018, made landfall in Guangdong, China and adversely affected many companies with suppliers and partners located in this area. Despite the supply disruption, some affected companies (e.g., Pop! Promos, a company supplying promotional products) faced stable orders and maintained the ability to fulfill orders from redundant sources (Ruvo 2018). The independence between demand and disruption is also widely assumed in the location-inventory problems with uncertain demand and disruption (Qi et al. 2010). Moreover, the disruption-correlated uncertain demand can be easily incorporated in our approach if (1) the stochastic demand is modeled by scenarios depending on uncertain disruptions in the scenario-based stochastic model, or (2) the demand is defined as a linear function of uncertain disruptions, e.g., the settings in An et al. (2014) and Azad and Hassini (2019). More details of these generalizations can be found in Appendix B.1.

In the literature, the reliable facility location problems considering facility disruptions are generally built upon certain classical models (Snyder et al. 2006). This paper is based on the UFLP, one of the most classical and commonly-used facility location models. Thus, we consider the facility location decision and the transportation decision. Relevant works based on other facility location models also consider the demand allocation decision (Azad et al. 2013), the inventory positioning decision (Qi et al. 2010, most of which are based on the location-inventory problems), as well as the vehicle routing decision (Xie et al. 2015b, most of which are based on the location-routing problems). In this paper, we formulate a two-stage distributionally robust model with a preset ambiguity set for the distribution of disruptions. The facility location decision is made in the first stage before the disruptions, and the transportation decision is made in the second-stage with the realized disruption information. The objective is to minimize the fixed facility setup cost and the expected transportation cost of serving customers under the worst-case disruption distribution.

The RUFLP in this paper is clearly NP-hard as it takes the UFLP, the well-known NP-hard problem, as a special case. The distributionally robust model formulated in the general form makes this problem more difficult. To solve this problem, we develop a cutting plane algorithm, where the separation problem is exactly the evaluation of the worst-case expected transportation cost. For the stochastic case, the separation problem can be solved by a proposed polynomial-time algorithm under the assumption that the conditional disruption probability can be obtained easily. For the robust case, the separation problem is solved by a column generation approach.

The main contributions of this paper can be summarized as follows.

- We formulate a general model for reliable facility location, which extends several models in the literature, including the stochastic one with explicitly given disruption distribution and the robust one with given marginal and/or cross disruption probability.
- A cutting plane algorithm is developed, where the separation problem is solved respectively by a polynomial-time algorithm in the stochastic case and by a column generation approach in the robust case. The efficiency and effectiveness of the proposed algorithm are validated through extensive numerical experiments.
- The value of considering exact correlation in the robust problem is analyzed by comparing the proposed robust model with the one based on marginal moment information. The comparison indicates that the robust model based on marginal disruption probability is sometimes overly conservative, and even a small amount of information on disruption correlation can improve the location decision significantly.

The rest of this paper is organized as follows. In Section 2, we review the related literature. In Section 3, the distributionally robust model is formulated with the generally characterized ambiguity set for disruption distribution. In Section 4, the cutting plane algorithm is introduced, with its framework presented first, followed by the detailed methods to efficiently evaluate the worst-case expected transportation and penalty cost. Extensive numerical experiments are conducted in Section 5 to validate the performance of the proposed cutting plane algorithm. In the last section, this paper is concluded with the discussions of future works.

2. Literature Review

Facility disruptions are widely considered based on several classical facility location problems, including the UFLP, the set-covering problem, the maximal coverage problem, the p-median problem, the capacitated fixed cost facility location problem, and the multi-allocation hub location problem. To keep this literature review focused, we primarily review the UFLP-based works, i.e., the RUFLP. For the works based on other types of classical facility location problems, we refer to Snyder et al. (2006, 2016) for more detailed and extensive reviews. In what follows, we first review the stochastic and the robust facility location problems, respectively, and then review the distributionally robust optimization approach, which is the methodology adopted in this paper.

2.1. On the Stochastic Facility Location Problems

The RUFLP considering independent disruptions is firstly studied by Snyder and Daskin (2005). They are motivated by a facility location problem of 49 cities in the United States, and find that facility disruption at a certain city may result in a significant increase in transportation costs. By assuming that facility disruptions occur independently with identical probabilities, they formulate the RUFLP into a linear integer program and propose a Lagrangian relaxation algorithm. The assumption of identical disruption probabilities as assumed in Snyder and Daskin (2005) greatly simplifies the model formulation and the solution method of the RUFLP. When this assumption is relaxed, it becomes difficult to calculate the expected transportation cost associated with the disruptions. To address this difficulty, attempts are made by Shen et al. (2011), Cui et al. (2010), Aboolian et al. (2013), etc. Shen et al. (2011) extend the model of Snyder and Daskin (2005) by considering site-dependent disruption probabilities. They propose a nonlinear integer programming model where the expected transportation cost is calculated using highly nonlinear multiplicative terms. Several heuristic algorithms are developed to solve the proposed nonlinear model. Cui et al. (2010) propose another nonlinear mixed integer programming model that allows site-dependent disruption probabilities. By applying certain linearization techniques, they reformulate the nonlinear model into a compact mixed integer linear model, which is then solved by a Lagrangian relaxation algorithm. Aboolian et al. (2013) consider the same nonlinear model as Cui et al. (2010). Instead of using the linearization techniques as in Cui et al. (2010), they develop an approximation algorithm based on local search and cutting plane procedure. They show that their method outperforms the Lagrangian relaxation algorithm proposed by Cui et al. (2010) in both solving time and solution quality.

Another direction is to study the RUFLP by assuming correlated disruption. Liu et al. (2009) and Shen et al. (2011) formulate scenario-based models, which incorporate the correlated disruptions by properly defined scenarios. However, their models suffer from poor numerical efficiency, especially when the number of scenarios increases. Some works study the interdependent facility disruptions based on special physical structures, such as “ripple effect” and “supporting stations.” Liberatore et al. (2012) study the ripple effect, which occurs when the facility disruptions propagate, i.e., the failure in one facility will cause capacity losses in other facilities. The ripple effect is captured by a deterministic two-dimensional correlation matrix describing the pair-wise capacity losses. Li et al. (2013)

and Xie et al. (2015a) introduce an extra layer of "supporting stations," each of which is properly connected to certain facilities, to describe the correlated disruptions. In their work, each supporting station fails independently with identical probabilities, and the failures in supporting stations cause the connected facilities to fail as well. In addition, Li and Ouyang (2010), Berman and Krass (2011), Berman et al. (2013), and Lim et al. (2013) apply the continuum approximation approach to study the RUFLP with correlated disruption, which is approximated as a continuous location problem. In these works, the correlated disruptions are captured by conditional probabilities, beta-binomial distribution (Li and Ouyang 2010), correlation coefficients (Berman and Krass 2011, Berman et al. 2013), or correlated binomial random variables (Lim et al. 2013), respectively.

2.2. On the Robust Facility Location Problems

Thus far, all the aforementioned works consider the reliable facility location problem in stochastic settings, where the disruption probabilities are exactly known. As this paper proposes a general model in the form of a distributionally robust model, we further review the relevant robust models considering correlated disruptions.

In the context of robust reliable facility location, there is a large amount of literature that considers interdiction, i.e., the intentional attack of facilities (e.g., a terrorist strike) designed to cause maximum disruption on a network. These works usually focus on the question of how to fortify existing facilities (i.e., immunize the facilities from disruptions), or how to locate new facilities from a given set of potential sites, so as to minimize the demand-weighted distance from customers to non-disrupted facilities under the worst-case interdiction. The former question is studied in the so-called fortification-interdiction problems (Church and Scaparra 2007, Scaparra and Church 2008a,b, Liberatore et al. 2011, 2012), while the latter is studied in the location-interdiction problems (O'Hanley and Church 2011, An et al. 2014). Most of these works formulate robust models following the min-max-min framework, where the innermost problem minimizes the demand-weighted distance from customers to non-disrupted facilities, the problem in the middle represents the actions of an interdictor who attempts to maximize the minimal distance through interdictions, and the outermost problem minimizes the consequence of the worst-case interdiction through fortification/location decisions. These interdiction models differ from ours in two aspects. Firstly, rather than considering the intentional interdiction endogenously caused by a malicious agent, we are concerned with the probabilistic disruption exogenously caused by natural events. We refer to Golany et al. (2009) for a discussion of the differences between the intentional interdiction and the probabilistic disruption. Secondly, all of these interdiction models minimize the cost associated with the worst-case interdiction over all possible scenarios, whereas our model minimizes the expected cost corresponding to the worst-case distribution.

More recently, Lu et al. (2015) propose a distributionally robust reliable facility location model, which is closely related to this paper and to our knowledge is the only distributionally robust model on the RUFLP. In their work, the facility disruptions are allowed to be correlated with an uncertain joint distribution captured by the marginal disruption probability of each facility. They optimize the expected transportation and penalty cost under the worst-case distribution as we do. By proving the worst-case distribution, their robust model is simplified into a stochastic model, which is solved by the standard Benders decomposition method. Our work extends Lu et al. (2015) by considering a general ambiguity set for disruption distribution, which is not limited to the one specified by the marginal disruption probability of each facility. Besides, the cutting plane algorithm we propose can solve the general robust model without knowing/proving the worst-case distribution. More importantly, based on the general formulation, we could incorporate available correlation information explicitly in the robust model. As can be seen in the numerical study, even a small amount of information on disruption correlation can mitigate the conservativeness of the robust marginal moment model and improve the location decisions significantly.

2.3. On the Distributionally Robust Optimization

In this work, the RUFLP is studied in the framework of distributionally robust optimization (DRO), also known as minimax stochastic programming. DRO was pioneered by Scarf (1958) and studied extensively in recent decades with abundant theoretical analysis (Delage and Ye 2010, Wiesemann et al. 2014, and references therein) and a tremendous amount of applications in many areas such as facility location, inventory management, scheduling, finance, etc. A DRO model typically optimizes the worst-case expectation over all the distributions in a prespecified ambiguity set. Thus, it is widely believed that DRO bridges stochastic programming, which optimizes the expectation under a given distribution, and classical robust optimization, which optimizes the worst-case objective value over all the possible realizations of uncertain parameters in a prespecified uncertainty set.

In the DRO literature, existing works consider a wide range of forms for the ambiguity sets depending on specific context (Shapiro 2006 and references therein). The most popular type of the ambiguity sets is based on moment information, e.g., the exact moments (Bertsimas and Popescu 2005, Bertsimas et al. 2010), the bounds on moments (El Ghaoui et al. 2003, Delage and Ye 2010), or other generalized forms (Wiesemann et al. 2014). In this paper, we consider a specific ambiguity set, which is closely related to the moment-based ambiguity set and reflects the features of disruptive events for facilities. We describe the facility disruptions as a random binary vector. The ambiguity set for its uncertain joint distribution is characterized by certain pieces of information. Each piece of information corresponds to the probability of a disruptive event or scenario and the associated sets of affected and/or unaffected facilities. This enables the proposed ambiguity set to represent any dependency structure for the disruptions. Moreover, the proposed set takes several ambiguity sets in the literature as special cases, including a unique distribution as in stochastic programming and those with exact values or bounds of moments. We notice that more recently Ben-Tal et al. (2013), Bayraksan and Love (2015) and Esfahani and Kuhn (2018) propose ambiguity sets using Phi-divergences, which measure the distance between the uncertain distribution and the true one. Although in certain circumstances the Phi-divergence ambiguity set is believed to be less conservative than the moment-based one, it is difficult to uniquely specify correlations among random variables.

The tractability of a DRO model relies on both the nominal problem and the ambiguity set. For some well-structured problem classes, e.g., the Newsvendor problem, with certain ambiguity sets, the DRO models have closed-form solutions for the worst-case distributions and can be solved by stochastic programming approaches (Scarf 1958, Popescu 2007, Lu et al. 2015). It is more common that the analytical worst-case distribution is not available. In this case, global optimization approaches are applied after reformulating the DRO models into tractable optimization problems, e.g., large-scale convex programs (Wiesemann et al. 2014). This work also falls into this category, which adopts the cutting plane algorithm to efficiently solve the large-scale reformulation. We further refer to Chen et al. (2008), Goh and Sim (2010), See and Sim (2010), Kuhn et al. (2011), and Li et al. (2017) for another stream of works that considers the linear decision rule to solve or approximate the two-stage DRO models.

In this paper, we reformulate the distributionally robust model as a mixed integer linear program based on the Benders decomposition. In the literature, the Benders decomposition is usually applied to solve the two-stage scenario-based stochastic models. Specifically, the two-stage model is decomposed into a large-scale formulation with numerous constraints and then solved by the cutting plane algorithm with valid Benders cuts added iteratively. In this paper, however, we apply the Benders decomposition to solve the DRO model. In the cutting plane algorithm, the separation problem to identify valid cuts is solved by column generation and the classical Benders cut is reformulated and simplified by exploiting the closed-form solution to certain subproblems. We notice that some relevant works also propose cutting plane algorithms to solve robust problems. For example, Zeng and Zhao (2013) present a column-and-constraint generation method to solve a two-stage adjustable robust optimization model. In their algorithm, both constraints and columns are added simultaneously to the master problem in each iteration. The underlying robust model and the idea of the algorithm in their work are very different from those in this paper. Xu et al. (2018) propose a cutting plane algorithm framework to solve moment-based DRO models. This paper considers an ambiguity set more general than the moment-based one.

In sum, this work significantly differs from the existing stochastic models under correlated disruptions. In contrast to the robust models considering uncertain disruptions, this work is essentially different from the interdiction models in terms of the nature of disruptions, and greatly generalizes the only distributionally robust model in the literature.

3. Model Formulation

Consider the problem of locating facilities from a set $J = \{1, ..., |J|\}$ of potential locations to serve a set $I = \{1, ..., |I|\}$ of customers. The demand rate of customer $i \in I$ is d_i. The fixed setup cost to open facility $j \in J$ is f_j and the unit shipment cost from facility j to customer i is c_{ij}. If the demand of customer i is not served, a unit penalty cost c_{i0} is incurred. We assume $c_{i0} \geq c_{ij}$ for all $i \in I$ and $j \in J$, which indicates that any customer should be served as long as there is an available facility.

Facilities are unreliable with unexpected failures. Let $\tilde{\boldsymbol{\xi}} = (\tilde{\xi}_1, ..., \tilde{\xi}_{|J|})^\top$ represent the failure status of all the facilities, where $\tilde{\xi}_j \in \{0, 1\}$ is 1 if facility j is online and 0 if it is disrupted. The set of all possible realizations of $\tilde{\boldsymbol{\xi}}$ is denoted by

$$\Xi := \left\{ (\xi_1, ..., \xi_{|J|})^\top \,|\, \xi_j \in \{0, 1\} \ \forall j \in J \right\} = \{0, 1\}^{|J|}.$$

For any realization $\boldsymbol{\xi} \in \Xi$, let $p_{\boldsymbol{\xi}} := \text{Prob}(\tilde{\boldsymbol{\xi}} = \boldsymbol{\xi})$ be the probability that the corresponding disruptive scenario $\boldsymbol{\xi} \in \Xi$ occurs. A distribution of $\tilde{\boldsymbol{\xi}}$ can then be represented by a vector $\boldsymbol{p}$ of $p_{\boldsymbol{\xi}}$ for all $\boldsymbol{\xi} \in \Xi$. Note that $\boldsymbol{p}$ has $2^{|J|}$ components as the cardinality of Ξ is $2^{|J|}$. Due to the high dimensionality, it is often challenging to determine the distribution $\boldsymbol{p}$ of facility failures. Consequently, we assume that the distribution is partially characterized by n pieces of information. For any $k \in \{1,...,n\}$, the kth piece of information specifies that the probability of all facilities in set A_k being online and all facilities in set B_k being disrupted is within the interval $[\underline{q}_k, \overline{q}_k]$, i.e., $\text{Prob}(\tilde{\xi}_j = 1\,\forall j \in A_k, \tilde{\xi}_j = 0\,\forall j \in B_k) \in [\underline{q}_k, \overline{q}_k]$. Therefore, the distribution $\boldsymbol{p}$ should be contained in the following set P:

$$P := \left\{ \boldsymbol{p} \in [0,1]^{(2^{|J|})} \;\middle|\; \sum_{\substack{\boldsymbol{\xi} \in \Xi \mid \xi_j = 1\,\forall j \in A_k, \\ \xi_j = 0\,\forall j \in B_k}} p_{\boldsymbol{\xi}} \in [\underline{q}_k, \overline{q}_k] \;\forall k \in \{1,...,n\},\; \sum_{\boldsymbol{\xi} \in \Xi} p_{\boldsymbol{\xi}} = 1 \right\}. \tag{1}$$

To the best of our knowledge, the definition of P generalizes the characterization of the disruption distribution in several existing works. Some special cases are discussed as follows.

- Stochastic model. The case with a completely known distribution $\boldsymbol{p}$ can be viewed as P being a singleton defined by $2^{|J|}$ pieces of information, i.e., $n = 2^{|J|}$ and $\underline{q}_k = \overline{q}_k$ for all $k = 1,...,n$ in (1). Specially for the scenario-based stochastic model with S scenarios, P can be viewed as a singleton defined by S pieces of information. In this case, we have $n = S$ and $\underline{q}_k = \overline{q}_k$ for all $k = 1,...,S$, which corresponds to the probability for scenario k. The sets A_k and B_k correspond to the set of online and offline facilities in scenario k, respectively.

- Marginal distribution model. In this case, P can be characterized by $|J|$ pieces of information, each of which specifies the marginal probability for a facility to be online. More specifically, we have $\text{Prob}(\tilde{\xi}_j = 1) = q_j$ for any $j \in J$ and hence

$$P = \left\{ \boldsymbol{p} \in [0,1]^{(2^{|J|})} \;\middle|\; \textstyle\sum_{\boldsymbol{\xi} \in \Xi \mid \xi_j = 1} p_{\boldsymbol{\xi}} = q_j \;\forall j \in J,\; \sum_{\boldsymbol{\xi} \in \Xi} p_{\boldsymbol{\xi}} = 1 \right\}.$$

In other words, we have $n = |J|$, $A_k = \{k\}$, $B_k = \emptyset$ and $\underline{q}_k = \overline{q}_k = q_k$ for all $k = 1,...,|J|$ in (1). Note that Lu et al. (2015) consider the same characterization of the disruption probability.

- Moment model. Note that the κth cross moment of the random variables $\tilde{\xi}_j$ where $j \in \{j_1,...,j_\kappa\} \subseteq J$ is $\mathbb{E}_{\boldsymbol{p}}\left[\prod_{j=j_1}^{j_\kappa} \tilde{\xi}_j\right] = \text{Prob}(\tilde{\xi}_j = 1\,\forall j \in \{j_1,...,j_\kappa\})$. Thus, the set P can be used to represent the set of distributions specified by the moments of $\tilde{\boldsymbol{\xi}}$. In particular, suppose that the marginal moment of $\tilde{\xi}_j$ is q_j for any $j \in J$, while the cross moment of

$\tilde{\xi}_{j_1}$ and $\tilde{\xi}_{j_2}$ for any $j_1, j_2 \in J$ and $j_1 < j_2$ is $q_{j_1 j_2}$. Then the set P specified by the first two moments can be written as

$$P = \left\{ \boldsymbol{p} \in [0,1]^{(2^{|J|})} \;\middle|\; \begin{array}{ll} \sum_{\boldsymbol{\xi} \in \Xi | \xi_j = 1} p_{\boldsymbol{\xi}} = q_j, & \forall j \in J, \\ \sum_{\boldsymbol{\xi} \in \Xi | \xi_{j_1} = \xi_{j_2} = 1} p_{\boldsymbol{\xi}} = q_{j_1 j_2}, & \forall j_1, j_2 \in J, j_1 < j_2, \\ \sum_{\boldsymbol{\xi} \in \Xi} p_{\boldsymbol{\xi}} = 1 & \end{array} \right\}, \tag{2}$$

i.e., we have $n = |J|(|J|+1)/2$ in (1). The ambiguity set P in (1) is more general than that in (2) as (1) can also be used to model the cases with higher moments, partial moment information, and/or probabilities falling in certain intervals.

Based on the available information, we need to set up facilities to serve the customers. The facility setup decision is denoted by $\boldsymbol{x} = (x_1, ..., x_{|J|})^\top$, where $x_j \in \{0,1\}$ is 1 if facility j is set up and 0 otherwise. To decide $\boldsymbol{x}$, all the cost components, including the fixed setup cost f_j to open facility $j \in J$, the unit transportation cost c_{ij} from facility $j \in J$ to customer $i \in I$, and the unit penalty cost c_{i0} when customer $i \in I$ is not served, should be taken into consideration at the same time. As all the facilities are uncapacitated, any customer should be served by the closest open facility that is not disrupted. Therefore, given the decision $\boldsymbol{x}$ and the realization $\boldsymbol{\xi}$ of the disruption status, the total transportation and penalty cost to serve all the customers is

$$Q(\boldsymbol{x}, \boldsymbol{\xi}) := \sum_{i \in I} d_i \min_{j \in J \cup \{0\} | x_j \xi_j = 1} c_{ij},$$

where we assume $x_0 \equiv 1$ and $\xi_0 \equiv 1$. The unit penalty cost c_{i0} is incurred only when all the open facilities are disrupted. Recall that the disruption status $\tilde{\boldsymbol{\xi}}$ is uncertain. If the distribution of $\tilde{\boldsymbol{\xi}}$ is completely known, we can take into account the expectation of $Q(\boldsymbol{x}, \tilde{\boldsymbol{\xi}})$ when deciding $\boldsymbol{x}$. However, the distribution of $\tilde{\boldsymbol{\xi}}$ can only be characterized by the set P. Thus, we instead consider the worst-case expectation of $Q(\boldsymbol{x}, \tilde{\boldsymbol{\xi}})$ among all the distributions in the set P, i.e., $\max_{\boldsymbol{p} \in P} \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}, \tilde{\boldsymbol{\xi}})]$. As a result, an optimal decision $\boldsymbol{x}$ should minimize both the fixed setup cost and the worst-case expected transportation and penalty cost, which leads to the following robust optimization problem:

$$\mathcal{P}: \quad \min_{\boldsymbol{x} \in \{0,1\}^{|J|}} \left\{ \sum_{j \in J} f_j x_j + \max_{\boldsymbol{p} \in P} \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}, \tilde{\boldsymbol{\xi}})] \right\}. \tag{3}$$

Obviously, model (3) is an NP-hard problem, as it generalizes the UFLP, which is a well-known NP-hard problem. The NP-hardness implies that no polynomial-time algorithm exists for model (3) unless P=NP. However, we can reformulate model (3) into a mixed integer linear program with a large number of constraints and then apply the cutting plane algorithm to solve it.

4. The Cutting Plane Algorithm

Let η replace the worst-case expected transportation and penalty cost, then model (3) turns into

$$\mathcal{P}: \quad \min_{\boldsymbol{x}\in\{0,1\}^{|J|},\eta\in\mathbb{R}} \left\{ \sum_{j\in J} f_j x_j + \eta \,\middle|\, \eta \geq \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x},\tilde{\boldsymbol{\xi}})], \quad \forall \boldsymbol{p}\in P \right\}. \tag{4}$$

We notice that, when a stochastic model is considered, i.e., when P is defined as a singleton containing a completely known distribution, there is exactly one constraint in model (4). However, when a robust model is considered where P is generally defined as in (1), there are an infinite number of constraints in model (4) as P contains an infinite number of distributions. This makes model (4) a difficult problem to solve.

To solve model (4) when P is generally defined, we propose a cutting plane algorithm. As we shall see, the performance of the proposed cutting plane algorithm relies on the efficient evaluation of $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x},\tilde{\boldsymbol{\xi}})]$, i.e., the worst-case expected transportation and penalty cost. Therefore, in the remainder of this section, we firstly present the general framework of the proposed cutting plane algorithm as follows, and then show how to evaluate $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x},\tilde{\boldsymbol{\xi}})]$ efficiently.

Algorithm CP

Step 0. Consider a finite set Λ, each member of whom is a collection $(\boldsymbol{p},\boldsymbol{y})$ such that $\boldsymbol{p}\in P$ and $\boldsymbol{y}\in\{0,1\}^{|J|}$, i.e., $\Lambda\subseteq\left\{(\boldsymbol{p},\boldsymbol{y})\,\middle|\,\boldsymbol{p}\in P, \boldsymbol{y}\in\{0,1\}^{|J|}\right\}$.

Step 1. Solve the following problem

$$\begin{aligned}
\mathcal{RP}: \min_{\substack{\boldsymbol{x}\in\{0,1\}^{|J|},\\ \eta\in\mathbb{R}}} \quad & \sum_{j\in J} f_j x_j + \eta \\
\text{s.t.} \quad & \eta \geq \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{y},\tilde{\boldsymbol{\xi}})] + \sum_{j\in J}\Big(\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{y}\vee\boldsymbol{z}_j,\tilde{\boldsymbol{\xi}})] - \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{y},\tilde{\boldsymbol{\xi}})]\Big)\cdot x_j, \\
& \hspace{20em} \forall(\boldsymbol{p},\boldsymbol{y})\in\Lambda,
\end{aligned}$$

and obtain an optimal solution $(\boldsymbol{x}^*,\eta^*)$. In $\mathcal{RP}$, $\boldsymbol{z}_j$ denotes the vector of the same dimension as $\boldsymbol{y}$ such that $z_j=1$ and $z_l=0$ for all $l\neq j$, and "$\vee$" represents the operation of taking componentwise maximum.

Step 2. Evaluate the worst-case expected transportation and penalty cost, and obtain the worst-case distribution, i.e., solve the separation problem $Z_{sep}(\boldsymbol{x}^*) := \max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ and obtain the optimal solution $\boldsymbol{p}^*$. Particularly, if the stochastic model is considered where P is defined as a singleton, e.g., $P=\{\boldsymbol{p}^\dagger\}$, we have $Z_{sep}(\boldsymbol{x}^*)=\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ and $\boldsymbol{p}^*=\boldsymbol{p}^\dagger$.

Step 3. If $\eta^*\geq Z_{sep}(\boldsymbol{x}^*)$, return $\boldsymbol{x}^*$ as the optimal solution to model (4). Otherwise, i.e., $\eta^*<Z_{sep}(\boldsymbol{x}^*)$, add $(\boldsymbol{p}^*,\boldsymbol{x}^*)$ into Λ, or equivalently, add the valid cut

$$\eta\geq\mathbb{E}_{\boldsymbol{p}^*}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]+\sum_{j\in J}\Big(\mathbb{E}_{\boldsymbol{p}^*}[Q(\boldsymbol{x}^*\vee\boldsymbol{z}_j,\tilde{\boldsymbol{\xi}})]-\mathbb{E}_{\boldsymbol{p}^*}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]\Big)\cdot x_j \tag{5}$$

into $\mathcal{RP}$. Go to Step 1.

The following theorem guarantees that Algorithm CP provides the exact solution to model (4).

THEOREM 1. *Algorithm CP solves model* (4) *to optimality within a finite number of iterations.*

The proof of Theorem 1 can be found in Appendix A.1. In the proof, model (4) is firstly reformulated according to Benders decomposition. The reformulation is then written as a tractable mixed integer linear program, which is exactly model $\mathcal{RP}$ including all valid cuts in the form of (5).

In Algorithm CP, Step 1 is to solve problem $\mathcal{RP}$, which is a mixed integer linear program with a moderate number ($|\Lambda|$) of constraints and hence readily solved by commercial solvers like CPLEX. Step 2 is to evaluate $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ for given $\boldsymbol{x}^*$, while Step 3 is to formulate the valid cut (5) based on $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$. Apparently, both Step 2 and Step 3 can be easily implemented, as long as $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ can be evaluated efficiently.

To develop an efficient approach to evaluate $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$, we notice that different situations are encountered depending on the definition of P. To be more specific, if a robust model is considered where P is generally defined as in (1), the separation problem $Z_{sep}(\boldsymbol{x}^*)$ turns into a linear program with an exponential number of decision variables, which happens not to be easy. On the other hand, if a stochastic model is considered where $P=\{\boldsymbol{p}^\dagger\}$, we do not need to solve $Z_{sep}(\boldsymbol{x}^*)$ because the disruption distribution is completely known. However, evaluating $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ may be difficult because $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]=\sum_{\boldsymbol{\xi}\in\Xi|p^\dagger_{\boldsymbol{\xi}}>0}Q(\boldsymbol{x}^*,\boldsymbol{\xi})$ and $\boldsymbol{p}^\dagger$ may contain a large number of non-zero entries. As a typical example, if the disruption of each facility is independent, as assumed in most existing works, then $\boldsymbol{p}^\dagger$ consists of $2^{|J|}$ non-zero entries.

In view of these different situations, we discuss the evaluation of $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ for robust and stochastic models in the following two subsections, respectively.

4.1. Separation Problem of the Stochastic RUFLP

When a stochastic model is considered, where $P=\{\boldsymbol{p}^\dagger\}$, the separation problem can be simplified as $Z_{sep}(\boldsymbol{x}^*)=\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$. Therefore, rather than solving the optimization problem as in the robust case, we only need to evaluate $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$. Note that $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]=\sum_{\boldsymbol{\xi}\in\Xi|p^\dagger_{\boldsymbol{\xi}}>0}Q(\boldsymbol{x}^*,\boldsymbol{\xi})$. If $|\{\boldsymbol{\xi}\in\Xi|p^\dagger_{\boldsymbol{\xi}}>0\}|$ is moderate, e.g., the scenario-based stochastic model with a moderate number of scenarios, we can evaluate $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ directly by simple summation. For instance, if $S=|\{\boldsymbol{\xi}\in\Xi|p^\dagger_{\boldsymbol{\xi}}>0\}|$, then the evaluation of $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ can be completed in $O(S|I||J|)$. However, in more general cases, $\boldsymbol{p}^\dagger$ may contain numerous non-zero entries, e.g., $2^{|J|}$ non-zero entries in the example with independent disruptions, which makes the evaluation of $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ very tricky.

To address this difficulty, we assume that the conditional disruption probabilities, i.e., the disruption probability of a facility given that some other facilities are disrupted, can be calculated in $O(1)$. Several examples fall into this category including the one with independent disruptions, the one with a closed-form formula to compute the probability for any subset of facilities to fail simultaneously, and another where the correlations of disruptions are induced from shared hazard exposure and follow certain known patterns as assumed in Li and Ouyang (2010). With this assumption, the evaluation of $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ can be completed efficiently by Algorithm 1.

In Algorithm 1, variable γ stores the joint disruption probability of facilities $j_1,...,j_t$, i.e., $\gamma=\text{Prob}(\xi_{j_1}=0,...,\xi_{j_t}=0)$. Step A is to sort a list of at most $|J|$ facilities for $|I|$ times, thus Step A runs in $O(|I||J|\log|J|)$. As we assume that the conditional probability $\text{Prob}(\xi_{j_t}=0|\xi_{j_1}=0,...,\xi_{j_{t-1}}=0)$ can be obtained in $O(1)$, Step B runs in $O(|J|)$ and hence Lines 5-11 run in $O(|I||J|)$. To summarize, Algorithm 1 runs in $O(|I||J|\log|J|+|I||J|)$, i.e., $O(|I||J|\log|J|)$.

To implement Algorithm CP for the stochastic model, we must discuss the algorithmic details in formulating the valid cut (5) for given $\boldsymbol{x}^*$ and $\boldsymbol{p}^\dagger$. On one hand, for the scenario-based stochastic model, if $S=|\{\boldsymbol{\xi}\in\Xi|p^\dagger_{\boldsymbol{\xi}}>0\}|$ is moderate, then it is trivial that

---

Algorithm 1 Evaluate $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*, \tilde{\boldsymbol{\xi}})]$ for given $\boldsymbol{x}^*$ and $\boldsymbol{p}^\dagger$

Input: The facility location decision $\boldsymbol{x}^*$ and the disruption distribution $\boldsymbol{p}^\dagger$.

Output: The value of $\mathbb{E}_{\boldsymbol{p}^\dagger}[Q(\boldsymbol{x}^*, \tilde{\boldsymbol{\xi}})]$

1: **for** each customer $i \in I$ **do** ▷ Step A (Lines 1-3)

2: Sort all the facilities in $\{j \in J | x_j^* = 1\} \cup \{0\}$ in the nondecreasing order of unit transportation/penalty cost. Let $N := \sum_{j \in J} x_j^*$, i.e., the number of open facilities in decision $\boldsymbol{x}^*$, and denote $L_i := \{j_1, ..., j_{N+1}\}$ as the list of indexes of facilities satisfying $c_{ij_1} \le ... \le c_{ij_{N+1}}$.

3: **end for**

4: $cost \leftarrow 0$

5: **for** each customer $i \in I$ **do**

6: $\gamma \leftarrow 1$

7: **for** $j_t \in L_i$, i.e., $t = 1, ..., N+1$ **do** ▷ Step B (Lines 7-10)

8: $cost \leftarrow cost + \gamma \times (1 - \text{Prob}(\xi_{j_t} = 0 | \xi_{j_1} = 0, ..., \xi_{j_{t-1}} = 0)) \times d_i c_{ij_t}$

9: $\gamma \leftarrow \gamma \times \text{Prob}(\xi_{j_t} = 0 | \xi_{j_1} = 0, ..., \xi_{j_{t-1}} = 0)$

10: **end for**

11: **end for**

12: **return** $cost$

---

the valid cut (5) can be formulated in $O(S|I||J|^2)$. On the other hand, with the aforementioned assumption about conditional disruption probability, we can develop an efficient Algorithm 2 to formulate the valid cut. As discussed before, in Algorithm 2, Line 1 runs in $O(|I||J|\log|J|)$, and Lines 3-6 run in $O(|I||J|)$. Similar to Step B in Algorithm 1, Step C runs in $O(|J|)$ as well, thus Lines 7-17 run in $O(|I||J|^2)$. Therefore, Algorithm 2 runs in $O(|I||J|\log|J| + |I||J| + |I||J|^2)$, i.e., $O(|I||J|^2)$.

4.2. Separation Problem of the Robust RUFLP

When a robust model is considered, the separation problem $Z_{sep}(\boldsymbol{x}) = \max_{\boldsymbol{p} \in P} \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}, \tilde{\boldsymbol{\xi}})]$ for a given $\boldsymbol{x}$ can be written as

$$
\begin{aligned}
Z_{sep}(\boldsymbol{x}) = \max_{p_{\boldsymbol{\xi}} \ge 0} \; & \textstyle\sum_{\boldsymbol{\xi} \in \Xi} Q(\boldsymbol{x}, \boldsymbol{\xi}) p_{\boldsymbol{\xi}} \\
\text{s.t.} \; & \textstyle\sum_{\boldsymbol{\xi} \in \Xi} p_{\boldsymbol{\xi}} = 1, \\
& \textstyle\sum_{\boldsymbol{\xi} \in \Xi | \xi_j = 1 \, \forall j \in A_k, \xi_j = 0 \, \forall j \in B_k} p_{\boldsymbol{\xi}} \le \overline{q}_k, \qquad \forall k = 1, ..., n, \\
& -\textstyle\sum_{\boldsymbol{\xi} \in \Xi | \xi_j = 1 \, \forall j \in A_k, \xi_j = 0 \, \forall j \in B_k} p_{\boldsymbol{\xi}} \le -\underline{q}_k, \; \forall k = 1, ..., n,
\end{aligned}
\tag{6}
$$

where the constraints are from the general definition of P as in (1).

Algorithm 2 Formulate the valid cut (5) for given $\boldsymbol{x}^*$ and $\boldsymbol{p}^\dagger$

Input: The facility location decision $\boldsymbol{x}^*$ and the disruption distribution $\boldsymbol{p}^\dagger$.
Output: The valid cut (5)

1: Run Step A in Algorithm 1 to obtain L_i for all $i \in I$
2: $cost \leftarrow 0$, $cost(j) \leftarrow 0$ for all $j \in J$ such that $x_j^* = 0$
3: **for** each customer $i \in I$ **do**
4: $\quad \gamma \leftarrow 1$
5: $\quad$ Run Step B in Algorithm 1 to calculate $cost$
6: **end for**
7: **for** each $j \in J$ such that $x_j^* = 0$ **do**
8: $\quad$ **for** each customer $i \in I$ **do**
9: $\quad\quad \gamma \leftarrow 1$
10: $\quad\quad$ Insert j into the sorted list L_i
11: $\quad\quad$ **for** $j_t \in L_i$, i.e.,$t = 1, ..., N+2$ **do** ▷ Step C (Lines 11-14)
12: $\quad\quad\quad cost(j) \leftarrow cost(j) + \gamma \times (1 - \text{Prob}(\xi_{j_t} = 0 | \xi_{j_1} = 0, ..., \xi_{j_{t-1}} = 0)) \times d_i c_{ij_t}$
13: $\quad\quad\quad \gamma \leftarrow \gamma \times \text{Prob}(\xi_{j_t} = 0 | \xi_{j_1} = 0, ..., \xi_{j_{t-1}} = 0)$
14: $\quad\quad$ **end for**
15: $\quad\quad$ Remove j from the sorted list L_i
16: $\quad$ **end for**
17: **end for**
18: **return** the valid cut $\eta \geq cost + \sum_{j \in J | x_j^* = 0} (cost(j) - cost) \cdot x_j$

For the separation problem in (6), we have the following theorem, the proof of which can be found in Appendix A.2.

Theorem 2. *Given $\boldsymbol{x} \in \{0,1\}^{|J|}$, the separation problem in* (6) *is NP-hard.*

The NP-hardness of $Z_{sep}(\boldsymbol{x})$ suggests that there is no polynomial-time algorithm for it. Nevertheless, the fact that $Z_{sep}(\boldsymbol{x})$ is a linear program with an exponential number of decision variables motivates us to apply the column generation approach to solve it.

Implementing the column generation approach to solve $Z_{sep}(\boldsymbol{x})$ is identical to implementing the cutting plane approach to solve the corresponding dual problem, i.e., $Z_{sep}^D(\boldsymbol{x})$ as follows:

$$
\begin{aligned}
Z_{sep}^D(\boldsymbol{x}) = \min_{\alpha, \overline{\boldsymbol{\beta}}, \underline{\boldsymbol{\beta}}} \ & \alpha + \textstyle\sum_{k=1}^n (\overline{q}_k \overline{\beta}_k - \underline{q}_k \underline{\beta}_k) \\
\text{s.t.} \ & \alpha + \textstyle\sum_{k \in \{1,\dots,n\} | \xi_j = 1 \forall j \in A_k, \xi_j = 0 \forall j \in B_k} (\overline{\beta}_k - \underline{\beta}_k) \geq Q(\boldsymbol{x}, \boldsymbol{\xi}), \quad \forall \boldsymbol{\xi} \in \Xi, \\
& \overline{\beta}_k \geq 0, \underline{\beta}_k \geq 0, \quad \forall k = 1, \dots, n,
\end{aligned} \tag{7}
$$

where α, $\overline{\beta}_k$ and $\underline{\beta}_k$ for all $k = 1, ..., n$ are the dual variables associated with the constraints in model (6), and $\overline{\boldsymbol{\beta}}$ and $\underline{\boldsymbol{\beta}}$ are the vectors of $\overline{\beta}_k$ and $\underline{\beta}_k$ for all $k = 1, ..., n$, respectively. The column generation approach, or the cutting plane approach, starts from the relaxed problem of model (7) with only a subset of constraints considered. In each iteration, we solve the relaxed problem and obtain the current dual solution $\alpha^*, \overline{\boldsymbol{\beta}}^*, \underline{\boldsymbol{\beta}}^*$. Then, the following pricing problem (also known as the reduced cost problem), denoted as $RC(\boldsymbol{x})$,

$$RC(\boldsymbol{x}) := \max_{\boldsymbol{\xi} \in \Xi} \left\{ Q(\boldsymbol{x}, \boldsymbol{\xi}) - \alpha^* - \sum_{\substack{k \in \{1,...,n\} | \\ \xi_j = 1 \forall j \in A_k, \xi_j = 0 \forall j \in B_k}} (\overline{\beta}_k^* - \underline{\beta}_k^*) \right\}, \tag{8}$$

is solved to identify violated constraints. Note that a special case of $RC(\boldsymbol{x})$ with P characterized by the first two moments is presented in model (15) in the appendix. The objective value of $RC(\boldsymbol{x})$ indicates whether there are violated constraints for the current dual solution. Specifically, if $RC(\boldsymbol{x}) \leq 0$, all the constraints in model (7) are satisfied and the algorithm terminates with the current solution as the optimal solution. Otherwise, i.e., $RC(\boldsymbol{x}) > 0$, then the optimal solution to $RC(\boldsymbol{x})$ provides a violated constraint to be added to the relaxed problem. The updated relaxed problem is then solved, and this process continues and repeats until the optimal solution is returned or the relaxed problem turns out to be infeasible.

In general, the pricing problem $RC(\boldsymbol{x})$ is a highly nonlinear integer program. Furthermore, for $RC(\boldsymbol{x})$ we have the following corollary, which is a direct result from the proof of Theorem 2.

COROLLARY 1. *Given $\boldsymbol{x} \in \{0,1\}^{|J|}$ and the values of dual variables α^*, $\overline{\boldsymbol{\beta}}^*$, and $\underline{\boldsymbol{\beta}}^*$, the problem $RC(\boldsymbol{x})$ is NP-hard.*

When implementing the column generation approach, the pricing problem needs to be solved many times to identify effective columns, or decision variables, thus the efficiency of solving the pricing problem significantly affects the efficiency of the whole algorithm. Although the pricing problem (8) is proved to be NP-hard, indicating that no polynomial-time algorithm exists unless P=NP, we can instead reformulate it into an equivalent integer program with a moderate number of decision variables and constraints, which can be solved satisfactorily by commercial solvers like CPLEX. The equivalent reformulation is presented in the following theorem, the proof of which can be found in Appendix A.3.

THEOREM 3. *Let $A_k = \{a_k^1, ..., a_k^{|A_k|}\}$ and $B_k = \{b_k^1, ..., b_k^{|B_k|}\}$ for any $k \in \{1, ..., n\}$. Then model (8) is equivalent to*

$$
\begin{aligned}
RC(\boldsymbol{x}) = \max_{\substack{\boldsymbol{\xi}\in\Xi,\\ \boldsymbol{\pi},\boldsymbol{\lambda}}} \ & \sum_{i\in I} d_i\pi_i - \alpha - \sum_{k=1}^{n}(\overline{\beta}_k - \underline{\beta}_k)\lambda_k \\
\text{s.t.} \ & \pi_i \le c_{ij}x_j\xi_j + c_{i0}(1 - x_j\xi_j), && \forall i\in I, j\in J,\\
& \lambda_k \le \xi_j, && \forall k\in\{1,...,n\}, j\in A_k,\\
& \lambda_k \le 1-\xi_j, && \forall k\in\{1,...,n\}, j\in B_k,\\
& \lambda_k \ge \textstyle\sum_{j\in A_k}\xi_j + \sum_{j\in B_k}(1-\xi_j) - |A_k| - |B_k| + 1, && \forall k\in\{1,...,n\},\\
& \lambda_k \ge 0, && \forall k\in\{1,...,n\},
\end{aligned}
\tag{9}
$$

where $\boldsymbol{\pi}$ and $\boldsymbol{\lambda}$ are the vectors of π_i and λ_k, respectively.

According to Theorem 3, we can equivalently reformulate the pricing problem $RC(\boldsymbol{x})$ in (8), which is a highly nonlinear integer program, as model (9), which is a mixed integer linear program. Although model (9) has more decision variables and constraints than model (8), its numbers of variables and constraints are

$$
|J| + |I| + n \qquad \text{and} \qquad |I||J| + \sum_{k=1}^{n}(|A_k| + |B_k| + 2) \le |I||J| + n(|J| + 2),
$$

respectively, and hence are still polynomial in the input size, i.e., $|I|$, $|J|$ and n. Therefore, model (9) can readily be solved by solvers like CPLEX.

As an illustration of Theorem 3, we consider P defined as in (2), which is characterized by the first two moments of $\tilde{\boldsymbol{\xi}}$, i.e., $n = |J|(|J|+1)/2$. The corresponding $RC(\boldsymbol{x})$ in (15) in the appendix can be reformulated as

$$
\begin{aligned}
RC(\boldsymbol{x}) = \max_{\substack{\boldsymbol{\xi}\in\Xi,\\ \boldsymbol{\pi},\boldsymbol{\lambda},\boldsymbol{\mu}}} \ & \sum_{i\in I} d_i\pi_i - \alpha - \sum_{j\in J}\beta_j\xi_j - \sum_{j_1,j_2\in J, j_1<j_2}\beta_{j_1j_2}\lambda_{j_1j_2} \\
\text{s.t.} \ & \pi_i \le c_{ij}x_j\xi_j + c_{i0}(1 - x_j\xi_j), && \forall i\in I, j\in J,\\
& \lambda_{j_1j_2} \le \xi_{j_1}, && \forall j_1, j_2\in J, j_1<j_2,\\
& \lambda_{j_1j_2} \le \xi_{j_2}, && \forall j_1, j_2\in J, j_1<j_2,\\
& \lambda_{j_1j_2} \ge \xi_{j_1} + \xi_{j_2} - 1, && \forall j_1, j_2\in J, j_1<j_2,\\
& \lambda_{j_1j_2} \ge 0, && \forall j_1, j_2\in J, j_1<j_2.
\end{aligned}
\tag{10}
$$

Thus far, a column generation approach is proposed to evaluate $\max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*, \tilde{\boldsymbol{\xi}})]$ for given $\boldsymbol{x}^*$ and to derive the worst-case distribution $\boldsymbol{p}^*$. To implement Algorithm CP for the robust model, we must first discuss the algorithmic details in formulating the valid cut (5) for given $\boldsymbol{x}^*$ and $\boldsymbol{p}^*$.

Given $\boldsymbol{x}^*$ and $\boldsymbol{p}^*$, a direct way to formulate the valid cut (5) is presented in Algorithm 3. Variables *cost* and *cost*(j) store the values of $\sum_{\boldsymbol{\xi}\in\Xi|p^*_{\boldsymbol{\xi}}>0} p^*_{\boldsymbol{\xi}} Q(\boldsymbol{x}^*, \boldsymbol{\xi})$ and $\sum_{\boldsymbol{\xi}\in\Xi|p^*_{\boldsymbol{\xi}}>0} p^*_{\boldsymbol{\xi}} Q(\boldsymbol{x}^* \vee \boldsymbol{z}_j, \boldsymbol{\xi})$, respectively. Note that $cost(j) = cost$ for all $j \in J$ such that $x^*_j = 1$. Thus we only need to calculate $cost(j)$ for all $j \in J$ such that $x^*_j = 0$.

Algorithm 3 Formulate the valid cut (5) for given $\boldsymbol{x}^*$ and $\boldsymbol{p}^*$

Input: The facility location decision $\boldsymbol{x}^*$ and the disruption distribution $\boldsymbol{p}^*$.
Output: The valid cut (5)

$cost \leftarrow 0$, $cost(j) \leftarrow 0$ for all $j \in J$ such that $x^*_j = 0$
for each $\boldsymbol{\xi} \in \Xi$ such that $p^*_{\boldsymbol{\xi}} > 0$ **do**
  $Q \leftarrow 0$, $Q(j) \leftarrow 0$ for all $j \in J$ such that $x^*_j = 0$
  for each $i \in I$ **do**
    $c_{\min} \leftarrow c_{i0}$
    for each $j \in J$ such that $x^*_j = 1$ and $\xi_j = 1$ **do**
      $c_{\min} \leftarrow \min\{c_{\min}, c_{ij}\}$
    end for
    $Q \leftarrow Q + d_i c_{\min}$
    for each $j \in J$ such that $x^*_j = 0$ **do**
      if $\xi_j = 1$ **then**
        $Q(j) \leftarrow Q(j) + d_i \min\{c_{\min}, c_{ij}\}$
      else
        $Q(j) \leftarrow Q(j) + d_i c_{\min}$
      end if
    end for
  end for
  $cost \leftarrow cost + p^*_{\boldsymbol{\xi}} Q$, $cost(j) \leftarrow cost(j) + p^*_{\boldsymbol{\xi}} Q(j)$ for all $j \in J$ such that $x^*_j = 0$
end for
return the valid cut $\eta \geq cost + \sum_{j\in J|x^*_j=0} (cost(j) - cost) \cdot x_j$

As P is characterized by n pieces of information, given $\boldsymbol{p}^*$ as the optimal solution to $Z_{sep}(\boldsymbol{x}^*)$, we have $|\{\boldsymbol{\xi} \in \Xi | p^*_{\boldsymbol{\xi}} > 0\}| \leq n+1$, thus Algorithm 3 runs in $O(n|I||J|)$.

Note that, when $n' = |\{\boldsymbol{\xi} \in \Xi | p^\dagger_{\boldsymbol{\xi}} > 0\}|$ is moderate, Algorithm 3 for the robust model can be also applied to the stochastic case, whose computational complexity is $O(n'|I||J|)$. If $n' < |J|$, Algorithm 3 is more efficient than Algorithm 2, which runs in $O(|I||J|^2)$. On the other hand, Algorithm 2, designed for the stochastic model, can be applied to the robust model as well if we make the same assumption that the conditional disruption probabilities can be calculated in $O(1)$. Furthermore, if $|J|$ is significantly smaller than n, then Algorithm 2, which runs in $O(|I||J|^2)$, is more efficient than Algorithm 3, which runs in $O(n|I||J|)$.

5. Numerical Study

In this section, we implement the proposed cutting plane algorithm to solve the stochastic and robust RUFLP, evaluate its performance, and demonstrate its advantages by comparing it with existing algorithms. The numerical study consists of the following two parts:

(i) For the stochastic RUFLP, we consider the instances with independent disruptions following Aboolian et al. (2013). In Section 5.1, we compare the proposed cutting plane algorithm with the search-and-cut (SnC) algorithm by Aboolian et al. (2013), which to our knowledge is among the best algorithms for the problem with independent disruptions.

(ii) For the robust RUFLP, we consider the instances with the marginal moment and the partial cross moment of disruption probabilities. In Section 5.2, we compare the proposed robust model with the stochastic one, as well as the one by Lu et al. (2015), which to our knowledge is the only work considering the distributionally robust model for the reliable facility location problem with prespecified marginal disruption probabilities. To further demonstrate the practical performance of the two robust models, a data-driven simulation is conducted in Section 5.3 based on historical typhoon data in China.

The computation is conducted based on the same data set (from 1990 census data) as used in Aboolian et al. (2013) and Lu et al. (2015). All the computational experiments are coded with C++ and implemented using ILOG CPLEX Academic Initiative Edition 12.7.

5.1. Numerical Results on the Stochastic RUFLP

For the stochastic RUFLP, we implement the proposed cutting plane algorithm based on the same instances as in Aboolian et al. (2013) and compare it with the SnC algorithm by Aboolian et al. (2013).

The computation is conducted based on the instances with 50, 75, and 100 nodes, where each node represents a potential location as well as an aggregated demand point. The demand, the fixed facility setup cost, the unit transportation cost, and the unit emergency cost are set in the same way as in Aboolian et al. (2013) based on the data set. The facility disruption of each node happens independently and the disruption probability $1-q_j$ of facility j is given as $1-q_j = 0.01 + 0.1\alpha e^{-D_j/400}$, which is the same as in Aboolian et al. (2013). Specifically, D_j is the great circle distance in miles between node j and New Orleans, Louisiana, US, the center of Hurricane Katrina (we change the notation d_j in Aboolian et al. (2013) to D_j in this paper to avoid confusion). The value 0.1α can be interpreted as the probability that a disastrous event happens at the source New Orleans. Following Aboolian et al. (2013), we set the value of α from 1.0 to 1.5 with a step length of 0.05, i.e., the occurrence probability of disaster at the source is set from 0.1 to 0.15 with a step length of 0.005. To show the impact of α on the disruption probability, we take the 50-nodes instances as an example, where the minimum, median, and maximum values of D_j for all $j \in J$ are 16, 1023, and 2731 miles, respectively. When α change from 1.0 to 1.5, the corresponding minimal, median, and maximal disruption probabilities at corresponding facilities change from 0.1060, 0.0178, and 0.0101 to 0.1540, 0.0216, and 0.0102, respectively.

The proposed cutting plane algorithm is implemented following the framework of Algorithm CP introduced in Section 4. In each iteration, the relaxed master problem $\mathcal{RP}$ is solved by CPLEX under standard settings for mixed integer linear programs, the expected transportation cost is evaluated by Algorithm 1, and the cut is added by Algorithm 2. For the SnC algorithm proposed by Aboolian et al. (2013), we fix the maximum assignment level R to the number of nodes so that its objective values are comparable with ours. The neighborhood size is set to be 3 to achieve the best performance according to their computational results. Both algorithms are solved to a 0.5% optimality gap or a maximum CPU time of 3600 seconds, whichever occurs first. Specifically, the optimality gap is the relative gap between the best lower bound and the best upper bound so far. For the proposed cutting plane algorithm, the best lower bound is the optimal objective value of the relaxed master problem in the last iteration, and the best upper bound is the objective value corresponding to the best feasible solution so far. For the SnC algorithm by Aboolian et al. (2013), the lower and upper bounds are specially designed and obtained, in short, by solving a specific mixed integer program and by implementing a neighborhood search starting from the current lower bound, respectively.

Table 1 summarizes the computational results. The first two columns indicate the instance with the number of nodes and the value of α. In this study, we consider 33 instances in total. For each instance, we report the CPU time in seconds, the optimality gap, the total number of cuts, the number of open facilities in the best feasible solution, and the best upper bound when the algorithm terminates in the column titled "CPU time," "Gap," "#Cut," "#Open," and "UB," respectively, under the name of the corresponding algorithm.

From Table 1, we first observe that the proposed cutting plane algorithm significantly outperforms the SnC algorithm with respect to computational time, as shown in the column titled "CPU time." For example, for the instance with 50 nodes and $\alpha = 1.00$, the proposed cutting plane algorithm solves this instance in 1.6 seconds and is around 31 times faster than the SnC algorithm, which takes 51.4 seconds. For other instances with 50 or 75 nodes, the proposed cutting plane algorithm runs about 14-92 times faster than the SnC algorithm does. Particularly for the instances with 100 nodes, the SnC algorithm fails to solve 10 out of 11 instances to an optimality gap of 0.5% within 3600 seconds, while the proposed cutting plane algorithm solves all the instances within 400 seconds.

It is further observed that the efficiency of the SnC algorithm is quite sensitive to the value of α, i.e., the CPU time increases acutely with the value of α. For example, for the instances with 75 nodes, as the value of α increases from 1.00 to 1.50, the CPU time of the SnC algorithm increases from 224.1 seconds to 1033.9 seconds. This observation is consistent with that in Aboolian et al. (2013). In contrast, for the proposed cutting plane algorithm, the CPU time remains stable with the increase in α. For example, as the value of α increases from 1.00 to 1.50, the CPU time for instances with 75 nodes ranges from 12.7 seconds to 28.5 seconds. Some instances with a larger value of α are solved even faster. This observation demonstrates the stable and robust performance of the proposed cutting plane algorithm.

Secondly, for both algorithms, the number of cuts added is moderate and increases modestly with the problem size, as shown in the column titled "#Cuts." For example, for the instance with 50 nodes and $\alpha = 1.00$, the cutting plane algorithm and the SnC algorithm add 179 and 23 cuts, respectively. When the problem size increases, e.g., for the instance with 75 nodes and $\alpha = 1.00$, two algorithms add 320 and 24 cuts, respectively. This indicates that, although theoretically there are an exponential number of cuts to be

Table 1 Numerical results on the stochastic RUFLP

Instance		Cutting plane algorithm					SnC algorithm				
Nodes	α	CPU time(s)	Gap (%)	#Cut	#Open	UB ($\times 10^3$)	CPU time(s)	Gap (%)	#Cut	#Open	UB ($\times 10^3$)
50	1.00	1.6	0.04	179	9	1020.6	51.4	0.38	23	9	1020.2
50	1.05	1.5	0	183	9	1021.0	50.5	0.50	23	9	1021.0
50	1.10	2.4	0	165	9	1021.9	61.6	0.46	27	9	1021.9
50	1.15	2.3	0	199	9	1022.8	74.9	0.49	31	9	1022.8
50	1.20	2.3	0	201	9	1023.6	82.3	0.47	33	9	1023.6
50	1.25	1.6	0.49	153	9	1025.0	86.6	0.47	35	9	1024.5
50	1.30	2.0	0	205	9	1025.3	100.9	0.48	39	9	1025.3
50	1.35	1.8	0.42	195	9	1026.2	111.0	0.48	42	9	1026.2
50	1.40	2.3	0.11	207	9	1027.0	135.7	0.49	49	9	1027.0
50	1.45	1.7	0.49	219	9	1027.8	160.8	0.40	56	9	1027.8
50	1.50	1.9	0	232	9	1028.5	161.9	0.49	57	9	1028.5
75	1.00	15.9	0	320	10	1148.5	224.1	0.41	24	10	1148.5
75	1.05	17.2	0	379	10	1149.5	247.6	0.42	26	10	1149.5
75	1.10	12.7	0.20	352	10	1152.8	261.0	0.47	27	10	1150.5
75	1.15	16.9	0	338	10	1151.5	318.4	0.46	32	10	1151.5
75	1.20	14.8	0	358	10	1152.5	394.8	0.49	38	10	1152.5
75	1.25	19.2	0.20	353	10	1155.8	474.1	0.44	44	10	1153.5
75	1.30	21.8	0.48	414	10	1160.0	543.7	0.43	49	10	1154.5
75	1.35	23.2	0	413	10	1155.5	590.3	0.50	52	10	1155.5
75	1.40	18.6	0	328	10	1156.5	701.7	0.49	59	10	1156.5
75	1.45	18.5	0.43	361	10	1157.5	878.8	0.50	70	10	1157.5
75	1.50	28.5	0.28	488	10	1161.8	1033.9	0.50	79	10	1158.5
100	1.00	246.7	0.38	849	13	1248.9	2708.4	0.50	67	13	1248.9
100	1.05	218.1	0.31	834	13	1253.6	3620.9	0.53	80	13	1249.8
100	1.10	262.4	0.23	931	13	1253.5	3644.7	0.63	82	13	1250.7
100	1.15	299.4	0.45	872	12	1255.1	3649.9	0.76	81	13	1251.6
100	1.20	257.7	0.45	690	13	1252.4	3650.0	0.87	82	13	1252.4
100	1.25	263.4	0.33	830	13	1253.3	3632.3	0.99	81	13	1253.3
100	1.30	281.4	0.46	846	13	1257.8	3610.0	1.11	81	13	1254.2
100	1.35	259.4	0.40	872	13	1260.1	3638.0	1.23	82	13	1255.1
100	1.40	259.1	0.45	779	13	1259.0	3611.3	1.35	81	13	1256.0
100	1.45	306.3	0.24	929	13	1256.9	3641.2	1.46	82	13	1256.9
100	1.50	398.9	0	1100	13	1257.8	3617.9	1.58	82	13	1257.8

added when applying the cutting plane algorithm, only a very small portion of them are actually needed to obtain an optimal solution.

It is interesting that the proposed cutting plane algorithm adds many more cuts but solves faster than the SnC algorithm. This observation can be justified as follows. Note from the column titled “UB” that the best upper bound of two algorithms are close to each other, even for the instances with 100 nodes, where the SnC algorithm fails to reduce the optimality gap to and below 0.5%. One can infer that the SnC algorithm generates a good upper bound quickly, but updates the lower bound slowly, making the optimality gap converge inefficiently. In contrast, the proposed cutting plane algorithm adds more cuts into the relaxed master problem, which gives a good lower bound because the lower bound is obtained by solving the relaxed master problem at the current iteration.

5.2. Numerical Results on the Robust RUFLP

For the robust RUFLP, we consider the model with the marginal moment and the partial cross moment of disruption probabilities, which is referred to as the cross moment model. The cross moment model is solved by the cutting plane algorithm in Section 4 to numerically validate the effectiveness and efficiency of the proposed algorithm. The proposed cross moment model is compared with the marginal moment model in Lu et al. (2015), namely the robust model making reliable location decisions based on marginal disruption probabilities, which is the only distributionally robust model on the RUFLP to our knowledge. The comparison between the cross moment model and the marginal moment model demonstrates the value of cross moment information in making location decisions. We also compare the cross moment model with the stochastic model, which assumes independent disruptions among facilities. The comparison between the cross moment model and the stochastic model illustrates the value of correlation information in decision making.

We consider instances with 20, 50, 75, and 100 nodes. Following Lu et al. (2015), the marginal disruption probability is given by $1-q_j=\beta e^{-D_j/\theta}$ for all $j\in J$. Specifically, the parameter D_j denotes the great circle distance in miles between node j and New Orleans, the center of Hurricane Katrina. The parameter β measures the source disaster occurrence probability, i.e., the probability that a disastrous event happens at the source New Orleans. The parameter θ measures the disruption propagation effect, where the larger θ indicates that the disaster affects a larger area. In this numerical study, β takes the value of 0.1 or 0.2, and θ takes the value of 200, 400, or 800, similar to the settings in Lu et al. (2015). For the partial cross moment, it is assumed that if the distance between any two facilities is no less than 2500 kilometers, then the disruptions of these two facilities are independent, i.e., the cross disruption probability of these two facilities, say facilities j and k, is $1-(q_j+q_k-q_{jk})=(1-q_j)(1-q_k)$. For this data set, the number of facility pairs with cross disruption probability is 7, 24, 42, and 115 for the instances with 20, 50, 75, and 100 nodes, respectively. Taking the marginal probability into account, the value of n, i.e., the number of pieces of information used to define the ambiguity set P in (1), equals 27, 74, 117, and 225, respectively, for the corresponding instances. To illustrate the scalability of the proposed algorithm for larger n values, we conduct additional numerical studies and the results are presented in Appendix B.2. The demand, the fixed facility setup cost, the unit transportation cost, and the unit emergency cost are generated in a similar way as in Lu et al. (2015) based on the same data set.

The cross moment model is solved by Algorithm CP. In each CP iteration, the relaxed master problem $\mathcal{RP}$ is solved by CPLEX under standard setting for mixed integer linear programs. The separation problem Z_{sep} is handled by the column generation approach in Section 4.2, where the reduced cost RC is obtained by solving a revised formulation of model (10) in accordance with the partial cross moment information. A warm start is performed to solve the separation problem in the CP iterations. Note that each column in the separation problem $Z_{sep}(\boldsymbol{x})$ corresponds to a specific disruption scenario $\boldsymbol{\xi}$. Some trivial columns are always considered in the master problem, including the columns with all facilities online/offline, the columns with only one facility online/offline, the columns with a pair of facilities corresponding to a cross moment online/offline, etc. Besides, the columns generated in the previous CP iteration are randomly preserved and inherited to the next CP iteration for a better column generation start. Furthermore, Algorithm CP considers valid CP cuts (5) characterized by optimal solutions $\boldsymbol{p}^*$ and $\boldsymbol{x}^*$ to Z_{sep} and $\mathcal{RP}$, respectively. Notice that it is also possible to get a valid CP cut even if $\boldsymbol{p}^*$ and $\boldsymbol{x}^*$ are not optimal. Thus, to speed up the convergence, in certain randomly selected CP iterations, sub-optimal solutions of the separation problem Z_{sep} and the relaxed master problem $\mathcal{RP}$ are also used to generate the CP cuts.

As for the two benchmark problems, the marginal moment model is also solved following the framework of Algorithm CP, where the evaluation of the worst-case transportation cost is based on the closed-form worst-case distribution in Lu et al. (2015). The stochastic model is solved in the same way as in Section 5.1. For all the three models, Algorithm CP terminates when the optimality gap, i.e., the relative gap between the best upper bound (the objective value corresponding to the best feasible solution) and the best lower bound (the optimal objective value of $\mathcal{RP}$ in the current CP iteration), is no more than 0.5%. The algorithm then outputs the best feasible solution, which is regarded as an optimal solution in the following discussion. Indeed, we observe from additional numerical experiments (which are not reported for conciseness) that, for some instances if we set a lower stopping gap, for example, 0.01% or even 0% (i.e., the algorithm terminates only if the best lower bound equals the best upper bound, or equivalently, $\eta^* \geq Z_{sep}(\boldsymbol{x}^*)$ as in Step 3 of Algorithm CP), the algorithm may output the same solution as the one with the stopping gap being 0.5%. This implies that the algorithm actually finds an optimal solution before the optimality gap reaches 0.5% and takes some more time to reduce the optimality gap (or prove the optimality) by improving the best lower bound.

The numerical results are shown in Table 2. The first three columns represent the instances defined by the combination of (Nodes, β, θ). Thus, 24 instances are considered in total. For each instance, the cross moment model in this paper, the marginal moment model in Lu et al. (2015), and the stochastic model with independent disruptions are solved, respectively. For each model, the solution time, the number of cutting plane cuts, the number of open facilities in the best feasible solution, and the objective value of the best feasible solution (i.e., the best upper bound) are recorded accordingly in the columns titled "CPU time," "#Cut," "#Open," and "Obj." Particularly for the cross moment model, there are three columns under the name "CPU time": i) the column titled "Total" records the total CPU time to solve the instance, ii) the column titled "Z_{sep}" records the cumulative CPU time for solving Z_{sep}, i.e., identifying the valid cut (5) in the cutting plane process, and, iii) the column titled "RC" records the cumulative CPU time for solving RC, i.e., identifying the valid column for Z_{sep} in the column generation process.

To evaluate the value of cross moment information in making reliable location decisions, we further calculate the regret and the relative regret of ignoring cross moment information in the columns titled "Regret" and "Regret%" under the name "Marginal moment model," respectively. Similarly, we calculate the regret and the relative regret of ignoring the correlation information in the columns titled "Regret" and "Regret%" under the name "Stochastic model," respectively. The regret of the marginal moment model (or stochastic model) denotes the cost increase of ignoring the cross moment information (or the correlation information) and implementing the location decisions by the marginal moment model (or the stochastic model). To be specific, let $\boldsymbol{x}^\dagger$ be the optimal location decision by the marginal moment model or the stochastic model, P_{CM} be the set of distributions in the cross moment model, and $\mathrm{Obj}_{\mathrm{CM}}$ be the optimal objective value of the cross moment model. Then Regret and Regret% can be calculated as follows:

$$\begin{aligned}\text{Regret} \quad &= \boldsymbol{f}^\top \boldsymbol{x}^\dagger + \max\nolimits_{\boldsymbol{p}\in P_{\mathrm{CM}}} \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^\dagger, \tilde{\boldsymbol{\xi}})] - \mathrm{Obj}_{\mathrm{CM}} \\ \text{Regret}\% &= 100 \cdot \text{Regret}/\mathrm{Obj}_{\mathrm{CM}}\end{aligned}$$

From Table 2 we have several observations:

First, the proposed cutting plane algorithm can solve the cross moment model within reasonable CPU time with a moderate number of cutting plane cuts added (cf. column “Total” under “CPU time” and column “#Cut” of “Cross moment model”). Instances with 20 or 50 nodes can be solved efficiently within 2 seconds or 1 minute with less than 25 or 120 cutting plane cuts added, respectively, while those with 75 and 100 nodes can be solved within 2 and 6 hours with less than 1000 cuts added, respectively. Considering the frequency of making location decisions, the computational effort required by the proposed model and algorithm is satisfactory.

For most instances, the solving time of the separation problem Z_{sep} takes up more than half of the CPU time and this percentage goes to more than 80% for the instances with 100 nodes (cf. column “Z_{sep}” under “CPU time” of “Cross moment model”). When solving Z_{sep} by the column generation approach, the computation of the reduced cost RC takes a significant amount of time for most instances (cf. column “RC” under “CPU time” of “Cross moment model”). The long solving time of the separation problem explains why the CPU time of the cross moment model is much longer than that of the marginal moment model or the stochastic model (cf. column “CPU time” of “Marginal moment model” or “Stochastic model”). For the marginal moment model, we use the closed-form worst-case distribution proven by Lu et al. (2015), while for the stochastic model, the facility disruptions are assumed to be independent. In both cases, there is no need to solve the separation problem by the column generation approach, and hence the CPU time is much shorter.

Table 2 Numerical results on the robust RUFLP

Instance			Cross moment model						Marginal moment model						Stochastic model					
Nodes	β	θ	CPU time(s) Total	CPU time(s) Z_{sep}	CPU time(s) RC	#Cut	#Open	Obj ($\times 10^5$)	CPU time(s)	#Cut	#Open	Obj ($\times 10^5$)	Regret ($\times 10^5$)	Regret% (%)	CPU time(s)	#Cut	#Open	Obj ($\times 10^5$)	Regret ($\times 10^5$)	Regret% (%)
20	0.1	200	0.6	0.4	0.3	10	4	5.03	0.4	9	4	5.03	0	0	0.1	9	4	4.98	0	0
		400	0.8	0.4	0.3	17	4	6.11	0.2	14	4	6.11	0	0	0.1	9	4	5.06	0	0
		800	1.4	1.0	0.9	17	5	7.67	0.3	15	4	9.58	1.91	24.88	0.2	11	4	5.22	2.89	37.66
	0.2	200	0.4	0.2	0.2	10	4	5.10	0.1	9	4	5.10	0	0	0.2	9	4	5.02	0	0
		400	1.0	0.5	0.4	22	4	6.79	0.2	11	4	6.79	0	0	0.1	10	4	5.15	0.45	6.65
		800	1.2	0.7	0.6	21	5	8.44	0.3	9	4	12.72	4.28	50.64	0.2	14	4	5.48	7.72	91.38
50	0.1	200	7.7	3.7	0.8	39	6	6.59	4.4	35	6	6.60	0.01	0.21	3.5	31	6	6.57	0	0
		400	21.6	13.6	9.8	71	7	7.10	2.9	37	6	7.15	0.05	0.73	4.2	38	6	6.62	0.02	0.30
		800	25.8	18.0	14.5	78	7	7.87	1.6	30	6	10.45	2.58	32.79	4.8	38	6	6.75	3.02	38.36
	0.2	200	8.5	4.2	1.5	41	6	6.61	2.9	37	6	6.63	0.01	0.18	3.3	34	6	6.58	0	0
		400	17.8	10.9	8.0	60	7	7.36	4.6	47	6	7.68	0.32	4.39	5.4	42	6	6.69	0.33	4.44
		800	38.7	24.8	19.8	117	6	9.04	1.6	29	6	14.04	5.01	55.39	9.5	56	6	6.96	6.17	68.31
75	0.1	200	48.5	37.3	2.1	50	7	7.52	6.8	46	6	7.52	0	0.05	10.5	57	7	7.50	0	0
		400	168.9	136.7	76.8	87	7	7.93	11.9	60	6	8.14	0.21	2.64	5.7	52	7	7.57	0.22	2.72
		800	339.6	223.9	130.7	164	7	8.81	6.6	45	6	11.92	3.11	35.35	16.6	66	6	7.71	3.65	41.47
	0.2	200	66.4	48.5	9.4	73	7	7.57	14.3	49	7	7.57	0	0	8.3	56	6	7.53	0.01	0.09
		400	183.2	135.4	80.4	109	7	8.25	10.7	60	6	8.81	0.56	6.75	8.3	60	7	7.67	0.57	6.92
		800	4066.6	1152.4	747.9	734	6	10.28	4.8	30	6	16.10	5.82	56.61	22.9	90	7	7.96	7.20	70.03
100	0.1	200	1055.8	1000.3	292.1	64	7	8.08	20.7	58	7	8.08	0.01	0.09	14.6	63	7	8.06	0.01	0.09
		400	10936.7	10772.8	6522.2	128	7	8.39	40.4	85	7	8.77	0.38	4.47	28.7	83	7	8.15	0.39	4.70
		800	5034.9	4753.1	3630.1	239	7	9.25	27.1	62	7	12.86	3.61	39.05	51.0	109	7	8.33	4.30	46.44
	0.2	200	892.4	829.6	21.7	88	7	8.13	32.6	66	7	8.13	0	0	62.4	71	7	8.09	0.01	0.09
		400	6630.9	6455.0	4397.8	143	7	8.69	61.3	99	7	9.45	0.75	8.68	61.9	88	7	8.25	0.83	9.50
		800	20659.3	17057.7	12782.4	815	7	10.61	59.1	80	8	17.29	0.39	3.63	379.7	172	7	8.57	8.40	79.17

Second, the number of open facilities in the solution roughly increases with the number of nodes, the value of β (the source disaster probability), and the value of θ (the disruption propagation effect), for all the models (cf. column "#Open"). As a matter of fact, we observe from the computing process (which is not reported for conciseness) that the location decision and the number of open facilities fluctuate significantly as the algorithm converges. This implies that the robust models (both the cross moment model and the marginal moment model) along with the proposed algorithm make a complex tradeoff between opening more facilities to serve demand nodes with lower transportation cost and opening fewer facilities for a smaller risk of correlated facility disruptions.

We also notice that the cross moment solution opens more facilities than the marginal moment solution for most instances, especially for those with a larger value of θ. For example, for the instance (20,0.1/0.2,800), the cross moment model opens 5 facilities while the marginal moment model opens 4 facilities. This observation may be because the cross moment model makes the location decision based on the cross moment information, i.e., the independence of certain facility pairs, hence it is natural for the cross moment model to "aggressively" open more facilities that are independent of one another. In contrast, the marginal moment model makes the location decision ignoring the independent information, thus it tends to be more conservative, opening fewer facilities. The only exception is the instance (100,0.2,800), where the cross moment solution opens 7 facilities while the marginal moment solution opens 8 facilities. This contrasting observation can be explained by the incidental fact that the marginal moment solution happens to open two facilities that are independent of each other.

Recall that each combination of (Nodes, β, θ) only corresponds to one instance whose parameters are calculated based on the 1990 census data. To eliminate the incidental issues, we conduct further numerical experiments based on randomly generated instances with 20 nodes. The details of the numerical studies are presented later in this section and also in Appendix B.3. It is observed in Appendix B.3 that the cross moment model opens more facilities on average than the marginal moment model. On the other hand, the stochastic model opens the least number of facilities on average among the three models for most randomly generated instances. This observation can be explained by the fact that the stochastic model assumes independent disruptions for all the facility pairs, thus it tends to make the most "optimistic" location decision with the least number of open facilities.

Note that the numerical studies in Appendix B.3 also report how the number of open facilities for all the three models change with β and θ. To sum up the comparison of the location decisions by the three models, we could infer that, with more facility pairs having independent disruptions (or in other words, with more information on correlation specifying independent disruptions), the number of open facilities of the cross moment model is likely to increase first (deviating from the marginal moment decision) and then decrease (tending to the stochastic decision). Interestingly, this insight is partially verified by the numerical results in Appendix B.2.

Third, the regret of ignoring cross moment information fluctuates greatly for different instances and it can be substantial for some instances (cf. the columns "Regret" and "Regret%" of "Marginal moment model"). As can be observed from the columns under "Marginal moment model," the value of Regret% generally increases with the number of nodes and the values of β and θ, ranging from 0% to 56.61%. For most instances with $\theta = 800$, the value of Regret% is above 24%, implying that the cost increases by at least 24% if one ignores the cross moment information and implements the location decision by the marginal moment model in these instances. The only exception is the instance (100,0.2,800), where the value of Regret% suddenly decreases to 3.63%, comparing to 8.68% in instance (100,0.2,400) and 39.05% in instance (100,0.1,800). This is because the marginal moment model happens to choose some facilities that are independent of one another. The worst-case expected transportation cost of the marginal moment decision is consequently close to that of the cross moment decision, which is determined with the independent cross moment information. As a result, the value of Regret% is small.

We also observe that, the regret of ignoring correlation fluctuates greatly for different instances and is tremendous for some instances (cf. columns "Regret" and "Regret%" of "Stochastic model"). Note that Lu et al. (2015) compare the stochastic model with the marginal moment model to reveal the regret of ignoring correlation, while in this work, we compare the stochastic model with the cross moment model. As can be observed from the columns under "Stochastic model," the value of Regret% generally increases with the number of nodes and the values of β and θ, ranging from 0% to 91.38%. For most instances with $\theta = 800$, the value of Regret% is above 37%, implying that the cost increases by at least 37% if one ignores the correlation and implements the location decision by the stochastic model assuming independent disruptions in these instances.

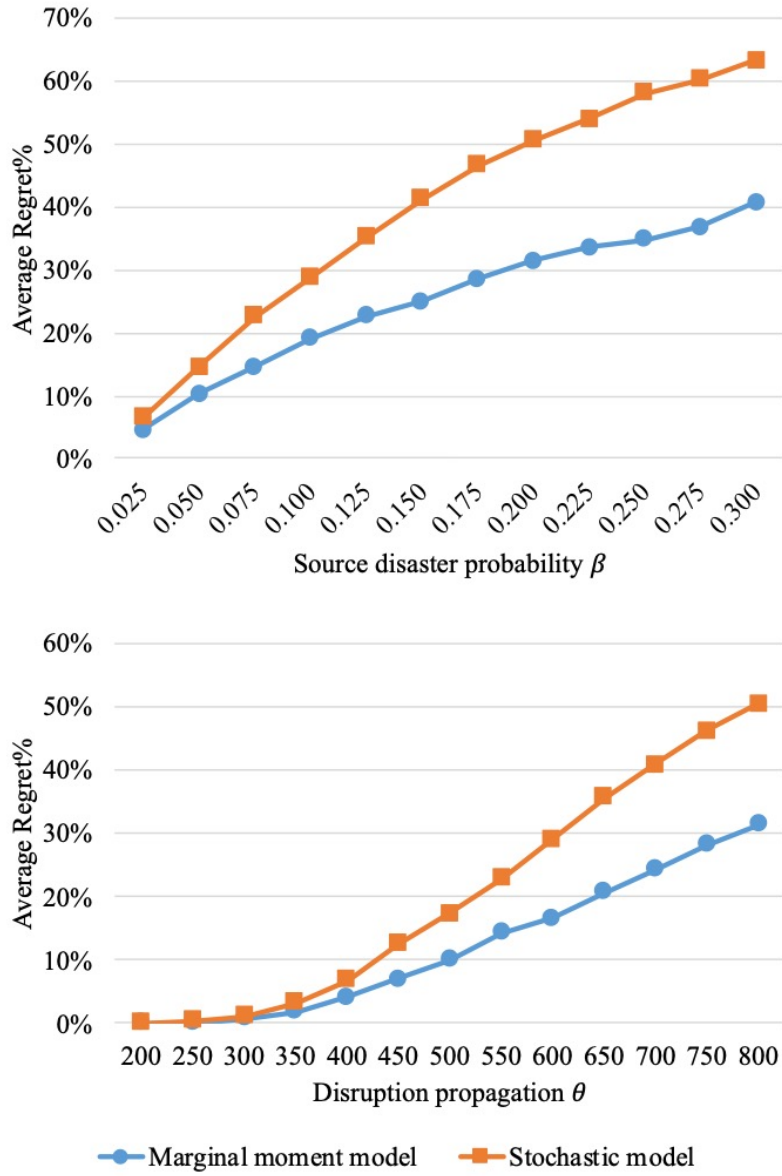

Figure 1 **Impact of β and θ on average Regret%**

To clearly see how the values of Regret% of the marginal moment model and the stochastic model change with β and θ, we conduct further experiments based on randomly generated instances with 20 nodes. In these instances, we randomly generate the location, demand, and fixed facility setup cost of each node, based on which the unit transportation cost and marginal/cross disruption probability are calculated accordingly. To see how Regret% changes with the value of β, we fix θ to be 800 and set the value of β from 0.025 to 0.300 with a step length of 0.025. To see how Regret% changes with the value of θ, we fixed β to be 0.2 and set the value of θ from 200 to 800 with a step length of

50. For each value of β or θ, 100 scenarios are randomly generated, and the average value of Regret% is calculated and presented in Figure 1. More details of the numerical studies based on randomly generated instances can be found in Appendix B.3. Figure 1 shows that the regret of the stochastic model is generally larger than that of the marginal moment model, indicating that the regret of ignoring correlation is generally higher than that of ignoring cross moment information. As the value of β increases, the average Regret% of the marginal moment model increases significantly from around 5% to around 40%, while the increase of the average Regret% of the stochastic model is substantially larger, ranging from around 6% to around 63%. A similar observation can be made when we increase the value of θ. With the increasing value of θ, the average Regret% of the marginal moment model grows from around 0% to around 31%, while that of the stochastic model grows from around 0% to around 50%.

5.3. Data-Driven Simulation

To compare the aforementioned two robust models in more practical disaster scenarios, we conduct the following data-driven simulation based on the historical typhoon data in China.

The data is collected by the Shanghai Typhoon Institute and the Shenzhen Meteorology Bureau of the China Meteorology Administration[1]. The data set covers all typhoons landed in China from 1949 to 2019 and records typhoon information such as serial number, typhoon name, landing location, landing time, landing intensity, and typhoon track.

In this simulation, we extract the data from 2007 to 2019, which includes 100 typhoons in total. Each typhoon is regarded as one scenario and the track of this typhoon indicates the affected locations in this scenario. To have a discrete location set, we focus on the southeast coastal area of China, which is the area most frequently affected by typhoons in China. This area is divided into 16 locations according to the provincial administrative division. They are East Guangdong, West Guangdong, South Fujian, North Fujian, South Hainan, North Hainan, South Taiwan, North Taiwan, South Zhejiang, North Zhejiang, Guangxi, Jiangsu, Jiangxi, Shandong, Anhui, and Shanghai, among which 5 provinces are divided into two parts respectively due to the high frequency of typhoon hits. For each location, the demand is set as the population and the fixed location cost is estimated according to

[1] Data source: Shanghai Typhoon Institute (http://tcdata.typhoon.org.cn) and Shenzhen Meteorology Bureau (http://tf.121.com.cn)

the housing price. The unit transportation cost between any two locations is proportional to the distance between them.

Table 3 Numerical results on simulation based on historical typhoon data

		Data from 2008 to 2017		Data from 2007 to 2016	
		CM	MM	CM	MM
CPU time (s)		31.99	0.12	50.02	0.04
#Cut		133	14	84	8
#Open		4	4	4	4
Open locations		Jiangsu, Guangxi, Jiangxi, Shandong	Shanghai, Guangxi, Jiangxi, Shandong	Anhui, Guangxi, Jiangxi, Shandong	Shanghai, Guangxi, Jiangxi, Shandong
Cost ($\times 10^6$)	Location	7.03	14.53	6.25	14.53
	Transportation	13.69	14.23	13.92	14.23
	Total (i.e., Obj)	20.71	28.75	20.17	28.75
Scenario 2017	Total cost ($\times 10^6$)	-	-	11.38	19.79
	Total cost%		-		73.90%
Scenario 2018	Total cost ($\times 10^6$)	13.10	20.53	12.01	20.53
	Total cost%		56.72%		71.03%
Scenario 2019	Total cost ($\times 10^6$)	12.08	19.42	10.99	19.42
	Total cost%		60.72%		76.74%

The results are presented in Table 3. We first use the data from 2008 to 2017, which covers 75 typhoon scenarios, to generate the marginal/cross disruption probabilities (cf. columns under "Data from 2008 to 2017"). To have more observations, we also use the data from 2007 to 2016, which covers 74 typhoon scenarios, to generate these parameters (cf. columns under "Data from 2007 to 2016"). The cross moment model (cf. "CM") and the marginal moment model (cf. "MM") are compared. For each model, the solution time, the number of CP cuts, the number of open locations in the optimal solution, and the name of the open locations are recorded accordingly in the rows titled with "CPU time," "#Cut," "#Open," and "Open locations." The fixed location cost, the worst-case expected transportation cost, and the total cost (i.e., the objective value) of the optimal solution are presented in the rows titled with "Cost" followed by "Location," "Transportation," and "Total," respectively. The data in 2018-2019 or 2017-2019 is used as testing scenarios to evaluate the performance of the optimal solutions by two robust models. Specifically, we calculate the total cost of the corresponding solutions under the typhoon scenario 2017, 2018, and 2019, respectively (cf. row "Total cost" of "Scenario 2017," "Scenario 2018," and "Scenario 2019," respectively). The relative total cost increase of the MM model comparing to the CM model is also presented in the row titled "Total cost%" of the corresponding scenario.

If we use the data from 2008 to 2017 to generate the disruption probabilities (cf. the columns under "Data from 2008 to 2017"), it can be observed that both models can be solved in short CPU time with a moderate number of CP cuts generated. Both models open four locations, but the CM and the MM models open Jiangsu and Shanghai, respectively, in addition to the three locations in both solutions (i.e., Guangxi, Jiangxi, and Shandong). Interestingly, the MM model chooses Shanghai as one of the open locations, leading to a very high location cost (cf. row "Location" of "Cost"). This is because the MM model is more conservative in the sense that it optimizes the worst-case expected transportation cost with the marginal moment information, and the corresponding worst-case expected transportation cost is higher than that of the CM model. Thus, the MM model tends to open some locations (i.e., Shanghai in this case) with high fixed location cost but low unit transportation cost, so as to decrease the worst-case expected transportation cost as far as possible and achieve the balance between the location cost and the transportation cost in the objective function.

Under the testing scenarios 2018 and 2019 (cf. rows "Scenario 2018" and "Scenario 2019"), the total costs of both robust models are much lower than the worst-case counterparts. For example, the total costs of the CM and the MM models under scenario 2018 are 13.10 and 20.53 million, respectively, while the worst-case counterparts are 20.71 and 28.75 million, respectively. However, the MM model bears a total cost that is 56.72% and 60.72% higher than the CM model under the two testing scenarios, respectively.

If we use the data from 2007 to 2016 to generate the disruption probabilities (cf. the columns under "Data from 2007 to 2016"), it can be observed that the optimal solution of the CM model is different with the open location Jiangsu changed to Anhui, while that of the MM model remains the same (cf. row "Open location"). In a certain sense, this indicates that the CM model is more sensitive to the input data and can better utilize the information, especially the correlation information in this simulation. As for the performance under the testing scenarios (cf. rows "Scenario 2017," "Scenario 2018," and “Scenario 2019”), the MM model still bears a higher total cost than the CM model. Moreover, the relative total cost increases are higher than those under “Data from 2008 to 2017,” e.g., raising from 56.72% to 71.03% under scenario 2018.

6. Conclusions

In this paper, we study the reliable supply chain network design problem that generalizes the classical uncapacitated facility location problem by considering uncertain facility disruptions. This problem is formulated as a distributionally robust model in a general form, which extends several existing models, including the stochastic model with given disruption distribution, the robust model with marginal disruption probability, and the robust model with cross disruption probability. To solve this model, an efficient cutting plane algorithm is proposed, where the separation problem is solved respectively by a polynomial-time algorithm in the stochastic case and by a column generation approach in the robust case. Extensive numerical studies shows that the proposed cutting plane algorithm not only outperforms the best algorithm in the literature for the stochastic model with independent disruptions, but also efficiently solves the robust model with correlated disruptions.

This paper can be extended to the case with uncertain demand when the demand is independent of the facility disruption, or the dependency between demand and disruption is scenario-based or expressed by linear functions. However, the proposed approach can not be easily applied to handle more general dependency between demand and facility disruption, which is one limitation of this work. Another limitation is that the facilities are all uncapacitated, and the inventory positioning decisions are not considered in this work. In our future work, this UFLP-based problem will be extended to incorporate the inventory positioning and allocation decisions. Moreover, if this problem is considered in a general supply chain with customer sourcing decisions, it might be interesting to study how the customer sourcing decisions are affected by facility disruption or supplier reliability.

Acknowledgments

We are very grateful to the Associate Editor and the two anonymous reviewers for their very constructive and insightful comments.

Appendix A: Proofs of Theorems

A.1. Proof of Theorem 1

Proof. Notice that the optimal solution $\boldsymbol{p}^*$ to $Z_{sep}(\boldsymbol{x}^*)$ must be attained at an extreme point of P, thus the number of iterations in Algorithm CP is bounded above by $|\{\boldsymbol{y} \in \{0,1\}^{|J|}\}|$, i.e., $2^{|J|}$, times the quantity of extreme points of P, which is a finite number. Therefore, Algorithm CP must terminate within finite iterations. To show that Algorithm CP terminates with the optimal solution to model (4), we firstly develop an algorithm for model (4) based on the Benders decomposition, and then show that the Benders decomposition algorithm can solve model (4) to optimality and is equivalent to Algorithm CP.

Given the solution $(\boldsymbol{x}, \eta)$ and the scenario $\boldsymbol{\xi}$, the subproblem $Q(\boldsymbol{x}, \boldsymbol{\xi})$ can be written as

$$\begin{aligned} Q(\boldsymbol{x},\boldsymbol{\xi}) &= \min_{\boldsymbol{v}} \left\{ \sum_{i\in I}\sum_{j\in J\cup\{0\}} d_i c_{ij} v_{ij} \;\middle|\; \begin{array}{ll} \sum_{j\in J\cup\{0\}} v_{ij}=1, & \forall i\in I, \\ -v_{ij}\geq -x_j\xi_j, & \forall i\in I, j\in J\cup\{0\}, \\ v_{ij}\geq 0, & \forall i\in I, j\in J\cup\{0\} \end{array} \right\} \\ &= \max_{\boldsymbol{\sigma},\boldsymbol{\omega}} \left\{ \sum_{i\in I}\sigma_i - \sum_{i\in I}\sum_{j\in J\cup\{0\}} x_j\xi_j\omega_{ij} \;\middle|\; \begin{array}{l} \sigma_i-\omega_{ij}\leq d_i c_{ij},\ \forall i\in I, j\in J\cup\{0\}, \\ \sigma_i\in\mathbb{R}, \omega_{ij}\geq 0,\ \forall i\in I, j\in J\cup\{0\} \end{array} \right\} \end{aligned}$$

where $\boldsymbol{v}$ is the vector of the primal variables v_{ij} for all $i\in I$ and $j\in J\cup\{0\}$, and $\boldsymbol{\sigma}$ and $\boldsymbol{\omega}$ are the vectors of the dual variables σ_i (for all $i\in I$) and ω_{ij} (for all $i\in I$ and $j\in J\cup\{0\}$), respectively. It is obvious that $Q(\boldsymbol{x},\boldsymbol{\xi})$ is always feasible and has a finite optimal objective value, because $v_{i0}=1$ and $v_{ij}=0$ for all $i\in I$ and $j\in J$ is a trivial feasible solution to $Q(\boldsymbol{x},\boldsymbol{\xi})$, and the corresponding objective value $\sum_{i\in I} d_i c_{i0}$ is an upper bound of $Q(\boldsymbol{x},\boldsymbol{\xi})$. Therefore, the dual problem of $Q(\boldsymbol{x},\boldsymbol{\xi})$ is always feasible with a bounded optimal value. Thus, an optimal solution to the dual problem of $Q(\boldsymbol{x},\boldsymbol{\xi})$ must be attained at an extreme point of its feasible region. Let Φ^{EP} denote the set of all the extreme points for the dual problem of $Q(\boldsymbol{x},\boldsymbol{\xi})$. Model (4) can be put in the following form:

$$\begin{aligned} \mathcal{P}: \min_{\substack{\boldsymbol{x}\in\{0,1\}^{|J|},\\ \eta\in\mathbb{R}}} \ & \sum_{j\in J} f_j x_j + \eta \\ \text{s.t.} \quad & \eta \geq \sum_{\boldsymbol{\xi}\in\Xi|p_{\boldsymbol{\xi}}>0} p_{\boldsymbol{\xi}} \left(\sum_{i\in I}\sigma_{\boldsymbol{\xi},i} - \sum_{i\in I}\sum_{j\in J\cup\{0\}} x_j\xi_j\omega_{\boldsymbol{\xi},ij} \right),\ \forall \boldsymbol{p}\in P, (\boldsymbol{\sigma}_{\boldsymbol{\xi}},\boldsymbol{\omega}_{\boldsymbol{\xi}})_{\forall\boldsymbol{\xi}\in\Xi}\in\Phi^{EP}, \end{aligned} \tag{11}$$

where $(\boldsymbol{\sigma}_{\boldsymbol{\xi}},\boldsymbol{\omega}_{\boldsymbol{\xi}})$ denotes the dual variables to $Q(\boldsymbol{x},\boldsymbol{\xi})$ with the subscript $\boldsymbol{\xi}$ expressed to avoid confusion. The entries of $\boldsymbol{\sigma}_{\boldsymbol{\xi}}$ and $\boldsymbol{\omega}_{\boldsymbol{\xi}}$ are denoted as $\sigma_{\boldsymbol{\xi},i}$ and $\omega_{\boldsymbol{\xi},ij}$, respectively.

We next develop an algorithm for model (11) based on the Benders decomposition. In each typical iteration of the Benders decomposition algorithm, we consider a relaxed master problem, which has the same objective as model (11), but involves only a subset of the constraints. The relaxed master problem is firstly solved and the corresponding optimal solution is denoted as $(\boldsymbol{x}^*,\eta^*)$. Then the separation problem $Z_{sep}(\boldsymbol{x}^*) = \max_{\boldsymbol{p}\in P}\mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*,\tilde{\boldsymbol{\xi}})]$ as defined in Algorithm CP is solved to check whether the current solution $(\boldsymbol{x}^*,\eta^*)$ is also a feasible solution to the original problem $\mathcal{P}$: (i) if $\eta^*\geq Z_{sep}(\boldsymbol{x}^*)$, $(\boldsymbol{x}^*,\eta^*)$ is feasible and also optimal to the original problem $\mathcal{P}$, (ii) otherwise, i.e., $\eta^* < Z_{sep}(\boldsymbol{x}^*)$, we have to identify a valid cut and add it to the relaxed master problem. In view of the constraints in model (11), the valid cut can be constructed as follows: Denote $\boldsymbol{p}^*$ as the optimal solution to $Z_{sep}(\boldsymbol{x}^*)$. For every $\boldsymbol{\xi}\in\Xi$ such that $p^*_{\boldsymbol{\xi}}>0$, solve the dual problem of $Q(\boldsymbol{x}^*,\boldsymbol{\xi})$ and obtain the dual optimal solution $(\boldsymbol{\sigma}^*_{\boldsymbol{\xi}},\boldsymbol{\omega}^*_{\boldsymbol{\xi}})$. Then, the following constraint is the constraint of model (11) violated by the current solution $(\boldsymbol{x}^*,\eta^*)$, i.e., the valid cut:

$$\eta \geq \sum_{\boldsymbol{\xi}\in\Xi|p^*_{\boldsymbol{\xi}}>0} p^*_{\boldsymbol{\xi}} \left(\sum_{i\in I}\sigma^*_{\boldsymbol{\xi},i} - \sum_{i\in I}\sum_{j\in J\cup\{0\}} x_j\xi_j\omega^*_{\boldsymbol{\xi},ij} \right). \tag{12}$$

This valid cut is then added to the relaxed master problem, which leads to a new iteration.

The Benders decomposition algorithm described above can solve model (11) to optimality. This is because: On one hand, as model (11) is a minimization problem, the optimal objective value of the relaxed master problem is smaller than or equal to that of the original problem $\mathcal{P}$. On the other hand, when the Benders decomposition algorithm terminates with the solution $(\boldsymbol{x}^*, \eta^*)$, we have $\eta^* \geq Z_{sep}(\boldsymbol{x}^*)$, i.e., $\eta^* \geq \mathbb{E}_{\boldsymbol{p}}[Q(\boldsymbol{x}^*, \tilde{\boldsymbol{\xi}})]$ for all $\boldsymbol{p} \in P$. Thus, $(\boldsymbol{x}^*, \eta^*)$ is feasible to the original problem $\mathcal{P}$, implying that the optimal objective value of the relaxed master problem is greater than or equal to that of the original problem $\mathcal{P}$. Therefore, when the Benders decomposition algorithm terminates, the optimal objective value of the relaxed master problem is equal to that of the original problem $\mathcal{P}$, and we have that the Benders decomposition algorithm can solve model (11) to optimality.

We next show that the Benders decomposition algorithm is equivalent to Algorithm CP. Note that the separation problems $Z_{sep}(\boldsymbol{x}^*)$ are the same for two algorithms, thus it remains to show that the valid cuts added in each typical iteration of these two algorithms, i.e., (5) and (12), are the same.

Notice that constraint (12) can be simplified because a dual optimal solution $(\boldsymbol{\sigma}^*_{\boldsymbol{\xi}}, \boldsymbol{\omega}^*_{\boldsymbol{\xi}})$ to $Q(\boldsymbol{x}^*, \boldsymbol{\xi})$ can be written explicitly as:

$$\begin{cases} \sigma^*_{\boldsymbol{\xi},i} &= d_i \min_{j \in J\cup\{0\} | x^*_j \xi_j = 1} c_{ij}, & \forall i \in I, \\ \omega^*_{\boldsymbol{\xi},ij} &= 0, & \forall i \in I, j \in J \cup \{0\} | x^*_j \xi_j = 1, \\ \omega^*_{\boldsymbol{\xi},ij} &= \max\{0, \sigma^*_{\boldsymbol{\xi},i} - d_i c_{ij}\} & \\ &= d_i \max\{0, \min_{l \in J\cup\{0\} | x^*_l \xi_l = 1} c_{il} - c_{ij}\}, & \forall i \in I, j \in J \cup \{0\} | x^*_j \xi_j = 0, \end{cases}$$

which are compactly written as

$$\begin{cases} \sigma^*_{\boldsymbol{\xi},i} &= d_i \min_{j \in J\cup\{0\} | x^*_j \xi_j = 1} c_{ij}, & \forall i \in I, \\ \omega^*_{\boldsymbol{\xi},ij} &= d_i \max\{0, \min_{l \in J\cup\{0\} | x^*_l \xi_l = 1} c_{il} - c_{ij}\}, & \forall i \in I, j \in J \cup \{0\}. \end{cases}$$

Thus, constraint (12) can be reformulated as:

$$\begin{aligned} & \eta \geq \sum_{\boldsymbol{\xi} \in \Xi | p^*_{\boldsymbol{\xi}} > 0} p^*_{\boldsymbol{\xi}} \left(\sum_{i \in I} \sigma^*_{\boldsymbol{\xi},i} - \sum_{i \in I} \sum_{j \in J} x_j \xi_j \omega^*_{\boldsymbol{\xi},ij} \right) \\ \Leftrightarrow\ & \eta \geq \sum_{\boldsymbol{\xi} \in \Xi | p^*_{\boldsymbol{\xi}} > 0} p^*_{\boldsymbol{\xi}} \sum_{i \in I} \sigma^*_{\boldsymbol{\xi},i} - \sum_{\boldsymbol{\xi} \in \Xi | p^*_{\boldsymbol{\xi}} > 0} p^*_{\boldsymbol{\xi}} \sum_{i \in I} \sum_{j \in J} x_j \xi_j \omega^*_{\boldsymbol{\xi},ij} \\ \Leftrightarrow\ & \eta \geq \sum_{\boldsymbol{\xi} \in \Xi | p^*_{\boldsymbol{\xi}} > 0} p^*_{\boldsymbol{\xi}} Q(\boldsymbol{x}^*, \boldsymbol{\xi}) + \sum_{j \in J} x_j \sum_{\boldsymbol{\xi} \in \Xi | p^*_{\boldsymbol{\xi}} > 0} p^*_{\boldsymbol{\xi}} \left[Q(\boldsymbol{x}^* \vee \boldsymbol{z}_j, \boldsymbol{\xi}) - Q(\boldsymbol{x}^*, \boldsymbol{\xi}) \right] \end{aligned}$$

where $\boldsymbol{z}_j$ and "$\vee$" are defined as in Algorithm CP. The third line comes from the facts that

$$\begin{aligned} \sum_{i \in I} \sigma^*_{\boldsymbol{\xi},i} &= \sum_{i \in I} d_i \min_{j \in J\cup\{0\} | x^*_j \xi_j = 1} c_{ij} = Q(\boldsymbol{x}^*, \boldsymbol{\xi}) \\ \sum_{i \in I} \xi_j \omega^*_{\boldsymbol{\xi},ij} &= \sum_{i \in I} \xi_j d_i \max\{0, \min_{l \in J\cup\{0\} | x^*_l \xi_l = 1} c_{il} - c_{ij}\} \\ &= \sum_{i \in I} \xi_j d_i \left(\min_{l \in J\cup\{0\} | x^*_l \xi_l = 1} c_{il} - \min\{c_{ij}, \min_{l \in J\cup\{0\} | x^*_l \xi_l = 1} c_{il}\} \right) \\ &= \xi_j \left(\sum_{i \in I} d_i \min_{l \in J\cup\{0\} | x^*_l \xi_l = 1} c_{il} - \sum_{i \in I} d_i \min\{c_{ij}, \min_{l \in J\cup\{0\} | x^*_l \xi_l = 1} c_{il}\} \right) \\ &= Q(\boldsymbol{x}^*, \boldsymbol{\xi}) - Q(\boldsymbol{x}^* \vee \boldsymbol{z}_j, \boldsymbol{\xi}). \end{aligned}$$

By inserting in the closed-form dual optimal solution, one can easily write constraint (12) in the form of constraint (5). The latter constraint is beneficial in the sense that, once the distribution $\boldsymbol{p}^*$ is available, the latter constraint can be constructed directly from the expectation of $Q(\cdot, \tilde{\boldsymbol{\xi}})$ without solving the dual problem of $Q(\boldsymbol{x}^*, \boldsymbol{\xi})$ for each scenario $\boldsymbol{\xi}$ with positive probability $p^*_{\boldsymbol{\xi}}$. In more general cases when the closed-form dual optimal solution is not available, we may not be able to derive cuts similar to those in constraint (5), but the cutting plane algorithm can still be applied by adding cuts in the form of constraint (12). To sum up, the Benders decomposition algorithm and Algorithm CP are exactly the same. We can conclude that: Algorithm CP terminates within a finite number of iterations and provides an optimal solution to model (4). This completes the proof. $\square$

A.2. Proof of Theorem 2

The following definitions and lemmas will be used in the proof of Theorem 2. Let K be a convex and compact set in $\mathbb{R}^n$.

DEFINITION 1. (Grötschel et al. 1993, Definition 2.1.1) Strong optimization problem: Given a vector $\boldsymbol{c} \in \mathbb{R}^n$, find a vector $\boldsymbol{y} \in K$ that maximizes $\boldsymbol{c}^\top \boldsymbol{x}$ on K, or assert that K is empty.

DEFINITION 2. (Grötschel et al. 1993, Definition 2.1.4) Strong separation problem: Given a vector $\boldsymbol{y} \in \mathbb{R}^n$, decide whether $\boldsymbol{y} \in K$, and if not, find a hyperplane that separates $\boldsymbol{y}$ from K; more exactly, find a vector $\boldsymbol{c} \in \mathbb{R}^n$ such that $\boldsymbol{c}^\top \boldsymbol{y} > \max\{\boldsymbol{c}^\top \boldsymbol{x} | \boldsymbol{x} \in K\}$.

DEFINITION 3. (Grötschel et al. 1993, Definition 2.1.5) Strong membership problem: Given a vector $\boldsymbol{y} \in \mathbb{R}^n$, decide whether $\boldsymbol{y} \in K$.

LEMMA 1. *(Grötschel et al. 1993, Theorem 6.4.9) Any one of the following two problems:*
- strong separation,
- strong optimization,
can be solved in oracle-polynomial time for any well-described polyhedron given by an oracle for the other problem.

LEMMA 2. *(Grötschel et al. 1993, Theorem 6.5.14) There exists an oracle-polynomial time algorithm that, for any well-described polyhedron given by a strong separation oracle and for any* $\boldsymbol{c} \in \mathbb{Q}^n$*, either*
(i) finds a basic optimum standard dual solution, or
(ii) asserts that the dual problem is unbounded or has no solution.

To put it shortly, Lemma 1 indicates the equivalence of strong separation problem and strong optimization problem in terms of polynomial-time solvability. Lemma 2 implies that the strong optimization problem of the dual problem is polynomial-time solvable as long as the strong separation problem of the primal problem is polynomial-time solvable.

The proof of Theorem 2 is presented as follows.

Proof of Theorem 2. To prove the NP-hardness of $Z_{sep}(\boldsymbol{x})$, it is sufficient to prove the NP-hardness of the following problem, which is a special case of the general $Z_{sep}(\boldsymbol{x})$ with the set P captured by the first two moments.

$$\begin{aligned} Z_{sep}(\boldsymbol{x}) = \max_{p_{\boldsymbol{\xi}} \geq 0} \ & \textstyle\sum_{\boldsymbol{\xi} \in \Xi} Q(\boldsymbol{x}, \boldsymbol{\xi}) p_{\boldsymbol{\xi}} \\ \text{s.t.} \ & \textstyle\sum_{\boldsymbol{\xi} \in \Xi} p_{\boldsymbol{\xi}} = 1, \\ & \textstyle\sum_{\boldsymbol{\xi} \in \Xi | \xi_j = 1} p_{\boldsymbol{\xi}} = q_j \qquad \forall j \in J, \\ & \textstyle\sum_{\boldsymbol{\xi} \in \Xi | \xi_{j_1} = \xi_{j_2} = 1} p_{\boldsymbol{\xi}} = q_{j_1 j_2} \quad \forall j_1, j_2 \in J, j_1 < j_2. \end{aligned} \tag{13}$$

Note that the dual problem of model (13), denoted as $Z^D_{sep}(\boldsymbol{x})$, is:

$$\begin{aligned} Z^D_{sep}(\boldsymbol{x}) = \min_{\alpha, \boldsymbol{\beta}} \ & \alpha + \textstyle\sum_{j \in J} q_j \beta_j + \sum_{j_1, j_2 \in J, j_1 < j_2} q_{j_1 j_2} \beta_{j_1 j_2} \\ \text{s.t.} \ & \alpha + \textstyle\sum_{j \in J | \xi_j = 1} \beta_j + \sum_{j_1, j_2 \in J, j_1 < j_2 | \xi_{j_1} = \xi_{j_2} = 1} \beta_{j_1 j_2} \geq Q(\boldsymbol{x}, \boldsymbol{\xi}), \quad \forall \boldsymbol{\xi} \in \Xi, \end{aligned} \tag{14}$$

where α, β_j, and $\beta_{j_1 j_2}$ are the dual variables associated with the constraints in model (13), respectively, and $\boldsymbol{\beta}$ is the vector of β_j and $\beta_{j_1 j_2}$ for all $j, j_1, j_2 \in J, j_1 < j_2$.

If we suppose that $Z_{sep}(\boldsymbol{x})$ in model (13) is polynomial-time solvable, then according to Lemma 1, its strong separation problem is also polynomial-time solvable. Consequently, the dual strong optimization problem of

$Z_{sep}(\boldsymbol{x})$, i.e., $Z^D_{sep}(\boldsymbol{x})$ in model (14), is also polynomial-time solvable (according to Lemma 2), and so is the strong separation problem of $Z^D_{sep}(\boldsymbol{x})$ (according to Lemma 1). Furthermore, the strong membership problem of $Z^D_{sep}(\boldsymbol{x})$ is also solvable, which is a trivial result from Definition 1 and Definition 3. However, we next show that the strong membership problem of $Z^D_{sep}(\boldsymbol{x})$ is NP-hard.

According to Definition 3, the strong membership problem of $Z^D_{sep}(\boldsymbol{x})$, denoted as $Z^D_{sep}(\boldsymbol{x})$-SMEM, can be written as follows:

- $Z^D_{sep}(\boldsymbol{x})$-SMEM: Given a vector $\boldsymbol{\pi}^*$ of the same dimension as $(\alpha, \boldsymbol{\beta}^\top)^\top$, decide whether $\boldsymbol{\pi}^* \in \Pi$, where Π is the feasible region of $Z^D_{sep}(\boldsymbol{x})$, i.e.,

$$\Pi := \left\{ (\alpha, \boldsymbol{\beta}^\top)^\top \;\middle|\; \alpha + \sum_{j \in J | \xi_j = 1} \beta_j + \sum_{j_1 j_2 \in J, j_1 < j_2, \xi_{j_1} = \xi_{j_2} = 1} \beta_{j_1 j_2} \geq Q(\boldsymbol{x}, \boldsymbol{\xi}), \quad \forall \boldsymbol{\xi} \in \Xi \right\}.$$

The $Z^D_{sep}(\boldsymbol{x})$-SMEM problem is equivalent to the following problem, which is denoted as $RC(\boldsymbol{x})$-DECISION:

- $RC(\boldsymbol{x})$-DECISION: Given $\boldsymbol{x} \in \{0,1\}^{|J|}$ and vector $\boldsymbol{\pi}^* = (\alpha^*, \boldsymbol{\beta}^{*\top})^\top$, decide whether $RC(\boldsymbol{x}) \leq 0$, where $RC(\boldsymbol{x})$ is defined as

$$RC(\boldsymbol{x}) := \max_{\boldsymbol{\xi} \in \Xi} \left\{ Q(\boldsymbol{x}, \boldsymbol{\xi}) - \alpha^* - \sum_{j \in J | \xi_j = 1} \beta^*_j - \sum_{j_1 j_2 \in J, j_1 < j_2, \xi_{j_1} = \xi_{j_2} = 1} \beta^*_{j_1 j_2} \right\}. \tag{15}$$

Note that $RC(\boldsymbol{x})$ is exactly the pricing problem (also known as the reduced cost problem) if we solve $Z_{sep}(\boldsymbol{x})$ using the column generation approach. The equivalence of the $Z^D_{sep}(\boldsymbol{x})$-SMEM problem and the $RC(\boldsymbol{x})$-DECISION problem can be seen from the fact that the answer to the $Z^D_{sep}(\boldsymbol{x})$-SMEM problem is "yes" if and only if that of the $RC(\boldsymbol{x})$-DECISION problem is "yes," i.e., $\boldsymbol{\pi}^* \in \Pi$ if and only if $RC(\boldsymbol{x}) \leq 0$.

We next show that the $RC(\boldsymbol{x})$-DECISION problem is NP-hard. This can be achieved by showing that there exists a NP-complete decision problem that reduces to the $RC(\boldsymbol{x})$-DECISION problem. Consider the following decision version of the MAXCUT problem (denoted as MAXCUT-DECISION), which is a well-known NP-complete problem: (Gary and Johnson 1979)

- MAXCUT-DECISION: Given a simple graph $G = (V, E)$ and an integer $K \in \mathbb{Z}^+$, decide whether there is a cut of size at least K, or more exactly, whether there is a partition of V into two disjoint sets V_1 and V_2 such that the quantity of edges between V_1 and V_2 is at least K.

The MAXCUT-DECISION problem is equivalent to: decide whether

$$\max_{y_i \in \{-1,1\}} \left\{ \frac{1}{4} \sum_{i,j \in V, i \neq j} w_{ij}(1 - y_i y_j) \right\} \geq K,$$

where $w_{ij} = 1$ for any $(i,j) \in E$, and otherwise, $w_{ij} = 0$. The decision variable y_i for all $i \in V$ represents the partition of V, i.e., $y_i = 1$ if $i \in V_1$ and $y_i = -1$ if $i \in V_2$. Note that the size of cut (the quantity of edges) can only take integer values and $K \in \mathbb{Z}^+$. Thus the answer to the MAXCUT-DECISION problem is "yes" if and only if the answer to the following problem is "no": decide whether

$$\max_{y_i \in \{-1,1\}} \left\{ \frac{1}{4} \sum_{i,j \in V, i \neq j} w_{ij}(1 - y_i y_j) \right\} \leq K - 1.$$

Given any instance of the MaxCut-Decision problem characterized by $G=(V,E)$ and K, we can construct in polynomial time an equivalent instance of the $RC(\boldsymbol{x})$-Decision problem. Recall that $Q(\boldsymbol{x},\boldsymbol{\xi})=\sum_{i\in I} d_i \min_{j\in J\cup\{0\}|x_j\xi_j=1} c_{ij}$. Let $c_{ij}=c_{i0}=0$ for all $i\in I$ and $j\in J$. Then we have $Q(\boldsymbol{x},\boldsymbol{\xi})=0$ and hence

$$RC(\boldsymbol{x})=-\alpha+\max_{\xi_j\in\{0,1\}}\left\{-\sum_{j\in J}\beta_j\xi_j-\frac{1}{2}\sum_{j_1,j_2\in J,j_1\neq j_2}\beta_{j_1j_2}\xi_{j_1}\xi_{j_2}\right\}.$$

Define a new decision variable $y_j:=2\xi_j-1$, i.e., $\xi_j=(y_j+1)/2$ for any $j\in J$. We have $y_j\in\{-1,1\}$ for all $j\in J$ and

$$\begin{aligned}RC(\boldsymbol{x})&=-\alpha+\max_{y_j\in\{-1,1\}}\left\{-\sum_{j\in J}\beta_j\cdot\frac{y_j+1}{2}-\frac{1}{2}\sum_{\substack{j_1,j_2\in J,\\ j_1\neq j_2}}\beta_{j_1j_2}\cdot\frac{y_{j_1}+1}{2}\cdot\frac{y_{j_2}+1}{2}\right\}\\&=-\alpha+\max_{y_j\in\{-1,1\}}\left\{\left(-\frac{1}{2}\sum_{j\in J}\beta_j-\frac{1}{8}\sum_{j_1,j_2\in J,j_1\neq j_2}\beta_{j_1j_2}\right)\right.\\&\qquad\left.-\sum_{j\in J}\left(\frac{1}{2}\beta_j+\frac{1}{4}\sum_{j'\in J,j'\neq j}\beta_{jj'}\right)\cdot y_j-\sum_{j_1,j_2\in J,j_1\neq j_2}\frac{1}{8}\beta_{j_1j_2}y_{j_1}y_{j_2}\right\}.\end{aligned}$$

Let $J=V$ and

$$\begin{cases}\dfrac{1}{8}\beta_{j_1j_2}=\dfrac{1}{4}w_{j_1j_2}, & \forall j_1,j_2\in J, j_1\neq j_2,\\ \dfrac{1}{2}\beta_j+\dfrac{1}{4}\sum_{j'\in J,j'\neq j}\beta_{jj'}=0, & \forall j\in J,\\ -\dfrac{1}{2}\sum_{j\in J}\beta_j-\dfrac{1}{8}\sum_{j_1,j_2\in J,j_1\neq j_2}\beta_{j_1j_2}=\dfrac{1}{4}\sum_{j_1,j_2\in J,j_1\neq j_2}w_{j_1j_2}, & \\ -\alpha=1-K & \end{cases}$$

i.e.,

$$\beta_{j_1j_2}=2w_{j_1j_2}\ \forall j_1,j_2\in J,j_1\neq j_2,\quad \beta_j=-\sum_{j'\in J,j'\neq j}w_{jj'}\ \forall j\in J,\quad \alpha=K-1.$$

Then, it is straightforward that this specific instance of the $RC(\boldsymbol{x})$-Decision problem is equivalent to the counterpart of the MaxCut-Decision problem, in the sense that the former one is a YES-instance if and only if the latter one is a NO-instance. Given any instance of the MaxCut-Decision problem, we can find its answer by considering an equivalent instance of the $RC(\boldsymbol{x})$-Decision problem. In other words, the MaxCut-Decision problem can be transformed to the $RC(\boldsymbol{x})$-Decision problem in polynomial time. It shows that the $RC(\boldsymbol{x})$-Decision problem is "at least as hard" as the MaxCut-Decision problem. Thus, the $RC(\boldsymbol{x})$-Decision problem is NP-hard, and so is the $Z^D_{sep}(\boldsymbol{x})$-SMEM problem. According to the discussion in the beginning of this proof, $Z_{sep}(\boldsymbol{x})$ is also NP-hard, and this completes the proof. □

A.3. Proof of Theorem 3

Proof. Substitute the expression of $Q(\boldsymbol{x},\boldsymbol{\xi})$, and $RC(\boldsymbol{x})$ can be formulated as

$$RC(\boldsymbol{x})=\max_{\boldsymbol{\xi}\in\Xi}\left\{\sum_{i\in I}d_i\min_{j\in J\cup\{0\}|x_j\xi_j=1}c_{ij}-\alpha-\sum_{k=1}^{n}(\overline{\beta}_k-\underline{\beta}_k)\prod_{j\in A_k}\xi_j\prod_{j'\in B_k}(1-\xi_{j'})\right\}.$$

Let π_i represent the term $\min_{j\in J\cup\{0\}|x_j\xi_j=1}c_{ij}$ for any $i\in I$. We can write $RC(\boldsymbol{x})$ as

$$\begin{aligned}RC(\boldsymbol{x})=\max_{\boldsymbol{\xi}\in\Xi,\boldsymbol{\pi}}\ &\textstyle\sum_{i\in I}d_i\pi_i-\alpha-\sum_{k=1}^{n}(\overline{\beta}_k-\underline{\beta}_k)\prod_{j\in A_k}\xi_j\prod_{j'\in B_k}(1-\xi_{j'})\\ \text{s.t.}\ \ &\pi_i\leq c_{ij}x_j\xi_j+c_{i0}(1-x_j\xi_j),\quad \forall i\in I,j\in J.\end{aligned}$$

Consider the term $-(\overline{\beta}_k - \underline{\beta}_k)\prod_{j\in A_k}\xi_j\prod_{j'\in B_k}(1-\xi_{j'})$ for any $k\in\{1,...,n\}$ in the objective function. We can introduce a new decision variable λ_k to represent $\prod_{j\in A_k}\xi_j\prod_{j'\in B_k}(1-\xi_{j'})$. Then the term $-(\overline{\beta}_k - \underline{\beta}_k)\prod_{j\in A_k}\xi_j\prod_{j'\in B_k}(1-\xi_{j'})$ can be replaced with $-(\overline{\beta}_k - \underline{\beta}_k)\lambda_k$ by adding the constraints

$$\lambda_k \le \xi_j,\ \forall j\in A_k, \quad \lambda_k \le 1-\xi_j,\ \forall j\in B_k, \quad \lambda_k \ge \sum_{j\in A_k}\xi_j + \sum_{j\in B_k}(1-\xi_j) - |A_k| - |B_k| + 1, \quad \text{and} \quad \lambda_k \ge 0.$$

This immediately yields the equivalent reformulation in model (9). $\square$

Appendix B: Supplementary Discussions

B.1. Model Generalization to the Cases with Uncertain Demand

In Section 1, we claim that the proposed model and algorithm can be extended in the following cases with uncertain demand if: i) the demand is independent of the facility disruption, ii) the demand is defined in each scenario of facility disruption in the scenario-based stochastic model, and iii) the demand is defined as a linear function of facility disruption, e.g., the settings in An et al. (2014) and Azad and Hassini (2019). The first two cases are trivial from the model formulation in Section 3 and Section 4. In this section, we illustrate the third case in detail following the setting in An et al. (2014).

If the disruption-dependent demand is denoted as $(1-\theta\xi_i)d_i$, i.e., a linear function on disruption status ξ_i with parameter $\theta\in(-\infty,1]$ indicating how demand is affected by disruption, we then update the definition of $Q(\boldsymbol{x},\boldsymbol{\xi})$ as

$$Q(\boldsymbol{x},\boldsymbol{\xi}) := \sum_{i\in I}(1-\theta\xi_i)d_i \min_{j\in J\cup\{0\}|x_j\xi_j=1} c_{ij}.$$

Note that Theorem 1 holds for this new $Q(\boldsymbol{x},\boldsymbol{\xi})$, because the Benders decomposition is derived based on scenarios and the Benders cut is simplified for each given scenario $\boldsymbol{\xi}$ in the proof of Theorem 1.

With this new $Q(\boldsymbol{x},\boldsymbol{\xi})$, the reduced cost is as follows,

$$RC(\boldsymbol{x}) = \max_{\boldsymbol{\xi}\in\Xi}\left\{\sum_{i\in I}(1-\theta\xi_i)d_i \min_{j\in J\cup\{0\}|x_j\xi_j=1} c_{ij} - \alpha - \sum_{k=1}^{n}(\overline{\beta}_k - \underline{\beta}_k)\prod_{j\in A_k}\xi_j\prod_{j'\in B_k}(1-\xi_{j'})\right\}.$$

Note that $(1-\theta\xi_i)\ge 0$ for all $\theta\in(-\infty,1]$. Thus $RC(\boldsymbol{x})$ can be written as

$$\begin{array}{rll} RC(\boldsymbol{x}) = \max\limits_{\boldsymbol{\xi}\in\Xi,\boldsymbol{\pi},\boldsymbol{\delta}} & \sum_{i\in I}d_i\delta_i - \alpha - \sum_{k=1}^{n}(\overline{\beta}_k - \underline{\beta}_k)\prod_{j\in A_k}\xi_j\prod_{j'\in B_k}(1-\xi_{j'}) & \\ \text{s.t.} & \delta_i \le \pi_i + M\xi_i, & \forall i\in I, \\ & \delta_i \le (1-\theta)\pi_i + M(1-\xi_i), & \forall i\in I, \\ & \pi_i \le c_{ij}x_j\xi_j + c_{i0}(1-x_j\xi_j), & \forall i\in I, j\in J, \end{array}$$

where M denotes a big constant. As shown in the proof of Theorem 3, we can linearize the remaining multiplication terms in the objective function, so $RC(\boldsymbol{x})$ can be also written as a mixed integer linear program.

To sum up, this paper can be extended to the demand setting in An et al. (2014) where the demand is a linear function of facility disruption.

B.2. The Scalability of Algorithm CP with More Cross Moment Information

In Section 5.2, we apply the Algorithm CP to solve the robust cross moment model, where n pieces of information are used to determine the ambiguity set P, including the marginal disruption probabilities and some of the cross disruption probabilities. To illustrate the scalability of the proposed algorithm with larger n values, we conduct additional numerical studies in this section.

In Section 5.2, the partial cross moment is set by assuming independent disruptions if the distance between any two facilities is no less than a distance threshold, which is set to 2500 km (kilometer). In this section, we change the distance threshold to generate multiple instances with different cross moment information and different n values. This numerical study is based on the instances with 20 nodes, $\beta = 0.2$ and $\theta = 800$. Because the maximal distance between two facilities in the data set with 20 nodes is 2701 km, we change the distance threshold from 2800 km to 2000 km with a step size of 100 km. We also solve instances with the distance threshold being 1500 km and 1000 km, respectively. Apparently, the smaller the distance threshold, the larger the n value, i.e., the more cross moment information is available. Note that the changing values of n in this numerical study do not affect the solution of the marginal moment model and the stochastic model assuming independent disruption between facilities. For all three models, the optimality gap (i.e., the relative gap between the best upper bound and the best lower bound) is set to be 0.01%, rather than 0.5% for the results in Table 2, to get more precise results.

Table 4 Numerical results with different n values

Model	Distance threshold	n	CPU time (s)			#Cut	#Open	Obj ($\times 10^5$)	MM model		Stochastic model	
			Total	Z_{sep}	RC				Regret ($\times 10^5$)	Regret% (%)	Regret ($\times 10^5$)	Regret% (%)
Stochastic	-	-	0.4	-	-	16	4	5.48	-	-	-	-
MM	-	-	0.3	-	-	13	4	12.72	-	-	-	-
CM	2800	20	0.7	0.3	0.3	13	4	12.72	0	0	3.44	27.04
	2700	21	1.1	0.4	0.4	19	5	9.98	2.74	27.51	6.18	61.99
	2600	22	1.1	0.4	0.4	14	5	9.44	3.28	34.71	6.72	71.14
	2500	27	2.4	1.5	1.4	21	5	8.44	4.28	50.64	7.72	91.38
	2400	34	3.8	2.4	2.2	25	4	8.19	0	0	7.97	97.32
	2300	40	4.6	3.4	2.9	19	4	7.19	1.00	13.96	8.97	124.87
	2200	43	3.8	2.9	2.4	27	4	7.19	1.00	13.96	8.97	124.87
	2100	46	7.3	5.9	5.2	37	4	7.19	1.00	13.96	8.97	124.87
	2000	51	7.8	6.5	5.6	25	4	6.72	1.46	21.69	0	0
	1500	83	54.4	52.3	47.7	37	4	6.48	1.27	19.60	0	0
	1000	117	176.7	173.9	163.3	43	4	6.36	1.34	21.01	0.12	1.82

The results are presented in Table 4. We first solve the stochastic model (cf. "Stochastic") and the marginal moment model (cf. "MM"). For the cross moment model (cf. "CM"), we consider 11 instances with different distance thresholds ranging from 2800 km to 1000 km (cf. column "Distance threshold") and accordingly different n values from 20 to 117 (cf. column "n"). For each instance, the solution time is recorded in the columns titled "CPU time." The total solution time, the cumulative solution time for Z_{sep}, and the cumulative solution time for RC are presented in the columns titled "Total," "Z_{sep}," and "RC," respectively. The number of cutting plane cuts, the number of open facilities in the optimal solution, and the optimal objective value are recorded in the columns titled "#Cut," "#Open," and "Obj." For the MM model or the stochastic model, the regret and relative regret of ignoring cross moment information or assuming independence are reported in the columns titled "Regret" and "Regret%" under the name "MM model" or "Stochastic model," respectively.

It can be observed from Table 4 that, with the decrease of the distance threshold, i.e., the increase of the n value, the CPU time of the CM model gradually increases and more CP cuts are generated. This is reasonable because larger n values indicate more cross moment information, and it takes more efforts for the CM model to calculate the worst-case expected transportation cost, balance the transportation cost with the fixed location cost, and determine the optimal decision.

Compare the CM model with the MM model. For the instance with 2800 km distance threshold, i.e., $n = 20$, indicating that no cross moment information is available, the CM model gives the same solution as the MM model and the Regret% under the MM model is 0%. With the increase of the n value, the Regret% of the MM model generally increases, ranging from 13.96% to 50.64%. Even for the instance with 2700 km distance threshold, i.e., $n = 21$, indicating that the available cross moment information is only one pair of facilities having independent disruptions, the Regret% of the MM model is 27.51%. For certain instances, e.g., the instance with 2400 km distance threshold in Table 4, the CM model gives the same solution as the MM model regardless of the cross moment information.

Compare the CM model with the stochastic model. As the n value increases, the Regret% of the stochastic model firstly increases from 27.04% to 124.87% and then suddenly drops to around 0%. Note that the cross moment information for the CM model is actually the independence between facility disruptions. The higher n value indicates more pairs of facilities have independent disruptions. Therefore, the sudden drop of the Regret% of the stochastic model comes from the fact that the CM model may give the same or similar solution as that of the stochastic model, given certain facility pairs with independent disruptions. This is consistent with another observation that, with the increasing value of n, i.e., more facility pairs are assumed to have independent disruptions, the number of open facilities by the CM model first increases from 4 facilities (which equals that of the MM solution) to 5 facilities and then decreases back to 4 facilities (which equals that of the stochastic solution).

B.3. Detailed Numerical Results Based on Randomly Generated Instances

In Section 5.2, the numerical experiment is conducted based on the existing data set, where each combination of (Nodes, β, θ) in Table 2 corresponds to only one instance with all parameters calculated according to the data set. In order to have more observations on how the values of Regret% and the numbers of open facilities for different models change with β and θ, we further conduct this numerical study based on randomly generated instances with 20 nodes. We randomly generate the location, demand, and fixed facility setup cost of each node, within the same range as those in the data set, respectively. The unit transportation cost and the marginal/cross disruption probability are calculated based on these randomly generated data. To see how Regret% changes with the value of β, we fix θ to be 800 and set the value of β from 0.025 to 0.300 with a step length of 0.025. To see how Regret% changes with the value of θ, we fixed β to be 0.2 and set the value of θ from 200 to 800 with a step length of 50. For each value of β or θ, 100 scenarios are randomly generated, and the average values are reported in Table 5. The average Regret% is also presented in Figure 1, while the average #Open is presented in Figure 2.

Table 5 Numerical results of random instances

Instance			CM model	MM model		Stochastic model	
Nodes	β	θ	#Open	#Open	Regret% (%)	#Open	Regret% (%)
20	0.025	800	8.21	7.97	4.56	7.83	6.44
20	0.050	800	8.23	8.01	10.33	7.87	14.59
20	0.075	800	8.14	8.07	14.65	7.93	22.65
20	0.100	800	8.29	8.14	19.05	7.99	28.70
20	0.125	800	8.32	8.16	22.58	8.00	35.25
20	0.150	800	8.29	8.19	24.99	8.01	41.22
20	0.175	800	8.21	8.17	28.57	8.03	46.60
20	0.200	800	8.33	8.11	31.40	8.08	50.54
20	0.225	800	8.28	8.15	33.66	8.13	54.06
20	0.250	800	8.29	8.15	34.78	8.19	58.06
20	0.275	800	8.32	8.16	36.79	8.23	60.37
20	0.300	800	8.23	8.10	40.68	8.24	63.30
20	0.200	200	7.84	7.81	0.07	7.80	0.07
20	0.200	250	7.88	7.87	0.04	7.81	0.39
20	0.200	300	8.04	7.92	0.82	7.89	1.16
20	0.200	350	8.22	8.03	1.86	7.93	3.32
20	0.200	400	8.26	8.04	4.06	7.96	6.76
20	0.200	450	8.26	8.14	6.96	7.92	12.55
20	0.200	500	8.28	8.16	10.03	7.97	17.40
20	0.200	550	8.28	8.14	14.24	7.98	22.88
20	0.200	600	8.32	8.16	16.56	7.96	29.15
20	0.200	650	8.30	8.16	20.68	7.98	35.70
20	0.200	700	8.25	8.20	24.33	8.02	40.85
20	0.200	750	8.27	8.15	28.12	8.04	46.21
20	0.200	800	8.33	8.11	31.40	8.08	50.54

Table 5 reveals the same observations on the values of Regret% of the MM and the stochastic models, which have been discussed in details in Section 5.2. The values of Regret% of the MM model and the stochastic model both increase with β and θ. Moreover, the value of Regret% of the stochastic model is generally larger than that of the MM model.

Table 5 and Figure 2 provide additional observations on the numbers of open facilities for different models (cf. column "#Open"). First, for any of these models, the number of open facilities generally increases with

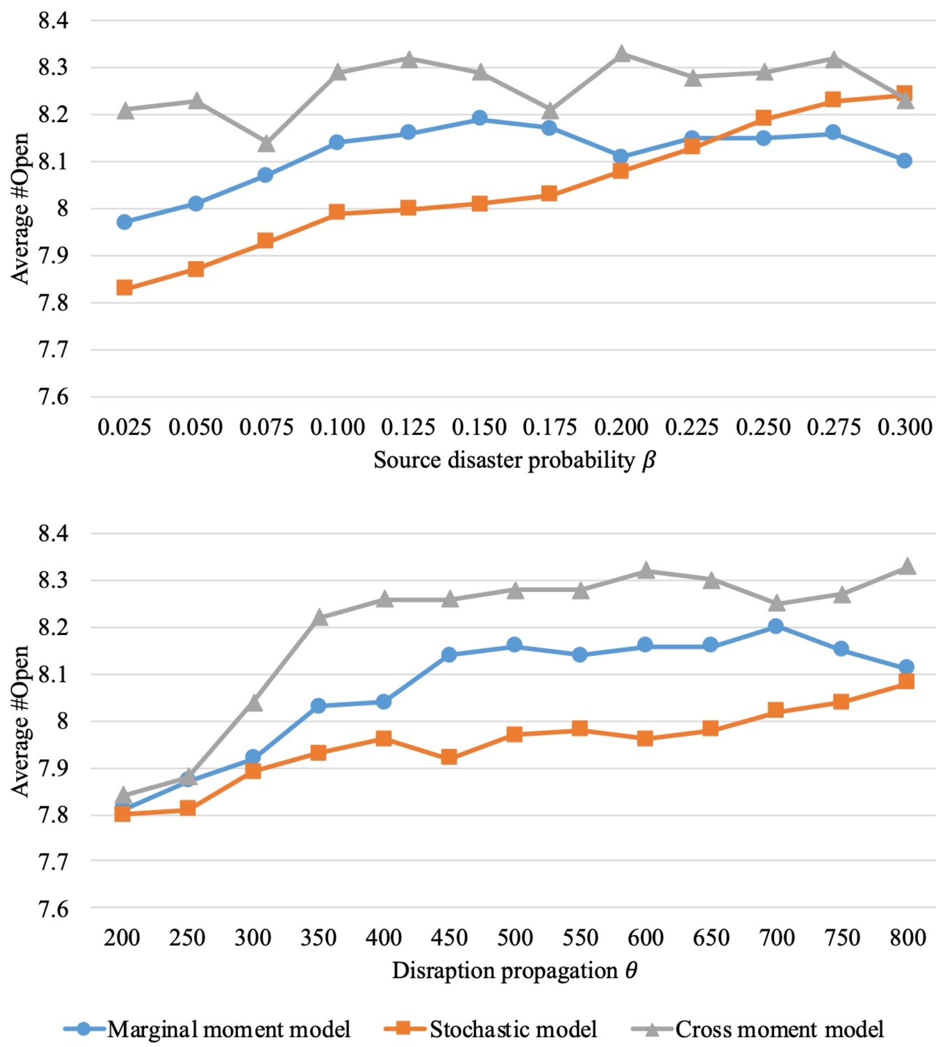

Figure 2 **Impact of β and θ on average #Open**

the values of β and θ (i.e., facility disruption probability). This is consistent with the observations in Section 4.2 and also the intuition that more facilities should be open under higher risks of disruptions. Second, for most cases, the cross moment model tends to open the largest number of facilities among the three models, while the stochastic model tends to open the least number of facilities for most instances with moderate values of β and θ, e.g., the instances with β below 0.225 and θ equal to 800. This is consistent with the observations for most instances in Section 4.2. Third, with the increasing values of β and θ, the number of open facilities by the stochastic model increases gradually and may exceed that of the marginal moment model or even the cross moment model when β is large enough. This could be explained by the fact that the stochastic model makes the location decision assuming independent disruptions for all the facility pairs. Thus, with the increase of facility disruption probability, the stochastic model tends to be more "confident" on opening more facilities with the knowledge of independent facility disruptions.

References

Aboolian R, Cui T, Shen ZJM (2013) An efficient approach for solving reliable facility location models. *INFORMS Journal on Computing* 25(4):720–729.

Ahmadi-Javid A, Seddighi AH (2013) A location-routing problem with disruption risk. *Transportation Research Part E: Logistics and Transportation Review* 53:63–82.

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

Azad N, Hassini E (2019) A benders decomposition method for designing reliable supply chain networks accounting for multimitigation strategies and demand losses. *Transportation Science* 53(5):1287–1312.

Azad N, Saharidis GK, Davoudpour H, Malekly H, Yektamaram SA (2013) Strategies for protecting supply chain networks against facility and transportation disruptions: an improved benders decomposition approach. *Annals of Operations Research* 210(1):125–163.

Bamberger RL, Kumins L (2005) Oil and gas: Supply issues after katrina. Technical report, Congressional Research Service, Library of Congress.

Bayraksan G, Love DK (2015) Data-driven stochastic programming using phi-divergences. *The Operations Research Revolution*, 1–19 (INFORMS).

Ben-Tal A, Den Hertog D, De Waegenaere A, Melenberg B, Rennen G (2013) Robust solutions of optimization problems affected by uncertain probabilities. *Management Science* 59(2):341–357, ISSN 0025-1909.

Berman O, Krass D (2011) On n-facility median problem with facilities subject to failure facing uniform demand. *Discrete Applied Mathematics* 159(6):420–432.

Berman O, Krass D, Menezes MB (2013) Location and reliability problems on a line: Impact of objectives and correlated failures on optimal location patterns. *Omega* 41(4):766–779.

Bertsimas D, Doan XV, Natarajan K, Teo CP (2010) Models for minimax stochastic linear optimization problems with risk aversion. *Mathematics of Operations Research* 35(3):580–602, ISSN 0364-765X.

Bertsimas D, Popescu I (2005) Optimal inequalities in probability theory: A convex optimization approach. *SIAM Journal on Optimization* 15(3):780–804.

Cashell BW, Labonte M (2005) The macroeconomic effects of Hurricane Katrina. Technical report, Congressional Research Service, Library of Congress.

Chen X, Sim M, Sun P, Zhang J (2008) A linear decision-based approximation approach to stochastic programming. *Operations Research* 56(2):344–357, ISSN 0030-364X.

Chongvilaivan A (2012) Thailand's 2011 flooding: Its impact on direct exports and global supply chains. Technical report, ARTNeT working paper series.

Christopher M, Peck H (2004) Building the resilient supply chain. *The International Journal of Logistics Management* 15(2):1–14.

Church RL, Scaparra MP (2007) Protecting critical assets: the r-interdiction median problem with fortification. *Geographical Analysis* 39(2):129–146.

Cui T, Ouyang Y, Shen ZJM (2010) Reliable facility location design under the risk of disruptions. *Operations Research* 58(4):998–1011.

Delage E, Ye Y (2010) Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research* 58(3):595–612, ISSN 0030-364X.

El Ghaoui L, Oks M, Oustry F (2003) Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. *Operations Research* 51(4):543–556, ISSN 0030-364X.

Esfahani PM, Kuhn D (2018) Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming* 171(1-2):115–166, ISSN 0025-5610.

Gary MR, Johnson DS (1979) *Computers and Intractability: A Guide to the Theory of NP-completeness* (WH Freeman and Company, New York).

Goh J, Sim M (2010) Distributionally robust optimization and its tractable approximations. *Operations Research* 58(4-part-1):902–917.

Golany B, Kaplan EH, Marmur A, Rothblum UG (2009) Nature plays with dice–terrorists do not: Allocating resources to counter strategic versus probabilistic risks. *European Journal of Operational Research* 192(1):198–208.

Grötschel M, Lovász L, Schrijver A (1993) *Geometric Algorithms and Combinatorial Optimization*, volume 2 (Springer Science & Business Media).

Haraguchi M, Lall U (2015) Flood risks and impacts: A case study of thailands floods in 2011 and research questions for supply chain decision making. *International Journal of Disaster Risk Reduction* 14:256–272.

Kuhn D, Wiesemann W, Georghiou A (2011) Primal and dual linear decision rules in stochastic and robust optimization. *Mathematical Programming* 130(1):177–209.

Latour A (2001) A fire in albuquerque sparks crisis for european cell-phone giants: Nokia handles shock with aplomb as ericsson of sweden gets burned. The Wall Street Journal, URL `https://www.wsj.com/articles/SB980720939804883010`.

Li X, Ouyang Y (2010) A continuum approximation approach to reliable facility location design under correlated probabilistic disruptions. *Transportation Research Part B: Methodological* 44(4):535–548.

Li X, Ouyang Y, Peng F (2013) A supporting station model for reliable infrastructure location design under interdependent disruptions. *Transportation Research Part E: Logistics and Transportation Review* 60:80–93.

Li Y, Shu J, Song M, Zhang J, Zheng H (2017) Multisourcing supply network design: two-stage chance-constrained model, tractable approximations, and computational results. *INFORMS Journal on Computing* 29(2):287–300, ISSN 1091-9856.

Liberatore F, Scaparra MP, Daskin MS (2011) Analysis of facility protection strategies against an uncertain number of attacks: the stochastic r-interdiction median problem with fortification. *Computers & Operations Research* 38(1):357–366.

Liberatore F, Scaparra MP, Daskin MS (2012) Hedging against disruptions with ripple effects in location analysis. *Omega* 40(1):21–30.

Lim MK, Bassamboo A, Chopra S, Daskin MS (2013) Facility location decisions with random disruptions and imperfect estimation. *Manufacturing & Service Operations Management* 15(2):239–249.

Liu C, Fan Y, Ordóñez F (2009) A two-stage stochastic programming model for transportation network protection. *Computers & Operations Research* 36(5):1582–1590.

Lu M, Ran L, Shen ZJM (2015) Reliable facility location design under uncertain correlated disruptions. *Manufacturing & Service Operations Management* 17(4):445–455.

Lutter P, Degel D, Büsing C, Koster AM, Werners B (2017) Improved handling of uncertainty and robustness in set covering problems. *European Journal of Operational Research* 263(1):35–49.

O'Hanley JR, Church RL (2011) Designing robust coverage networks to hedge against worst-case facility losses. *European Journal of Operational Research* 209(1):23–36.

Popescu I (2007) Robust mean-covariance solutions for stochastic optimization. *Operations Research* 55(1):98–112, ISSN 0030-364X.

Qi L, Shen ZJM, Snyder LV (2010) The effect of supply disruptions on supply chain design decisions. *Transportation Science* 44(2):274–289.

Ruvo C (2018) Typhoon impacts some promo product-producing factories in china. Advertising Specialty Institute, URL `https://www.asicentral.com/news/newsletters/promogram/september-2018/typhoon-impacts-some-promo-product-producing-factories-in-china/`.

Santos MC, Luss H, Nace D, Poss M (2019) Proportional and maxmin fairness for the sensor location problem with chance constraints. *Discrete Applied Mathematics* 261:316–331.

Scaparra MP, Church RL (2008a) A bilevel mixed-integer program for critical infrastructure protection planning. *Computers & Operations Research* 35(6):1905–1923.

Scaparra MP, Church RL (2008b) An exact solution approach for the interdiction median problem with fortification. *European Journal of Operational Research* 189(1):76–92.

Scarf H (1958) *A min-max solution of an inventory problem*, 201–209 (California: Stanford University Press).

See CT, Sim M (2010) Robust approximation to multiperiod inventory management. *Operations Research* 58(3):583–594, ISSN 0030-364X.

Shapiro A (2006) Worst-case distribution analysis of stochastic programs. *Mathematical Programming* 107(1-2):91–96, ISSN 0025-5610.

Sheffi Y (2001) Supply chain management under the threat of international terrorism. *The International Journal of Logistics Management* 12(2):1–11.

Shen ZJM, Zhan RL, Zhang J (2011) The reliable facility location problem: Formulations, heuristics, and approximation algorithms. *INFORMS Journal on Computing* 23(3):470–482.

Snyder LV, Atan Z, Peng P, Rong Y, Schmitt AJ, Sinsoysal B (2016) OR/MS models for supply chain disruptions: A review. *IIE Transactions* 48(2):89–109.

Snyder LV, Daskin MS (2005) Reliability models for facility location: the expected failure cost case. *Transportation Science* 39(3):400–416.

Snyder LV, Scaparra MP, Daskin MS, Church RL (2006) Planning for disruptions in supply chain networks. *Tutorials in Operations Research* 2:234–257.

Tang CS (2006) Robust strategies for mitigating supply chain disruptions. *International Journal of Logistics: Research and Applications* 9(1):33–45.

Wiesemann W, Kuhn D, Sim M (2014) Distributionally robust convex optimization. *Operations Research* 62(6):1358–1376.

Xie S, Li X, Ouyang Y (2015a) Decomposition of general facility disruption correlations via augmentation of virtual supporting stations. *Transportation Research Part B: Methodological* 80:64–81.

Xie W, Ouyang Y, Wong SC (2015b) Reliable location-routing design under probabilistic facility disruptions. *Transportation Science* 50(3):1128–1138, ISSN 0041-1655.

Xu H, Liu Y, Sun H (2018) Distributionally robust optimization with matrix moment constraints: Lagrange duality and cutting plane methods. *Mathematical Programming* 169(2):489–529, ISSN 0025-5610.

Zeng B, Zhao L (2013) Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters* 41(5):457–461.