Allocating Protection Resources to Facilities When the Effect of Protection is Uncertain

Hugh R. Medal, Edward A. Pohl, Manuel D. Rossetti

Abstract

We study a new facility protection problem in which one must allocate scarce protection resources to a set of facilities given that allocating resources to a facility only has a *probabilistic* effect on the facility's post-disruption capacity. This study seeks to test three common assumptions made in the literature on modeling infrastructure systems subject to disruptions: 1) *perfect protection*, e.g., protecting an element makes it fail-proof, 2) *binary protection*, i.e., an element is either fully protected or unprotected, and 3) *binary state*, i.e., disrupted elements are fully operational or non-operational. We model this facility protection problem as a two-stage stochastic program with endogenous uncertainty. Because this stochastic program is non-convex we present a greedy algorithm and show that it has a worst-case performance of 0.63. However, empirical results indicate that the average performance is much better. In addition, experimental results indicate that the mean-value version of this model, in which parameters are set to their mean values, performs close to optimal. Results also indicate that the perfect and binary protection assumptions together significantly affect the performance of a model. On the other hand, the binary state assumption was found to have a smaller effect.

Keywords: Facility protection; Stochastic programming; Catastrophe planning and management

1 Introduction

This paper examines the problem of allocating scare protection resources amongst a set of facilities to mitigate the system-wide consequences of disruptions, extending previous facility protection optimization models by adding several types of model fidelity: (1) Rather than assuming a facility is either hardened or not hardened, we model *multi-level protection*, in which a facility's protection level increases as more protection resources are allocated to the facility. (2) Rather than assuming a protected facility never fails and an unprotected facility always fails, we model *imperfect protection*, in which the facility's likelihood of failure depends on the level of protection. (3) Rather than assuming that a facility is either fully operational or non-operational, we model *multi-state capacity*, in which a facility's post-disruption capacity state depends probabilistically on its protection level. We call this problem the *probabilistic facility protection problem* (PFPP).

1.1 Background and Motivation

Recent history shows that disruptions do occur in infrastructure networks and that these disruptions can have large system-wide consequences. On March 11, 2011, a tsunami smashed into Japan's northeastern coast, resulting in the loss of many lives, the damage of a large amount of property, and a near nuclear disaster. The destruction caused by the tsunami affected the supply chains of many companies operating in Japan. At Renesas Electronics Corporation sensitive equipment was damaged, bringing production to a halt. Unfortunately, several Japanese automakers used Renesas as their only source for microchip controllers, causing their supply chains to have a vulnerable link. After this link was severed, some automakers were forced to halt car production for up to six months due to lack of supply (Kim, 2012). Toyota vows to be prepared for the next disruption, asking their suppliers to either spread production or hold extra stock. "Our plan is to manage risk while at the same time reducing costs," said vice president Shinichi Sasaki (Kim, 2012).

To help companies like Toyota mitigate against disruptions to their distribution networks, researchers have developed models for allocating resources among networks to minimize the risk to the overall network. In addition to private companies, government agencies such as the United States Department of Homeland Security (DHS) Office of Infrastructure Protection are also interested in allocating resources to protect infrastructure networks. The infrastructure protection problem seeks to allocate scarce resources amongst the elements of an infrastructure network (e.g., bridges, buildings, ports, etc.) in order to hedge against the network-wide consequences of disruptions.

1.2 Related Literature

While many have studied the facility protection problem, there are three important limitations of the existing research. (1) Most models assume binary protection, meaning that a facility is either protected or not. (2) Most models assume perfect protection, meaning that a protected facility cannot fail. (3) Most models assume that a disruption has a perfect impact on an element, causing it to fail or leaving it unaffected. While these assumptions have helped researchers develop tractable models and discover interesting findings, they may not be appropriate for all contexts. In reality, protection is multi-level and imperfect, and a disruption may have an imperfect effect, leaving a facility only partially degraded. Thus, an important question remains in this area of research: how do these assumptions affect the quality of solutions produced by a model? In addition, because most of the modeling approaches for facility protection rely on these assumptions, it is not clear how to relax these assumptions without substantial loss of model tractability.

The goal of this paper is to relax these assumptions while still providing a tractable solution approach. Toward this end, we model a facility protection problem with imperfect, multi-level protection and imperfect disruptions, namely the PFPP. The broader goal is to increase what is known about the implications of various assumptions on infrastructure protection models.

In the past decade there has been an increase of research on infrastructure protection modeling. One of the earliest papers, written by Church and Scaparra (2007), studied the problem of protecting facilities in order to mitigate against the worst-case disruption scenario, which consists of the failure of r facilities. This model is appropriate for protecting against an attacker with a known amount of resources. They show that protecting facilities can produce a significant reduction in the consequences due to disruptions. The authors later developed more advanced solution methods for the problem (Scaparra and Church, 2008b,a) and others have made various extensions such a stochastic number of facility failures (Liberatore *et al.*, 2011) and a knapsack budget constraint on the attacker's budget (Aksen *et al.*, 2010). In all of these studies, the models assume binary, perfect protection and perfect disruptions. In addition, each of these studies assume that facilities have infinite capacity. Scaparra and Church (2012) modeled facilities as having two capacity states, non-operational with zero capacity and fully operational with finite capacity, and found that explicitly modeling capacity does improve solution quality. However, an important question remains: Is it important to include multiple, i.e., more than two, levels of capacity?

Others have studied the problem of protecting network elements, i.e., nodes and arcs, against the worst-case disruption scenario. Bier *et al.* (2007) studied the problem of protecting a power grid against attacks. Yao *et al.* (2007) studied a similar problem but apply a more advanced solution procedure. **?** modeled the protection of networks that are subject to an attack by an attacker that uses heuristics to allocate resources. Cappanera and Scaparra (2011) presented an approach for protecting a network so as to minimize the shortest path length after an attack. All of these models assume binary, perfect protection and perfect disruptions.

Researchers have also studied infrastructure protection against random disruptions, starting with a tutorial paper by Snyder *et al.* (2006), who presented a model for protecting facilities. O'Hanley *et al.* (2007) studied the problem of protecting natural reserve sites against random failures. Lim *et al.* (2010) presented an extension of the warehouse location problem in which the decision-maker chooses between locating unreliable facilities and locating perfectly reliable, i.e., hardened backup facilities, at a higher cost. Liu *et al.* (2009) model the protection of bridges with the objective of minimizing the expected total travel time. These models also assume binary, perfect protection and perfect disruptions.

Some studies have relaxed Assumptions 1 (binary protection) and 2 (perfect protection):

Assumptions 1 and 2 Du and Peeta (2014) extended the work of Peeta *et al.* (2010) by allowing retrofitting decisions to be represented by a continuous variable between zero and one. Ramirez-Marquez *et al.* (2009) also included multi-level protection in a network protection study. They employed a sampling-based evolutionary algorithm to compute solutions. Losada *et al.* (2012) also included multi-level protection in a facility interdiction problem where multiple resource units can be allocated to a facility. Zhu *et al.* (2013) added a protection layer to the model by Losada *et al.* (2012). Specifically, the probability that a particular facility fails is a function of the amount of resources allocated by the protector and the interdictor.

Assumption 2 Peeta *et al.* (2010) modeled imperfect protection for a problem in which a decision-maker invests resources to retrofit unreliable road segments in a highway network in order to minimize the expected post-failure transportation cost. A segment's failure probability is low if it is retrofitted and high if it is not.

The present paper is the to first to relax Assumption 1 (single-level protection), Assumption 2 (perfect protection), and Assumption 3 (binary-state capacity) in combination. A benefit of relaxing these three in combination is that is allows one to analyze the interaction between these assumptions.

When modeling the failure of capacitated facilities or network elements, a scenario-based stochastic programming formulation is usually the best option (**?**Snyder *et al.*, 2006; **?**; Peng *et al.*, 2011; Peeta *et al.*, 2010). However, using stochastic programming is problematic if imperfect protection is modeled because imperfect protection implies that the resource allocation decision affects the *probability* of the scenarios. This is in contrast with typical stochastic programming models in which the probability distribution governing the scenarios is known (Birge and Louveaux, 2011). However, there is a small body of research on stochastic programming with decision-dependent uncertainty (DDU), in which uncertainty depends on the decision variables. Adding DDU to a stochastic program makes it significantly more difficult to solve (Jonsbraaten *et al.*, 1998; Shapiro *et al.*, 2009), partly because the resulting program is usually non-convex (Shapiro *et al.*, 2009). Despite its difficulties, there has been some research on stochastic programming with DDU. According to Goel and Grossmann (2006), research on DDU can be classified according to the effect of the first-stage decisions: (1) first-stage decisions affect the information that the decision-maker has about the uncertainty (Jonsbraaten *et al.*, 1998; Goel and Grossmann, 2006) and (2) first-stage decisions affect the actual probabilities. Only a few have studied category (2), which is the category studied in this paper.

Peeta *et al.* (2010) and Du and Peeta (2014), who studied a network protection problem as a stochastic program with DDU of category (2), are the papers most closely related to ours in terms of the methodology used. In Peeta *et al.* (2010) the objective function was approximated by a Taylor series expansion, which resolved the non-convexity caused by DDU. This approximate model is then solved using sample average approximation (Kleywegt *et al.*, 2001). Because they approximate the objective function, their approach does not guarantee optimality or provide a performance guarantee of their problem. Du and Peeta (2014) use an iterative heuristic algorithm in conjunction with Monte Carlo simulation. Our work differs from Peeta *et al.* (2010) and Du and Peeta (2014) in that, rather than approximating the objective function or using a heuristic approach, we provide an approximate algorithm with a performance guarantee. In addition, we also present an exact solution approach, based on a reformulation that resolves the non-convexity caused by the DDU. Finally, they do not consider multiple levels of capacity as we do in this paper.

1.3 Contributions

The contributions of this paper are as follows: (1) We demonstrate that the PFPP can be modeled as a two-stage stochastic program with decision-depend uncertainty (DDU). (2) We show that PFPP is submodular under certain assumptions. (3) We propose a simple greedy algorithm that, due to the submodularity property, has a worst case bound of $1 - e^{-1} \approx 0.63$. (4) We describe a reformulation that resolves the non-convexity of the DDU-formulation, resulting in a mixed-integer linear program.

(5) We report the results of computational experiments in which the greedy algorithm performed much better than the worst case bound. (6) We report findings that indicate that the mean-value version of the problem produces near-optimal solutions. (7) We also use our model to test the effect that Assumptions 1–3 have on solution quality.

2 Problem Description and Model

In the *probabilistic facility protection problem* (PFPP) a set of spatially-located facilities, which are exposed to hazards, must together serve a set spatially-located demand points despite the fact that hazards may degrade the capacity of facilities to serve demand points. To mitigate against the risk of hazards, a planner allocates protection resources to facilities, constrained by a fixed budget. Although the planner does not know the severity or locations of the disruptions, he does have complete information about their probabilities.

The hazards are due to extreme weather events, which are governed by some known distribution. After a hazard occurs, a facility's post-hazard capacity state is a probabilistic function of 1) the severity of the hazard and 2) the level of protection at the facility. Other researchers have cited that modeling stochasticity in hazard outcomes as well as post-hazard capacities is important (Du and Peeta, 2014, Section 1).

After each facility's post-hazard capacity state is known, the set of facilities serve the set of demand points. We consider a disaster response context in which the facilities represent distribution centers containing prepositioned goods; in the event of a disaster these goods are transported to the disaster site. Because of the urgent nature of disaster response, the delivery time is critical. Thus, when a disruption causes the capacity of facilities to degrade, recipients have to wait longer to receive relief goods because they may have to come from a farther location.

2.1 Mathematical Model

We model the PFPP as a two-stage stochastic program with decision-dependent uncertainty. Let $\mathcal{J} = \{1, \dots, J\}$ be a set of facility indices, and let first-stage variables $\boldsymbol{y} = (y_j)_{j \in \mathcal{J}} \in \mathcal{Y} \subseteq \mathbb{R}^J$ represent the amount of protection allocated to each facility $j \in \mathcal{J}$. The budget for protection is denoted as b. The planner has perfect information about the efficacy of his or her protection allocation decisions, knowing the functional relationship between the amount of protection allocated to a facility and the probability distribution for that facility's post-disruption capacity state.

2.1.1 Uncertainty

The random hazards are governed by a random vector $\tilde{\boldsymbol{\psi}} = (\tilde{\psi}_j)_{j\in\mathcal{J}}$, whose elements describe the hazard intensity level that each facility j is exposed to. For example, if the hazard is a hurricane, facilities near the center of the storm will be exposed to stronger winds than those further away. The post-hazard capacity state for each facility is governed by the independent random vector, $\tilde{\boldsymbol{a}} = (\tilde{a_j})_{j\in\mathcal{J}}$, and the element corresponding to a facility j, $\tilde{a_j}$, is conditional upon the hazard intensity that j is exposed to, $\tilde{\psi_j}$, and the protection allocated to j, y_j. Thus, $g_j(a_j|\psi_j; y_j)$ is the conditional probability function of the hazard level random variable for facility j, and

$$g_j(a_j|y_j) = \mathbb{E}_{\tilde{\psi_j}} g_j(a_j|\psi_j; y_j) \tag{1}$$

is the probability function.

Disruptions are represented by a finite set of scenarios, Ω, with scenario $\omega \in \Omega$ defining the capacity level for each facility j, a_j^ω, as well as the hazard level for each facility, ψ_j^ω. The probability for scenario ω is

$$\mathbb{P}^\omega(\boldsymbol{y}) = \prod_{j\in\mathcal{J}} g_j(a_j^\omega|\psi_j^\omega; y_j). \tag{2}$$

2.1.2 Second Stage

The second stage of the model is a transportation problem, seeking to transport relief goods to a disaster site while providing maximum utility to the recipients. The utility that victims receive from a shipment of goods from facility j to site i is $u_{ij}(t_{ij}(d_{ij}))$, where $t_{ij}(\cdot)$ converts distance to travel time, and d_{ij} is the distance from i to j. For brevity, we abbreviate the utility as u_{ij}. Each site i has a demand of e_i units. When the total available facility capacity after a disruption is insufficient to satisfy total customer demand, a dummy facility, indexed $J + 1$, may satisfy demand at a much lower utility, $u_{i,J+1} = u_{ij}(t_{ij}(d' \max_{ij}\{d_{ij}\}))$, where d' is an arbitrary penalty multiplier. A linear programming formulation of the transportation problem for scenario ω is

$$
\begin{aligned}
h(\boldsymbol{a}^\omega) = \quad \max. \quad & \sum_{i\in\mathcal{I}}\sum_{j=1}^{J+1} e_i u_{ij} x_{ij} && \text{(3a)}\\
\text{s.t.} \quad & \sum_{i\in\mathcal{I}} e_i x_{ij} \le a_j^\omega \quad \forall j \in \mathcal{J}, && \text{(3b)}\\
& \sum_{j=1}^{J+1} x_{ij} = 1 \quad \forall i \in \mathcal{I}, && \text{(3c)}\\
& x_{ij} \ge 0 \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. && \text{(3d)}
\end{aligned}
$$

The objective of model (3) is to maximize the total weighted utility (3a). The constraints ensure that: total customer demand allocated to a facility does not exceed that facility's capacity (3b), all of the demand is satisfied for each demand point (3c), and the assignment variables are non-negative (3d).

An important property of this second-stage problem is that it does not depend on the first-stage decisions; thus, $h^\omega = h(\boldsymbol{a}^\omega)$ is a model parameter.

2.1.3 Discrete Equivalent Problem

Thus, the two-stage problem is:

$$\max_{\boldsymbol{y}\in\mathcal{Y}} \mathcal{Q}(\boldsymbol{y}) = \sum_{\omega\in\Omega} \mathbb{P}^\omega(\boldsymbol{y})\, h(\boldsymbol{a}^\omega). \tag{4}$$

3 Model Properties

As we will show, although (4) is, in general, non-convex, it is submodular under certain assumptions.

3.1 Concavity of Single-Facility Case, $\mathcal{Q}(y_1)$

As we will show, for the single facility case, $\mathcal{Q}(y_1)$ is concave if the facility capacity random variable has a binomial distribution and $h(\cdot)$ is submodular.

Definition 1. Let $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{h}$ be vectors in $\mathbb{R}$ such that $\mathbf{x} \le \mathbf{y}$ and $\mathbf{h} \ge 0$. A function $f(\cdot)$ is submodular if $f(\mathbf{x}+\mathbf{h}) - f(\mathbf{x}) \ge f(\mathbf{y}+\mathbf{h}) - f(\mathbf{y})$, where $\mathbf{x}+\mathbf{h}$ denotes element-wise addition

(Nemhauser *et al.*, 1978).

Let the probability mass function for a binomial random variable with n trials and k successes, and a success probability of p be $b_{k,n}(p) = \binom{n}{k} p^k (1-p)^{n-k}$.

First, we will establish that the following expectation is concave on $[0,1]$ if $h(k)$ is submodular:

$$\mathcal{Q}(p) = \sum_{k=0}^{n} h(k) b_{k,n}(p). \tag{5}$$

Shaked (1980) showed that an exponential family probability distribution is convexly parameterized, meaning that the expectation of a convex function over an exponential family distribution is convex in the parameter of the distribution. Rather than modify this result to show that $\mathcal{Q}(p)$ is concave when $h(k)$ is submodular, we provide a proof specific to the binomial distribution. (Noah Stein at Analog Devices Lyric Labs proved this result for the case in which $h(k)$ is convex. Here we present a modified version for the concave and submodular case.)

Lemma 1. *$\mathcal{Q}(p)$ is concave on $[0,1]$ if $h(k)$ is submodular.*

Proof. See Appendix A □

Corollary 1. *Let $f(\cdot) : [0,b] \to [0,1]$ be a non-decreasing concave function. $\mathcal{Q}(y_1)$ is concave if the range of $\tilde{a}_1$ is $\{0,1,\ldots,\bar{a}_1\}$, $g_1(a_1|\tilde{\psi}_1; y_1)$ is binomially distributed with $\bar{a}_1$ trials and success probability $p = f(y_1)$, and $h(a_1)$ is submodular.*

Proof. See Appendix A □

3.2 Submodularity of General Case, $\mathcal{Q}(\boldsymbol{y})$

First, we will establish that the following expectation is concave on $[0,1]^J$ if $h(k)$ is submodular:

$$\mathcal{Q}(\mathbf{p}) = \sum_{k_1=0}^{n} \cdots \sum_{k_m=0}^{n} h(\mathbf{k}) \prod_{j \in \mathcal{J}} b_{k,n}(p_j),$$

where $\mathbf{p} = (p_1, \ldots, p_m)$ is a vector in $[0,1]^m$ and $\mathbf{k} = (k_1, \ldots, k_m)$. This result is known for the case in which the random variables are Bernoulli (Vondrak, 2008).

Theorem 1. *$\mathcal{Q}(\mathbf{p})$ is submodular if and only if $h(\mathbf{k})$ is submodular.*

Proof. See Appendix A □

Corollary 2. *Let $f_j(\cdot) : [0, b] \to [0, 1]$ be a non-decreasing concave function for all $j \in \mathcal{J}$. $\mathcal{Q}(\boldsymbol{y})$ is concave if, for all $j \in \mathcal{J}$, the range of $\tilde{a}_j$ is $\{0, 1, \ldots, \bar{a}_j\}$, $g_j(a_j|\tilde{\psi}_j; y_j)$ is binomially distributed with $\bar{a}_j$ trials and success probability $p = f(y_j)$, and $h(\boldsymbol{a})$ is submodular.*

Proof. See Appendix A □

Remark 1. Our second stage problem 3 is submodular because it is a linear program. However, many integer programs also have this property (Nemhauser *et al.*, 1978).

As the sequel will show, the submodularity of $\mathcal{Q}(\boldsymbol{y})$ implies a constant-factor guarantee for a greedy algorithm.

4 Solution Methods

4.1 Greedy Algorithm

This section describes a greedy algorithm for PFPP. This greedy algorithm is similar to the one used by Vondrak (2008) to obtain a relaxation solution to the deterministic submodular maximization problem.

To obtain a constant-factor performance guarantee of the greedy algorithm we make the following observations and assumptions. First, $\mathcal{Q}(\boldsymbol{y})$ is monotone, which is straightforward to show. Second, the set of feasible $\boldsymbol{y}$ forms a uniform matroid. To do this, we discretize the values of y_j into K levels. Thus, the feasible region is

$$\mathcal{Y} = \{\boldsymbol{y} : \sum_{j \in \mathcal{J}} y_j \leq b, \boldsymbol{y} \geq \boldsymbol{0}\}. \tag{6}$$

The greedy algorithm (shown in Algorithm 1) iteratively adds protection resources to the facilities with the largest marginal benefit until b resources have been allocated. The main loop of the algorithm (lines 8 to 10) finds the set of the best b/K facilities according to marginal benefit and adds 1 unit of protection to each of these facilities, continuing until the budget is exhausted. Note that the algorithm requires that for a given K the budget (b) should be chosen so that b

mod $K = 0$. If this is not the case, then b' (Step 6 of the algorithm) must be rounded down, causing the algorithm to not use all of the budget.

Algorithm 1 Greedy algorithm for imperfect facility protection.

```
1  function GREEDY
2      Let $\mathbf{e}_j$ be a vector with a 1 at position $j$ and 0s elsewhere.
3      Let $\mathbf{1}_X$ denote a zero vector with 1s in the positions of the elements of the set $X$.
4      Let $X[k]$ denote the set of the $k$ largest values of $X$.
5      Given solution $\boldsymbol{y}$, the marginal benefit for facility $j$ is
           $\Delta_j \mathcal{Q}(\boldsymbol{y}) = \sum_{\omega \in \Omega} \mathbb{P}^{\omega}(\boldsymbol{y}) \left[h(\boldsymbol{a}^{\omega} + \mathbf{e}_j) - h(\boldsymbol{a}^{\omega})\right]$
6      Set $b' = b/K$. Start with $t = 0$, and $\boldsymbol{y}(0) = \mathbf{0}$.
7      while $t < K$ do
8          Set $I(t) = \{\Delta_j \mathcal{Q}(\boldsymbol{y}) : j \in \mathcal{J}\}[b']$
9          $\boldsymbol{y}(t+1) = \boldsymbol{y}(t) + \mathbf{1}_{I(t)}$
10         $t = t + 1$
11     return $\boldsymbol{y}(K)$
```

For a given problem instance, let $\boldsymbol{y}^G$ be the solution returned by the greedy algorithm (Algorithm 1), and let $\boldsymbol{y}^{OPT}$ be the optimal solution. Relying on the fact that $\mathcal{Q}(\boldsymbol{y})$ is submodular (corollary 2), the following theorem gives a constant-factor performance guarantee for Algorithm 1.

Corollary 3. $\mathcal{Q}(\boldsymbol{y}^G) \geq (1 - \frac{1}{e})\mathcal{Q}(\boldsymbol{y}^{OPT})$

Proof. See Appendix A □

Remark 2. Studying a set covering problem, Wolsey (1982) presented a continuous version of Algorithm 1, which is also valid for our problem. We presented a discrete version here so that we can easily compare it with the mixed-integer stochastic program that we present in the next subsection.

4.2 Two-Stage Stochastic Programming Model

As a complement to the greedy approximation algorithm, we describe an exact procedure in this section. Because our natural formulation of the PFPP as a two-stage stochastic program with decision-dependent uncertainty (DDU) (4) is difficult to solve directly (Jonsbraaten *et al.*, 1998; Shapiro *et al.*, 2009), a series of reformulations are presented, eventually leading to a mixed-integer linear stochastic program.

In this section we first formulate our problem as a non-convex two-stage stochastic program (§4.2.1) and then reformulate it as a conventional two-stage stochastic mixed-integer linear program (§4.2.2), relying on the assumption that the set of facility capacity conditional random variables $\{\tilde{a}_j|\tilde{\boldsymbol{\psi}}\}_{j\in\mathcal{J}}$ are mutually independent. That is, given a particular hazard outcome, the capacity states of the facilities are independent. This is not a restrictive assumption because it still allows the hazard levels for each facility to be correlated. This assumption facilitates a "probability chain" modeling framework (Morton *et al.*, 2007; Losada *et al.*, 2012; O'Hanley *et al.*, 2013) that we use to linearize the model.

4.2.1 Non-Convex Formulation

First, we discretize the random vectors. Redefine $\tilde{\boldsymbol{\psi}} = (\tilde{\psi}_{jm})_{j\in\mathcal{J},m\in\mathcal{M}}$ as the random hazard vector, whose elements equal 1 if facility j is exposed to hazard level m and 0 otherwise. The binary random vector $\tilde{\boldsymbol{\xi}} = (\tilde{\xi}_{j\ell})_{j\in\mathcal{J},\ell\in\mathcal{L}}$, is a vector of facility capacity states where $\tilde{\xi}_{j\ell} = 1$ if facility j is in capacity state ℓ after the disruptive event and 0 otherwise. Thus, given the realization $\boldsymbol{\xi}$, the amount of capacity for facility j is $\sum_{\ell\in\mathcal{L}} a_{j\ell}\xi_{j\ell}$, where $a_{j\ell}$ is the amount of capacity available at facility j when j is in state ℓ. Thus, the post-disruption transportation problem is represented as

$$
\begin{aligned}
h(\boldsymbol{\xi}) = \quad \max. \quad & \sum_{i\in\mathcal{I}}\sum_{j=1}^{J+1} e_i d_{ij} x_{ij} && \text{(7a)}\\
\text{s.t.} \quad & \sum_{i\in\mathcal{I}} e_i x_{ij} \le \sum_{\ell\in\mathcal{L}} a_{j\ell}\xi_{j\ell} \quad \forall j\in\mathcal{J}, && \text{(7b)}\\
& (3c)\text{–}(3d).
\end{aligned}
$$

Next, we discretize the allocation decision variables. Let the set $\mathcal{K} := \{0, 1, \ldots, K-1\}$ be a set of protection allocation levels, indexed by k, and the set $\mathcal{L} := \{0, 1, \ldots, L-1\}$ be a set of facility capacity levels indexed by ℓ. Redefine $\boldsymbol{y} = (y_{jk})_{j\in\mathcal{J},k\in\mathcal{K}}$ as a vector of binary variables, where $y_{jk} = 1$ if k resources are allocated to facility j and 0 otherwise. With this redefinition, PFPP can be formulated as the following two-stage stochastic program:

$$
\begin{aligned}
\max. \quad & \mathbb{E}_{\tilde{\boldsymbol{\psi}}}\left[\mathbb{E}_{\tilde{\boldsymbol{\xi}}|\tilde{\boldsymbol{\psi}},\boldsymbol{y}}[h(\tilde{\boldsymbol{\xi}})]\right] && \text{(8a)}\\
\text{s.t.} \quad & \sum_{k\in\mathcal{K}} y_{jk} = 1 \quad \forall j \in \mathcal{J}, && \text{(8b)}\\
& \sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}} k y_{jk} \le b, && \text{(8c)}\\
& y_{jk} \in \{0,1\} \quad \forall j \in \mathcal{J}, k \in \mathcal{K}. && \text{(8d)}
\end{aligned}
$$

Recall that the probability of a scenario ω is $\mathbb{P}^{\omega}(\boldsymbol{y})$, which is the likelihood of the realizations $\boldsymbol{\xi}^{\omega}$ and $\boldsymbol{\psi}^{\omega}$ given allocation vector $\boldsymbol{y}$. Assuming that facilities fail independently given a hazard level outcome,

$$
\mathbb{P}^{\omega}(\boldsymbol{y}) = \prod_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\sum_{\ell\in\mathcal{L}}\sum_{m\in\mathcal{M}} \mathbb{P}_{j\ell mk}\, \xi^{\omega}_{j\ell}\psi^{\omega}_{jm}\, y_{jk}, \tag{9}
$$

where $\mathbb{P}_{j\ell mk} = \mathbb{P}(\xi_{j\ell} = 1 \,|\, \tilde{\psi}_{jm} = 1\,;\, y_{jk} = 1)$ is the probability that element j is in capacity state ℓ when exposed to hazard level m given that k protection resources were allocated to j. For ease of notation, denote the probability that facility j is in capacity state ℓ in scenario ω as

$$
\mathbb{P}^{\omega}_{j\ell k} = \sum_{m\in\mathcal{M}} \mathbb{P}_{j\ell mk}\, \xi^{\omega}_{j\ell}\psi^{\omega}_{jm}. \tag{10}
$$

Hence, (8) can be expressed as the following stochastic programming extensive form:

$$
\begin{aligned}
\max. \quad & \sum_{\omega\in\Omega}\prod_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}}\left(\sum_{\ell\in\mathcal{L}}\mathbb{P}^{\omega}_{j\ell k}\right) y_{jk}\sum_{i\in\mathcal{I}}\sum_{j=1}^{J+1} e_i d_{ij} x^{\omega}_{ij} && \text{(11a)}\\
\text{s.t.} \quad & \sum_{i\in\mathcal{I}} e_i x^{\omega}_{ij} \le \sum_{\ell\in\mathcal{L}} a_{j\ell}\xi^{\omega}_{j\ell} \quad \forall j \in \mathcal{J}, \omega \in \Omega, && \text{(11b)}\\
& \sum_{j=1}^{J+1} x^{\omega}_{ij} = 1 \quad \forall i \in \mathcal{I}, \omega \in \Omega, && \text{(11c)}\\
& x^{\omega}_{ij} \ge 0 \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \omega \in \Omega, && \text{(11d)}\\
& (8b)\text{–}(8d).
\end{aligned}
$$

Because the objective function (11a) contains product terms involving decision variables, this formulation is a mixed-integer non-convex stochastic program.

4.2.2 Discretized Reformulation

Because the second-stage problems do not depend on the first-stage decisions, $\boldsymbol{y}$, we can represent the optimal second-stage objective value in scenario ω as the fixed value h^ω and reformulate (11) as

$$\begin{aligned} \max. \quad & \sum_{\omega \in \Omega} h^\omega \prod_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} \left(\sum_{\ell \in \mathcal{L}} \mathbb{P}^\omega_{j\ell k} \right) y_{jk} && \text{(12a)} \\ \text{s.t.} \quad & (8b)\text{–}(8d). \end{aligned}$$

In Formulation (12), the expected cost is calculated in the objective function (12a) by adding the probability-weighted transportation costs (i.e., $h^\omega \prod_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \sum_{\ell\in\mathcal{L}} \mathbb{P}^\omega_{j\ell k} y_{jk}$) over all scenarios; this is how the expected cost is typically calculated in stochastic programs. However, to remove the remaining non-linearity in (12), we use a reformulation that calculates the probability-weighted transportation cost for each scenario using recursive equations, which removes the product $\prod_{j\in\mathcal{J}}$.

By the independent facility failures assumption, we substitute the product

$$\prod_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} \sum_{\ell \in \mathcal{L}} \mathbb{P}^\omega_{j\ell k} y_{jk}$$

with a recursive expression by using bookkeeping variables to calculate the product of probabilities. Let z^ω be a variable that holds the probability-weighted transportation cost for scenario ω. Let $w_{rk\omega}$ be a bookkeeping variable that holds value of h^ω multiplied by the likelihood that facilities $1, \ldots, r-1$ are in their corresponding capacity states defined by scenario ω if k units have been allocated to facility r and zero otherwise. Thus, we want each variable $w_{rk\omega}$ to hold the value

$$h^\omega \prod_{j=1}^{r-1} \sum_{k \in \mathcal{K}} \sum_{\ell \in \mathcal{L}} \mathbb{P}^\omega_{r,\ell,k} y_{jk}.$$

To ensure that the value of $w_{rk\omega}$ is computed correctly, we use the following recursive equations

$$h^{\omega} = \sum_{k \in \mathcal{K}} w_{1k\omega} \quad \forall \omega \in \Omega, \tag{13}$$

$$\sum_{k \in \mathcal{K}} \sum_{\ell \in \mathcal{L}} \mathbb{P}^{\omega}_{r-1,\ell,k} w_{r-1,k\omega} = w_{rk\omega} \quad \forall r = 2, \ldots, J; \omega \in \Omega, \tag{14}$$

$$\sum_{k \in \mathcal{K}} \sum_{\ell \in \mathcal{L}} \mathbb{P}^{\omega}_{J\ell k} w_{Jk\omega} = z^{\omega}, \tag{15}$$

which ensure that z^{ω} is equal to the transportation cost for scenario ω (h^{ω}) times probability of scenario ω (the product of the probabilities that each facility realizes its capacity state) along with the constraints

$$w_{rk\omega} \leq y_{rk} \quad \forall r = 1, \ldots, J; k \in \mathcal{K}; \omega \in \Omega, \tag{16}$$

which ensure that $w_{rk\omega}$ is positive only if k units are allocated to the r^{th} facility. These recursive equations make our reformulation similar to the network-flow type reformulation employed in Morton *et al.* (2007) (see also Losada *et al.*, 2012). O'Hanley *et al.* (2013) called these recursive equations a "probability chain." Figure 1 illustrates the probability chain flow network for a scenario ω of a problem instance consisting of three facilities. The flow into node 0 is the optimal total weighted utility for scenario ω as computed by (7). The flow variables ($w_{rk\omega}$) are shown above the arcs and the arc weights are shown below. When flow travels through an arc the flow is multiplied by the arc weight, which is equal to the state probability for the corresponding facility and allocation level. Flow travels through the network, being multiplied by the weight of each arc and maintaining balance of flow at each node. At each node r the flow is multiplied by the probability that facility r is in capacity state ℓ given that $y_{rk} = 1$. Thus, after visiting the last node, the flow is equal to $h^{\omega}\mathbb{P}^{\omega}(\boldsymbol{y})$, the total transportation utility multiplied by the probability of scenario ω.

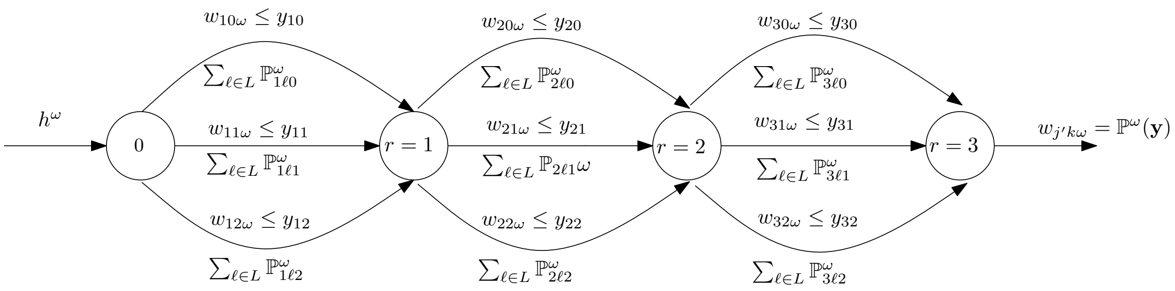

h^ω
0
$r = 1$
$r = 2$
$r = 3$
$w_{10\omega} \leq y_{10}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{1\ell 0}$
$w_{11\omega} \leq y_{11}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{1\ell 1}$
$w_{12\omega} \leq y_{12}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{1\ell 2}$
$w_{20\omega} \leq y_{20}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{2\ell 0}$
$w_{21\omega} \leq y_{21}$
$\sum_{\ell \in L} \mathbb{P}_{2\ell 1}\omega$
$w_{22\omega} \leq y_{22}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{2\ell 2}$
$w_{30\omega} \leq y_{30}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{3\ell 0}$
$w_{31\omega} \leq y_{31}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{3\ell 1}$
$w_{32\omega} \leq y_{32}$
$\sum_{\ell \in L} \mathbb{P}^\omega_{3\ell 2}$
$w_{j'k\omega} = \mathbb{P}^\omega(\mathbf{y})$

Figure 1: Probability chain flow network for scenario ω with three facilities (adapted from Losada *et al.* (2012))

The PFPP can now be reformulated as a mixed-integer linear program:

$$\max. \quad \sum_{\omega \in \Omega} \sum_{k \in \mathcal{K}} w_{Jk\omega} \tag{17a}$$

$$\text{s.t.} \quad h^\omega = \sum_{k \in \mathcal{K}} w_{1k\omega} \quad \forall \omega \in \Omega, \tag{17b}$$

$$\sum_{k \in \mathcal{K}} \sum_{\ell \in \mathcal{L}} \mathbb{P}^\omega_{r,\ell,k} w_{r-1,k\omega} = \sum_{k \in \mathcal{K}} w_{rk\omega} \quad \forall r = 2, \ldots, J; \omega \in \Omega, \tag{17c}$$

$$w_{rk\omega} \leq h^\omega y_{rk} \quad \forall r = 1, \ldots, J; k \in \mathcal{K}; \omega \in \Omega, \tag{17d}$$

$$w_{rk\omega} \geq 0 \quad \forall r = 1, \ldots, J; k \in \mathcal{K}; \omega \in \Omega, \tag{17e}$$

$$(8b)–(8d).$$

The objective (17a) is equivalent to (12a).

Remark 3. Because the values of h^ω are computed *a priori*, the second stage transportation problem does not need to be linear or even convex.

Remark 4. Given a fixed allocation vector ($\hat{\boldsymbol{y}}$) this formulation can be solved by inspection: simply calculate the probability of each scenario ω.

In a set of pilot tests, we solved formulation (17) using Gurobi, an off-the-shelf optimization software; the computation times were large for even small-sized problems (49 demand points, 5-10 facilities). Because formulation (17) has a block-diagonal structure (the constraints can be separated by ω), we employed an L-shaped decomposition strategy (Van Slyke and Wets, 1969). The following section examines the practical performance of the greedy algorithm (Algorithm 1) and the L-Shaped strategy.

5 Numerical Experiments

This section reports the results of computational experimentation with the PFPP. All experiments were run on compute nodes contained in the Arkansas High Performance Computing Cluster using a 64-bit Linux operating system. A node has 2 Xeon X5670 Intel processors, which each have 8 cores and a clock speed of 2.93GHz and share 24GB of memory.

The greedy algorithm was implemented in Python. We used the single-cut L-shaped method (Van Slyke and Wets, 1969), also implemented in Python, to solve the two-stage stochastic programming formulation (17); the master problem and sub-problems were solved using the Gurobi commercial solver, version 5.5.0. (We found that solving the subproblems by inspection and then calculating the optimal solution was no faster than solving them using Gurobi.) While we could have used enhancements to the basic L-shaped method such as multiple cuts per iteration (Birge and Louveaux, 1988) or regularization (Ruszczyński, 1986), the main purpose of using this method is to find an optimal solution in order to evaluate the performance of the greedy algorithm as well as parameter sensitivity. A one-hour time limit was used for both the greedy algorithm and the L-shaped method. The L-shaped method was terminated when the optimality gap, calculated $\frac{ub-lb}{lb}$, reached 0.001.

Datasets The 49- and 88-node facility location datasets from Daskin (1995) were used (denoted d49 and d88, respectively) with the following utility function:

$$u_{ij}(t_{ij}(d_{ij})) = 3\exp\left(-3\left(\frac{d_{ij}/100}{\max_{i,j}\{d_{ij}\}/100}\right)\right),$$

which models a convex decrease in the utility as the distance increases. The 49-node dataset includes the capitals of lower-48 states in US plus Washington, D.C., and the 88-node dataset contains the top 88 cities according to population.

To obtain a set of located facilities for a problem instance, we solved the capacitated p-median problem (Hakimi, 1964), setting each facility's capacity to $a_j = \frac{\sum_{i\in\mathcal{I}} e_i}{J(1-0.1)}$,
implying that the system is designed with 10% excess capacity.

Hazard Scenarios To model the random hazards, the scenarios in Table 1 were used along with an additional "no hazard" scenario with probability 0.70. Each hazard has a geographic center defined by a latitude and longitude. In every scenario each facility is exposed to one of three hazard intensity levels. When the a hazard is realized, all facilities within a radius of $rad1$ are exposed to a level 1 hazard; all facilities within a radius of $rad2$ are exposed to level 2. All other facilities are not affected by the hazard and thus exposed to a level 0 hazard. Although this data is artificial, a more realistic set of hazard scenarios with probabilities could be constructed based on historical events data from the NOAA Storm Events Database. This database could be used to determine the mean time between each type of storm. Then, for a user-defined planning horizon, probabilities can be obtained for each event.

Table 1: Hazard scenarios

No.	**Type**	**GeoCenter**	**lat**	**lng**	**rad1**	**rad2**	**prob.**
0	No Hazard	-	-	-	-	-	0.70
1	Earthquake	New Madrid, MO	-89.32	36.35	100	500	0.005
2	Hurricane	New York City, NY	-74	40.42	50	200	0.01
3	Hurricane	New Orleans, LA	-90.4	29.57	100	500	0.05
4	Earthquake	Portland, OR	-122.37	45.33	50	200	0.005
5	Tornado	Oklahoma City, OK	-97.29	35.29	50	200	0.05
6	Snow Storm	Minneapolis, MN	-93.13	44.59	100	500	0.08
7	Wildfire	Boulder, CO	-105.16	39.59	50	200	0.05
8	Flood	Memphis, TN	-90.2	35.8	25	100	0.05

Three hazard exposure cases were investigated in order of decreasing severity.

- *Conditional*: The objective of the model is to minimize the expected total utility conditional upon a hazard scenario occurring. Thus, the probability of hazard scenario s is $\frac{\mathbb{P}_s}{\sum_s \mathbb{P}_s}$.

- *All-Exposed*: All facilities are exposed to hazard level 1, and the objective is to minimize the expected total utility given this hazard level. This case is applicable for a context in which facilities are at high risk of a disruption; for example a military theater of operations.

- *Half-Exposed*: All facilities exposed to level 2.

Model Parameters We used a penalty multiplier of $d' = 2.0$ and set the capacity for each level as $a_{j\ell} = \frac{\ell}{L} a_j$.

Two-Stage Stochastic Programming Model The probabilities of capacity states given allocation amounts are defined by the binomial distribution so that $\mathbb{P}_{j\ell mk} = \text{binomial}(\ell, L-1, p_{jkm})$ and $p_{jkm} = \left(\frac{k}{K}\right)^{\frac{m}{M}}$. The scenario set for the three hazard exposure cases are defined as follows:

- **Conditional**: The scenario set is generated by considering all combinations of hazard levels and facility capacity states. The probabilty of a given scenario ω is the probability of its corresponding hazard scenario (see Table 1) times the conditional probability for the facility capacities state,
$$\prod_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}} \left(\sum_{\ell\in\mathcal{L}} \mathbb{P}^{\omega}_{j\ell k} \right) \tag{18}$$
- **All-Exposed**: In every scenario each facility is exposed to hazard level 1, i.e., for every scenario ω and facility j $\psi^{\omega}_{jm} = 1$ for $m = 1$ and 0 otherwise. The scenario set is generated by considering all combinations of facility capacity states. The probabilty of a given scenario ω is the conditional probability for the facility capacities state (18).

- **Half-exposed**: The scenario set is generated in the same way as the all-exposed case except that in every scenario each facility is exposed to hazard level 2.

Greedy Algorithm When computing the marginal benefit of each facility in line 5 of Algorithm 1, each scenario needs to only contain information on the capacity levels for each facility ($\boldsymbol{\xi}^{\omega}$). Thus, the set of scenarios, Ω, is all possible combinations of facility capacity level vectors. The probability of scenario ω is then computed as

$$\mathbb{P}^{\omega}(\boldsymbol{y}) = \prod_{j\in\mathcal{J}} g_j(a^{\omega}_j | y_j),$$

where $g_j(a^{\omega}_j | y_j)$ is computed as in (1).

5.1 Runtime and Solution Quality of Algorithms

We measured the solution quality of the greedy algorithm (Algorithm 1) by comparing the performance of its solution, $\boldsymbol{y}^G$, with that of the incumbent solution returned by the L-Shaped method after one hour, $\boldsymbol{y}^{lb}$. Table 2 shows the ratio $\mathcal{Q}(\boldsymbol{y}^G)/\mathcal{Q}(\boldsymbol{y}^{lb})$ for several problem instances. The budget (b) values were obtained via the equation $J \times K \times B$, with B, the budget multiplier, equaling 1/3 and 2/3. Section 1 in the Supplemental Material shows the number of scenarios for each instance. The presence of "mem." in a cell within the table indicates that the L-Shaped algorithm ran out of memory due to problem size; all row statistics do not include these instances. The "Gap" column lists the optimality gap of the L-Shaped algorithm at termination.

Not surprisingly, the run time of the greedy algorithm is always less than that of SP. However, the table also shows that the greedy algorithm always obtained a solution that is very near the optimal solution, as indicated by a very high $\mathcal{Q}(\boldsymbol{y}^G)/\mathcal{Q}(\boldsymbol{y}^{lb})$ ratio. While corollary 3 guarantees a $\mathcal{Q}(\boldsymbol{y}^G)/\mathcal{Q}(\boldsymbol{y}^{lb})$ ratio of $1 - e^{-1} \approx 0.63$, for these problem instances the ratio was never less than 0.954. In fact, for several instances this ratio is more than 1.0, which is because the L-Shaped method was not able to find the optimal solution within one hour. In this case, $\mathcal{Q}(\boldsymbol{y}^{lb})$ is the lower bound returned by the L-Shaped method. The greedy algorithm performed better as the intensity of the hazard exposure cases decreased from "All-Exposed" to "Conditional."

5.2 Value of the Stochastic Solution (VSS)

We also analyzed the performance of solutions obtained by the *mean value problem* (MVP), a deterministic model that replaces the random parameters in PFPP with their mean values (see Appendix B for a formulation). Let $\boldsymbol{y}^{MVP}$ and EV be the optimal solution and optimal objective value to **MVP**. The *value of the stochastic solution ratio* (see Birge and Louveaux (2011)) is defined as $VSSR = \mathcal{Q}(\boldsymbol{y}^{MVP})/\mathcal{Q}(\boldsymbol{y}^{lb})$. In addition, we also measured the *upper bound ratio* $UBR = EV/\mathcal{Q}(\boldsymbol{y}^{lb})$, which is a measure of the quality of the upper bound provided by **MVP**.

Table 3 contains values of $VSSR$ and UBR for several problem instances. For each hazard exposure case (e.g., "All-Exposed"), three quantities are listed: the optimality gap returned by the L-Shaped method after one hour, $VSSR$, and UBR. As the table shows, the solution quality of the **MVP** solution is very good, as indicated by the high $VSSR$ values. In fact, $VSSR > 1.0$ for some instances because the L-Shaped method did not find an optimal solution within one hour.

The quality of the upper bound provided by **MVP** is also generally good, as indicated by the upper bound ratio ranging from 1.014 to 1.354, with an average of 1.06.

We performed a Welch two-sample T-test on the values of the $\mathcal{Q}(\boldsymbol{y}^G)/\mathcal{Q}(\boldsymbol{y}^{lb})$ columns in Table 2 and the $VSSR$ columns in Table 3. The difference was insignificant for the All-Exposed and Half-Exposed cases, with P-Values of 0.701 and 0.981, respectively. However, the difference was significant for the Conditional case (P=0.025), with the Greedy algorithm performing statistically better. However, the difference in means was only 0.004.

For all of the instances shown in Table 3, the mean value model finish in less than one hundreth of a second, which is significantly faster than the greedy algorithm.

5.3 Significance of Modeling Imperfect, Multi-level Protection and Multiple Capacity Levels

As mentioned in the introduction, most infrastructure protection models only include two levels of allocation ($K = 2$) and two capacity states ($L = 2$). In this section we examine the effect of these modeling assumptions on solution quality.

Let $\hat{\boldsymbol{y}}_{KL}$ be the optimal solution to an instance of the two-stage SP model that models K allocation levels and L capacity states. To measure the sensitivity of the PFPP to changes in K and L, let $\tilde{K}$ and $\tilde{L}$ be the "true" values of the number of allocation levels and the number of capacity levels, respectively. (For example, to solve an instance of the PFPP with 4 possible allocation levels ($\tilde{K} = 4$) and 3 possible capacity levels ($\tilde{L} = 3$), we might use the two-stage SP model (17) as an approximation to the true problem, selecting $K = 2$ and $L = 2$.) Let $\boldsymbol{y}^*_{\tilde{K}\tilde{L}}$ be the solution that we would obtain if we solved the "true" problem with $\tilde{K}$ allocation levels and $\tilde{L}$ capacity levels. This solution can be obtained via the two-stage SP model with $\tilde{K}$ allocation levels and $\tilde{L}$ capacity levels. The function $\mathcal{Q}_{\tilde{\mathcal{L}}}(\cdot)$ computes the total weighted utility for a given solution applied to a "true" problem instance with $\tilde{L}$ capacity levels. However, when applying a solution obtained by solving a model with K allocation levels and L capacity levels, $\hat{\boldsymbol{y}}_{KL}$, to a "true" problem instance with $\tilde{K}$ allocation levels and $\tilde{L}$ capacity levels, the following translation must be used in order to ensure a fair comparison:

$$\text{Translated solution} = \frac{\tilde{K}-1}{K-1}\hat{\boldsymbol{y}}_{KL}, \tag{19}$$

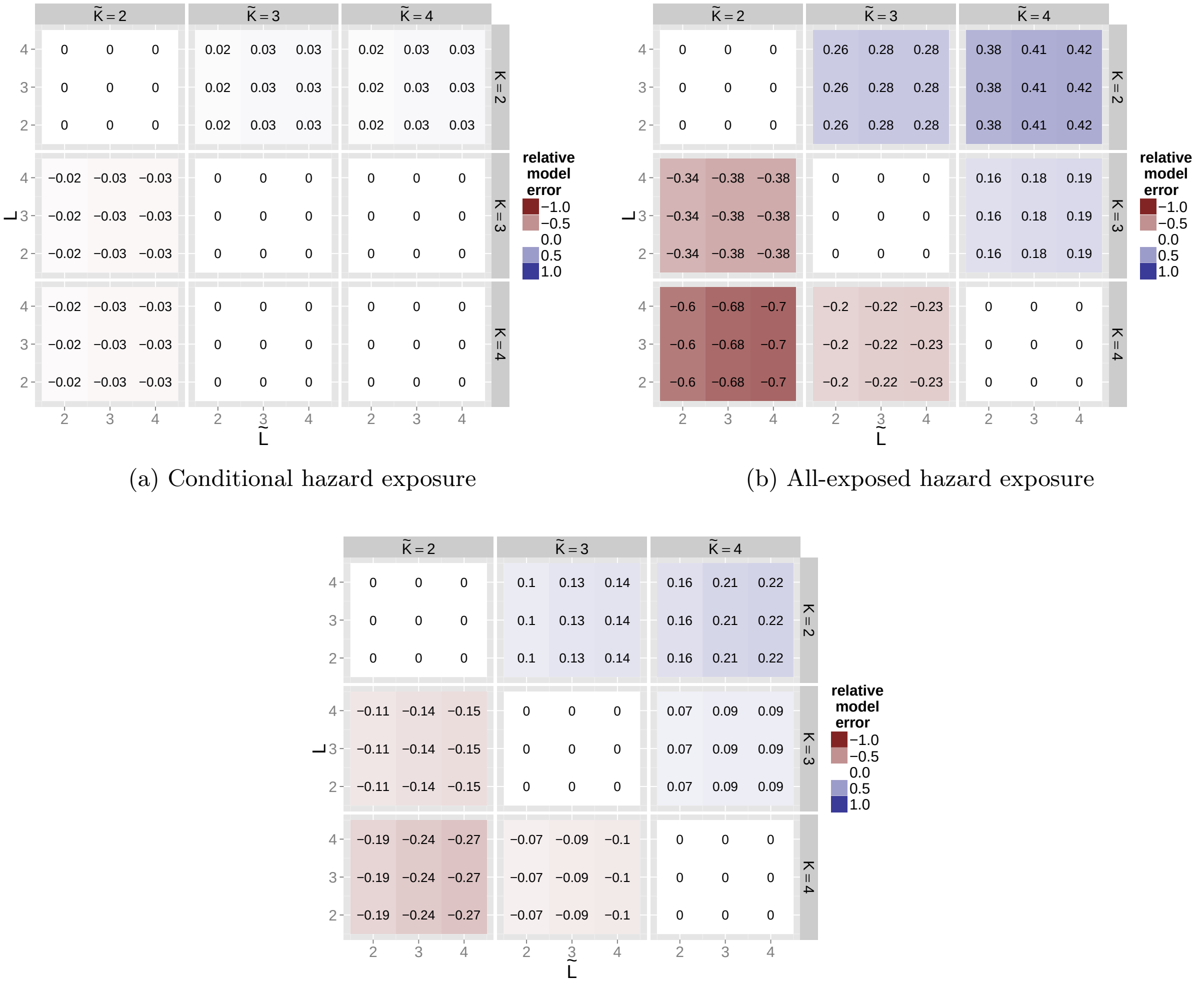
which denotes multiplying all of the elements of $\hat{\boldsymbol{y}}_{KL}$ by $\frac{\tilde{K}-1}{K-1}$.

To perform a sensitivity analysis, we computed the *relative model error,*

$$\gamma_{KL\tilde{K}\tilde{L}} = \frac{\mathcal{Q}_{\tilde{\mathcal{L}}}(\boldsymbol{y}^*_{\tilde{K}\tilde{L}}) - \mathcal{Q}_{\tilde{\mathcal{L}}}\left(\frac{\tilde{K}-1}{K-1}\hat{\boldsymbol{y}}_{KL}\right)}{\mathcal{Q}_{\tilde{\mathcal{L}}}(\boldsymbol{y}^*_{\tilde{K}\tilde{L}})}, \tag{20}$$

for several problem instances. The solutions $\hat{\boldsymbol{y}}_{KL}$ and $\boldsymbol{y}^*_{\tilde{K}\tilde{L}}$ are the incumbent solutions returned by the L-Shaped algorithm after one hour. The budget (b) values were set to $J \times K \times B$, with B, the budget multiplier, equaling 0.25 and 0.75. For each K, L combination the relative model error was computed for the three hazard exposure cases: All-exposed, Half-Exposed, and Conditional. Figures 2–5 illustrate the effect that the following model and dataset parameters have on the relative model error: hazard exposure case, number of facilities (J), budget multiplier (B), and the number of demand points (I). In addition, ANOVA tests were run to test the significance of the effect of each of these parameters (see Section 2 in the Supplemental Material for P-Values).

Figure 2 shows how the relative model error changes for the three hazard exposure cases. In each of the subfigures, each 9x9 block represents a $K, \tilde{K}$ combination. The cells within a 9x9 block contain the relative model error for different $L, \tilde{L}$ combinations. Figure 2a shows very little model error for the conditional hazard exposure case. Figure 2b shows the most hazard exposure, and Figure 2c shows a moderate amount. Figure 2 also shows that there is little change within a 9x9 block, indicating that the number of capacity levels modeled (L) does not have a significant effect on the relative model error. However, the difference between blocks is more pronounced, especially in Figures 2b and 2c, indicating that the number of allocation levels modeled (K) has a significant effect on the relative model error. When the number of allocation levels modeled (K) is equal to the actual number of allocation levels ($\tilde{K}$), then the relative model error is zero. When K is less than $\tilde{K}$ the error is positive, indicating that the model is deficient. When K is greater than $\tilde{K}$ the error is negative, indicating that the model is produces an objective value that is better than what can be achieved in reality. (The ANOVA P-Values in Section 2 in the Supplemental Material also support the significance of the number of allocation levels and the insignificance of the number of



(a) Conditional hazard exposure

(b) All-exposed hazard exposure

(c) Half-exposed hazard exposure

Figure 2: Robustness error for d49 dataset–analysis of hazard exposure case (number of facilities (J) = 4, budget (B) = 0.25)

capacity levels.) Thus, the main findings from our sensitivity analysis are as follows:

F1) **Significance of the number of allocation levels.** When the number of allocation levels modeled (K) is less than the actual number of allocation levels ($\tilde{K}$), then the relative model error is positive. When K is greater than $\tilde{K}$ the error is negative.

F2) **Insignificance of the number of capacity levels.** The number of capacity levels modeled (L) does not have a significant affect on the relative model error.

The results in the remainder of this section support these two main findings.

Figure 3 shows the effect of the number of facilities on the relative model error. The top 81

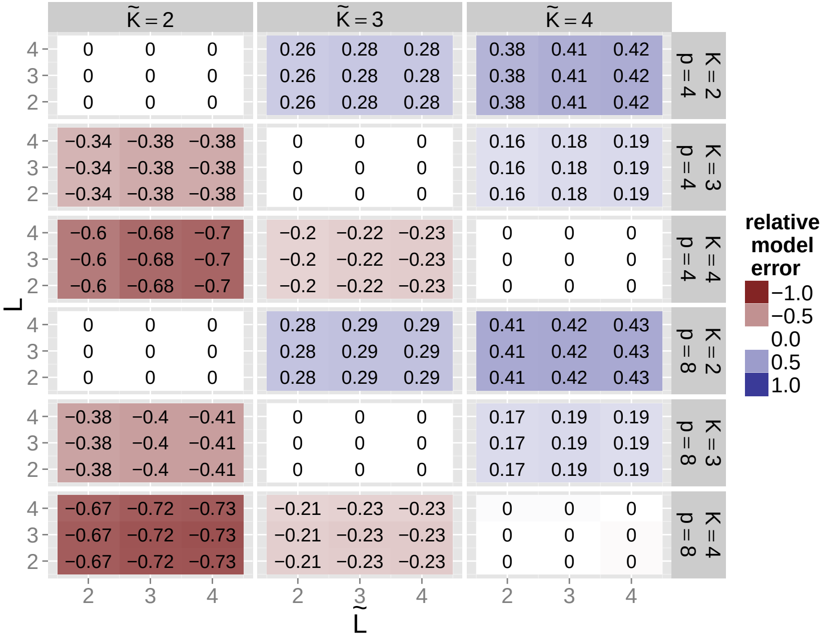


Figure 3: Robustness error for d49 dataset – analysis of number of facilities (budget (B) = 0.25, all-exposed hazard exposure)

cells show the error for 4 facilities, and the bottom 81 cells show the error for 8 facilities. The figure shows that both the top half and the bottom half show the presence of findings F1) and F2), indicating that these two findings are robust with regard to the number of facilities.

Figure 4 shows the effect of the budget on the relative model error. The top 81 cells show the error for a budget multiplier of 0.25, and the bottom 81 cells show the error for a multiplier of 0.75. The figure shows that the top half shows the presence of findings F1) and F2), while the bottom shows these findings, but with a much smaller magnitude of difference. Thus, these results indicate the following additional finding:

F3) **Signficance of the budget.** When the budget value is very high (75% of the maximum possible budget), then the number of allocation levels modeled has a less significant effect on the relative model error.

Figure 5 shows the effect of the dataset on the relative model error. The top 81 cells show the error for the d49 dataset, and the bottom 81 cells show the error the d88 dataset. The figure shows that both the top half and the bottom half show the presence of findings F1) and F2), indicating that these two findings are robust with regard to the choice between these two datasets. The dataset

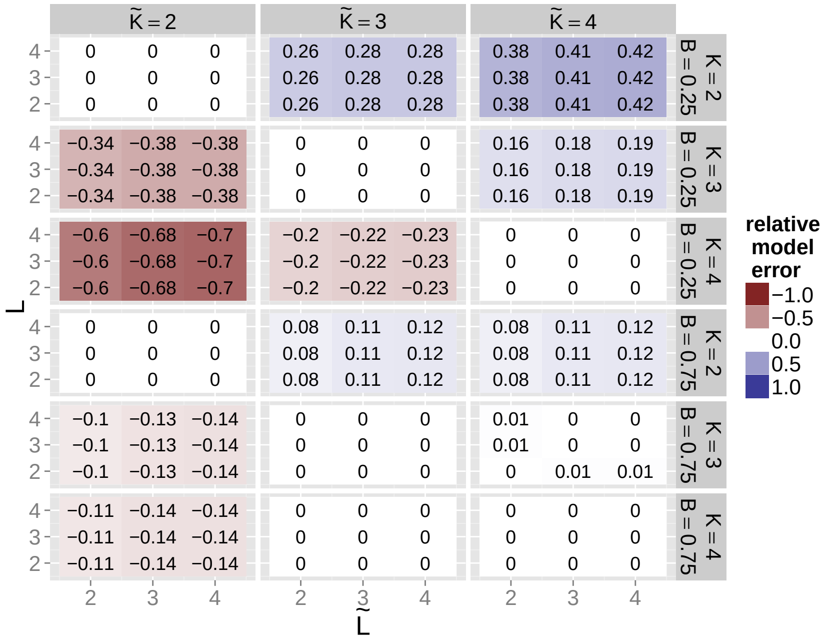


Figure 4: Robustness error for d49 dataset–analysis of budget (number of facilities $(J) = 4$, all-exposed hazard exposure)

determines the number of demand points as well as their geographical dispersion.

6 Conclusions and Future Work

This paper tested several common assumptions in infrastructure protection modeling: 1) the protection of an infrastructure element is binary (either protected or not), 2) the protection of a facility is perfect (a protected facility is immune to failure), and 3) facility failures are binary (completely operational or completely failed). Toward this end, we formulated the *probabilistic facility protection problem* (PFPP) as a two-stage stochastic program with decision-dependent uncertainty.

Due to the initial non-convex formulation, we presented a greedy algorithm and derived a worst-case performance ratio of 0.63. We also presented a linearized mixed-integer linear stochastic programming formulation and solved it using the L-Shaped method. Despite the greedy algorithm's worst-case bound of 0.63, our experimentation found this ratio to be much better in practice (between 0.954 and 1.0 when the L-Shaped method terminated at optimality).

We also found that a mean-value model compared favorably with the greedy algorithm: its performance ratio was between 0.985 and 1.0 when the stochastic programming formulation was

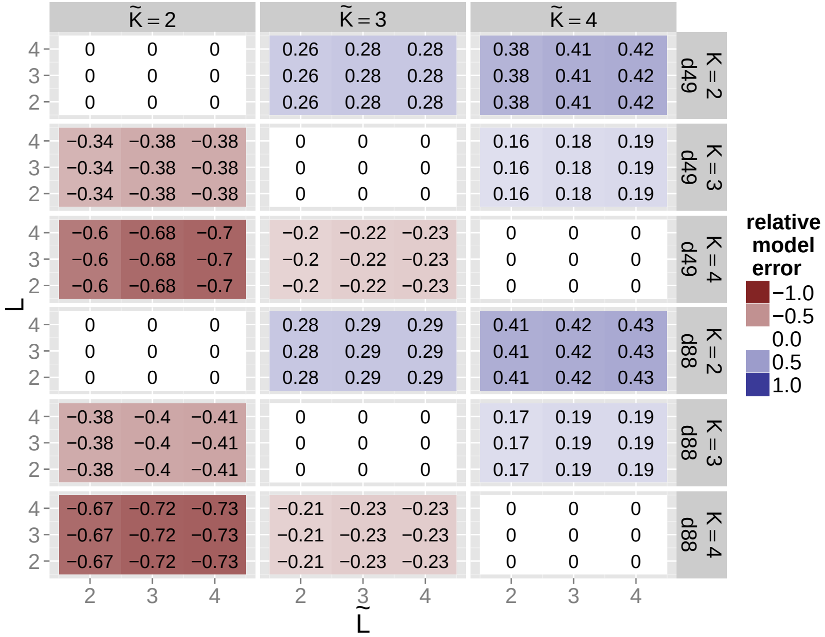


Figure 5: Robustness error– analysis of dataset (number of facilities $(J) = 4$, budget $(B) = 0.25$, all-exposed hazard exposure)

solved to optimality.

Our numerical results point to several insights for infrastructure protection modeling: 1) It is important to model imperfect, multi-level protection. If this feature is not included in a model, the solution quality can be significantly degraded, especially if the budget is low: in one instance the optimal objective of the incorrectly-modeled problem was only 55.0% of the true optimal objective value. 2) A large number of capacity levels are not necessary; two capacity levels are usually sufficient to obtain good quality solutions.

6.1 Discussion

The main implication of our results is that choosing the correct parameters, especially the number of allocation levels, is more important than choosing the correct model or algorithm. Indeed, the problem was robust to the choice of algorithm, the greedy algorithm often returning near-optimal solutions. Moreover, the problem was robust to the choice of model, the mean-value model also often returning near-optimal solutions. On the other hand, the problem was found to be not nearly as robust to changes in the number of allocation levels. When this parameter was incorrectly chosen to be lower than the true value, the solution quality was noticeably less. As a result, infrastructure-protection modelers are advised to model the correct number of allocation levels.

6.2 Future Work

The results of this paper indicate several avenues of future research. First, although we have considered that the post-disruption facility capacities are binomially distributed, other probability distributions would be interesting to study. For example, Shaked (1980) has shown that several distributions from the exponential family are convexly parameterized, i.e., the expectation of a convex function is convex in the distribution parameter. Like the binomial, these distributions may also be submodular in the multivariate case under the assumptions of the PFPP. Second, it would be useful to examine other risk measures such as value-at-risk and conditional-value-at-risk. Third, although we have considered deterministic demand, studying the problem with stochastic demand could yield additional insights. Fourth, there has been some work on multi-stage stochastic programming with decision-dependent uncertainty, but there have been no studies on the type of DDU that we considered. Finally, a case study is an important next step along this line of research. Because the findings of this paper are based on synthetic data, an important next step is to validate these findings on real data. Such a case study would involve using historical weather data to estimate the frequence and severity of extreme events. In addition, a real distribution network should be used that includes facility location and capacity data.

Acknowledgment

We would like to thank two anonymous referees for their valuable comments, which helped improve this paper. We also wish to thank the US Department of Homeland Security (DHS) for sponsoring this work through the Mack-Blackwell Rural Transportation Center at the University of Arkansas through Grant Number DHS-1101. However, the views expressed in this paper do not represent those of DHS, but rather those of the authors. This research was also supported in part by the National Science Foundation through grants MRI-R2 #0959124 , ARI #0963249, Epscor Track II #0918970, and a grant from the Arkansas Science and Technology Authority, with resources managed by the Arkansas High Performance Computing Center.

References

Aksen, D., Piyade, N., and Aras, N. (2010) The Budget Constrained R-Interdiction Median Problem with Capacity Expansion. in *Central European Journal of Operations Research*, **18**(3), 269–291.

Bier, V. M., Gratz, E. R., Haphuriwat, N. J., Magua, W., *et al.* (2007) Methodology for Identifying Near-Optimal Interdiction Strategies for a Power Transmission System. in *Reliability Engineering and System Safety*, **92**(9), 1155–61.

Birge, J. R. and Louveaux, F. V. (1988) A Multicut Algorithm for Two-stage Stochastic Linear Programs. in *European Journal of Operational Research*, **34**(3), 384–392.

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

Boyd, S. P. and Vandenberghe, L. (2004) *Convex optimization*, Cambridge university press, Cambridge.

Cappanera, P. and Scaparra, M. P. (2011) Optimal Allocation of Protective Resources in Shortest-Path Networks. in *Transportation Science*, **45**(1), 64–80.

Church, R. L. and Scaparra, M. P. (2007) Protecting Critical Assets: the R-Interdiction Median Problem With Fortification. in *Geographical Analysis*, **39**(2), 129–146.

Daskin, M. S. (1995) *Network and Discrete Location: Models, Algorithms, and Applications*, John Wiley and Sons, Hoboken, New Jersey.

Du, L. and Peeta, S. (2014) A Stochastic Optimization Model to Reduce Expected Post-Disaster Response Time Through Pre-Disaster Investment Decisions. in *Networks and Spatial Economics*, **14**, 271–295.

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

Hakimi, S. L. (1964) Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph. in *Operations Research*, **12**(3), 450–459.

Jonsbraaten, T. W., Wets, R. J. B., and Woodruff, D. L. (1998) A class of stochastic programs with decision dependent random elements. in *Ann. Oper. Res.*, **82**, 83–106.

Kim, C.-R. (2012) Toyota: supply chain will be ready by autumn for next big quake. in *Reuters*, march 2, 2012.

Kleywegt, A. J., Shapiro, A., and Homem-de Mello, T. (2001) The sample average approximation method for stochastic discrete optimization. in *SIAM J. Optim.*, **12**(2), 479–502.

Liberatore, F., Scaparra, M. P., and Daskin, M. (2011) Analysis of Facility Protection Strategies Against Uncertain Numbers of Attacks: The Stochastic R-Interdiction Median Problem with Fortification. in *Computers and Operations Research*, **38**(1), 357–366.

Lim, M., Daskin, M., Chopra, S., and Bassamboo, A. (2010) A Facility Reliability Problem: Formulation, Properties, and Algorithm. in *Naval Research Logistics*, **57**(1), 58–70.

Liu, C., Fan, Y., and Ordóñez, F. (2009) A Two-Stage Stochastic Programming Model for Transportation Network Protection. in *Computers and Operations Research*, **36**(5), 1582–90.

Losada, C., Scaparra, Church, R., and Daskin, M. (2012) The stochastic interdiction median problem with disruption intensity levels. in *Annals of Operations Research*, **201**(1), 345–365.

Morton, D. P., Pan, F., and Saeger, K. J. (2007) Models for Nuclear Smuggling Interdiction. in *IIE Transactions*, **38**(1), 3–14.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978) An analysis of approximations for maximizing submodular set functions – I. in *Mathematical Programming*, **14**(1), 265–294.

O'Hanley, J. R., Church, R. L., and Gilless (2007) The importance of *In Situ* site loss in nature reserve selection: Balancing notions of complementarity and robustness. in *Biological Conservation*, **2**(1), 170–180.

O'Hanley, J. R., Paola Scaparra, M., and Garcia, S. (2013) Probability Chains: A General Linearization Technique for Modeling Reliability in Facility Location and Related Problems. in *European Journal of Operational Research*, **230**(1), 63–75.

Peeta, S., Salman, F. S., Gunnec, D., and Viswanath, K. (2010) Pre-Disaster Investment Decisions for Strengthening a Highway Network. in *Computers and Operations Research*, **37**(10), 1708–1719.

Peng, P., Snyder, L. V., Liu, Z., and Lim, A. (2011) Reliable Logistics Networks Design with Facility Disruptions. in *Transportation Research-Part B*, **45**(8), 1190–1211.

Ramirez-Marquez, J. E., Rocco, C. M., and Levitin, G. (2009) Optimal Protection of General Source-Sink Networks via Evolutionary Techniques. in *Reliability Engineering and System Safety*, **94**(10), 1676–1684.

Ruszczyński, A. (1986) A regularized decomposition method for minimizing a sum of polyhedral functions. in *Mathematical Programming*, **35**(3), 309–333.

Scaparra, M. P. and Church, R. L. (2008a) A Bilevel Mixed-Integer Program for Critical Infrastructure Protection Planning. in *Computers and Operations Research*, **35**(6), 1905–1923.

Scaparra, M. P. and Church, R. L. (2008b) An Exact Solution Approach for the Interdiction Median Problem with Fortification. in *European Journal of Operational Research*, **189**(1), 76–92.

Scaparra, M. P. and Church, R. L. (2012) Protecting Supply Systems to Mitigate Potential Disaster: A Model to Fortify Capacitated Facilities. in *International Regional Science Review*, **35**(2), 188–210.

Shaked, M. (1980) On Mixtures from Exponential Families. in *Journal of the Royal Statistical Society. Series B (Methodological)*, **42**(2), 192–198.

Shapiro, A., Dentcheva, D., and Ruszczyński, A. (2009) *Lectures on stochastic programming: modeling and theory*, volume 9 of *MPS-SIAM Series on Optimization*, Society for Industrial and Applied Mathematics, Philadelphia, PA.

Snyder, L. V., Scaparra, M. P., Daskin, M. S., and Church, R. L. (2006) Planning for Disruptions in Supply Chain Networks, in M. P. Johnson, B. Norman, and N. Secomandi (eds.), *TutORials in Operations Research: Models, Methods, and Applications for Innovative Decision Making*, TutORials in Operations Research, chapter 9, pp. 234–257, INFORMS, Baltimore, MD.

Topkins, D. (1998) *Supermodularity and Complementarity*, Princeton University Press, Princeton, New Jersey.

Van Slyke, R. and Wets, R. (1969) L-shaped linear programs with applications to control and stochastic programming. in *SIAM Journal on Applied Mathematics*, **17**, 638–663.

Vondrak, J. (2008) Optimal Approximation for the Submodular Welfare Problem in the Value Oracle Model, in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, STOC '08, pp. 67–74, ACM, New York, NY, USA.

Wolsey, L. A. (1982) An analysis of the greedy algorithm for the submodular set covering problem. in *Combinatorica*, **2**(4), 385–393.

Yao, Y., Edmunds, T., Papageorgiou, D., and Alvarez, R. (2007) Trilevel Optimization in Power Network Defense. in *IEEE Transactions On Systems Man And Cybernetics Part C*, **37**(4), 712–718.

Zhu, Y., Zheng, Z., Zhang, X., and Cai, K. (2013) The r-interdiction median problem with probabilistic protection and its solution algorithm. in *Computers and Operations Research*, **40**(1), 451–462.

Biographies

Hugh Medal is an Assistant Professor in the Department of Industrial and Systems Engineering at Mississippi State University. His research and teaching interests are in operations research, with a specialty in improving the security of networked systems. He has published articles on this topic in journals such as the European Journal of Operational Research, the Reliability Engineering and Systems Safety Journal, and the International Journal of Risk Assessment and Management. His research has been funded by agencies such as the U.S. Army, the Joint Fire Science Program, and the U.S. Department of Homeland Security.

Edward A. Pohl is a Professor and Head of the Industrial Engineering Department at the University of Arkansas. Ed also serves as the Director of the Center for Innovation in Healthcare Logistics (CIHL). Ed previously served as the Director of the Operations Management Program. He has participated and led several risk and supply chain related research efforts at the University of Arkansas. Before coming to Arkansas, Ed spent twenty-one years in the United States Air Force where he served in a variety of engineering, operations analysis and academic positions during his career. Previous assignments include the Deputy Director of the Operations Research Center at the United States Military Academy, Operations Analyst in the Office of the Secretary of Defense where he performed independent cost schedule, performance and risk assessments on Major DoD acquisition programs, and as a munitions logistics manager at the Air Force Operational Test Center. Ed received his Ph.D. in Systems and Industrial Engineering from the University of Arizona. He holds a M.S. in Systems Engineering from the Air Force Institute of Technology, and M.S. in Reliability Engineering from the University of Arizona, an M.S. in Engineering Management from the University of Dayton, and a B.S. in Electrical Engineering from Boston University. His primary research interests are in risk, reliability, engineering optimization, healthcare and supply chain risk analysis, decision making, and quality. Ed is a Certified Materials and Resource Professional (CMRP). Ed is a Fellow of IIE, and a Senior Member of IEEE, Senior Member of ASQ, member of INCOSE, INFORMS, ASEE, ASEM and AHRMM.

Manuel Rossetti is a Professor of Industrial Engineering and the Director for the NSF I/UCRC

Center for Excellence in Logistics and Distribution (CELDi) at the University of Arkansas. Dr. Rossetti has published over 90 journal and conference articles in the areas of simulation, logistics/inventory, and healthcare and has been the PI or Co-PI on funded research projects totaling over 4.5 million dollars. He was selected as a Lilly Teaching Fellow in 1997/98 and was voted Best IE Teacher by IE students in 2007 & 2009. He won the IE Department Outstanding Teacher Award in 2001-02, 2007-08, and 2010-11. He received the College of Engineering Imhoff Teaching Award in 2012 and was elected an IIE Fellow. In 2013, the UA Alumni Association awarded Dr. Rossetti the Charles and Nadine Baum Faculty Teaching Award, the highest award for teaching at the university. He is also the author of the book, Simulation Modeling and Arena, published by John Wiley & Sons.

A Proofs

Proof. **Lemma 1**. The first derivative of $b_{k,n}(p)$ is:

$$\frac{d}{dp} b_{k,n}(p) = n \left(b_{k-1,n-1}(p) - b_{k,n-1}(p) \right). \tag{21}$$

The first derivative of (5) is

$$\mathcal{Q}'(p) = \sum_{k=0}^{n} h(k) n \left(b_{k-1,n-1}(p) - b_{k,n-1}(p) \right) = n \left[\sum_{k=0}^{n} h(k) b_{k-1,n-1}(p) - \sum_{k=0}^{n} h(k) b_{k,n-1}(p) \right].$$

Re-indexing the first summation with $k = \ell + 1$ and the second with $k = \ell$, we have

$$\mathcal{Q}'(p) = \frac{d}{dp} \mathbb{E}_{X \sim \text{Binom}(n,p)} h(X) = n \sum_{\ell=0}^{n-1} \left[h(\ell+1) - h(\ell) \right] b_{\ell,n-1}(p) = n \mathbb{E}_{\tilde{a}_1 \sim \text{Binom}(n-1,p)} (\Delta h)(X),$$

where Δ is the forward difference operator, i.e.,$(\Delta h)(\tilde{a}_1) = h(\tilde{a}_1 + 1) - h(\tilde{a}_1)$.

Applying this result again,

$$\mathcal{Q}''(p) = n^2 \mathbb{E}_{X \sim \text{Binom}(n-1,p)} (\Delta^2 h)(X),$$

where Δ^2 is the second forward difference operator. Thus, $\mathcal{Q}''(p) \leq 0$ on $[0,1]$ when $h(\cdot)$ is concave, implying that $\mathcal{Q}(p)$ is concave.

□

Proof. **Corollary 1**. Let

$$\mathcal{Q}(y_1) = \mathbb{E}_{\tilde{\psi}_1} \sum_{k=0}^{n} h(k) g_1(a_1 | \tilde{\psi}_1; y_1).$$

In the case that $\tilde{\psi}_1$ has a point mass distribution, $\mathcal{Q}(y_1)$ is concave due to Lemma 1 and the fact that the composition of concave functions $\mathcal{Q}(f(y_1))$ preserves concavity because $f(\cdot)$ is non-decreasing. In the case in which $\tilde{\psi}_1$ has more than one element in its range, $\mathcal{Q}(y_1)$ is concave because a non-negative weighted sum preserves concavity (Boyd and Vandenberghe, 2004). □

Proof. **Theorem 1.** The first derivative of $\mathcal{Q}(\mathbf{p})$ is

$$\frac{\partial \mathcal{Q}(\mathbf{p})}{\partial p_{j'}} = \sum_{k_1=0}^{n} \cdots \sum_{k_m=0}^{n} h(\mathbf{k}) \prod_{j \neq j'} b_{k_j,n}(p_j) n \left(b_{k_{j'}-1,n-1}(p_{j'}) - b_{k_{j'},n-1}(p_{j'}) \right).$$

Re-indexing the two terms of the summation separately with $k_{j'} = \ell + 1$ and $k_{j'} = \ell$ gives

$$\frac{\partial \mathcal{Q}(\mathbf{p})}{\partial p_{j'}} = n \sum_{\ell=0}^{n-1} \sum_{\substack{k_j=0 \\ j \neq j'}}^{n} \left[h(\mathbf{k}(j', \ell+1)) - h(\mathbf{k}(j', \ell)) \right] \prod_{j \neq j'} b_{k_j,n}(p_j) b_{\ell,n-1}(p_{j'}).$$

The second derivative is

$$\frac{\partial^2 \mathcal{Q}(\mathbf{p})}{\partial p_{j'} \partial p_{j''}} = n \sum_{\ell=0}^{n-1} \sum_{\substack{k_j=0 \\ j \notin \{j', j''\}}}^{n} \left[h(\mathbf{k}(j', \ell+1)) - h(\mathbf{k}(j', \ell)) \right]$$

$$\prod_{j \notin \{j', j''\}} b_{k_j,n}(p_j) b_{\ell,n-1}(p_{j'}) n \left(b_{k_{j''}-1,n-1}(p_{j''}) - b_{k_{j''},n-1}(p_{j''}) \right).$$

Re-indexing the two terms of the summation separately with $k_{j''} = \ell' + 1$ and $k_{j''} = \ell'$ gives

$$\frac{\partial^2 \mathcal{Q}(\mathbf{p})}{\partial p_{j'} \partial p_{j''}} = n^2 \sum_{\ell=0}^{n-1} \sum_{\ell'=0}^{n-1} \sum_{\substack{k_j=0 \\ j \notin \{j',j''\}}}^{n} \Big[\big(h(\mathbf{k}(j',\ell+1)(j'',\ell'+1)) - h(\mathbf{k}(j',\ell+1)(j'',\ell'))\big) - \big(h(\mathbf{k}(j',\ell)(j'',\ell'+1)) - h(\mathbf{k}(j',\ell)(j'',\ell'))\big)\Big] \prod_{j \notin \{j',j''\}} b_{k_j,n}(p_j) b_{\ell,n-1}(p_{j'}) b_{k_{j''},n-1}(p_{j''}).$$

Let $\mathbf{x} = \mathbf{k}(j',\ell)(j'',\ell')$, $\mathbf{y} = \mathbf{k}(j',\ell+1)(j'',\ell')$ and $\mathbf{h} = (\ell'+1)e_{j''}$, where e_j is a vector with a 1 at element j and a zero everywhere else. Note that $\mathbf{x} \leq \mathbf{y}$. By the submodularity of $h(\cdot)$ (see definition 1), we have that the difference in brackets is negative. Thus, $\frac{\partial^2 \mathcal{Q}(\mathbf{p})}{\partial p_{j'} \partial p_{j''}} \leq 0$, meaning that $\mathcal{Q}(\mathbf{p})$ has increasing differences (Topkins, 1998). Because a function has increasing differences if and only if it is submodular (Topkins, 1998, pgs. 42–45), we have the result. □

Proof. **Corollary 2.** Let

$$\mathcal{Q}(\boldsymbol{y}) = \mathbb{E}_{\tilde{\boldsymbol{\psi}}} \sum_{k_1=0}^{n} \cdots \sum_{k_m=0}^{n} h(\mathbf{k}) \prod_{j \in \mathcal{J}} g_j(a_j | \tilde{\psi}_j; y_j).$$

Using vector versions of the arguments made in corollary 1, $\mathcal{Q}(\boldsymbol{y})$ is submodular. □

Proof. **Corollary 3**. To show this result we will convert the discretized version of our optimization problem (4) into the problem of maximizing a submodular set function. Define the set S_j as the set that contains the protection units allocated to facility j; note that $0 \leq |S_j| \leq K$. Then define the set $S = \bigcup_{j \in \mathcal{J}} S_j$.

Thus, adding an element to S is equivalent to adding an additional protection unit to some facility j. We can then define the submodular set function

$$f(S) = \mathcal{Q}(\boldsymbol{y}(S)) = \sum_{\omega \in \Omega} \mathbb{P}^{\omega}(\boldsymbol{y})\, h(\boldsymbol{a}^{\omega}),$$

where $\boldsymbol{y}(S)$ as a vector whose j^{th} element equals $|S_j|$. By the definition of the feasible region (6), our problem can be posed as the maximization a monotone submodular function over a uniform matroid: $\max_{S \subseteq N} \{f(S) : |S| \leq b\}$.

Thus, our greedy algorithm (Algorithm 1), which is equivalent to the R-step greedy algorithm presented in Nemhauser *et al.* (1978), has the desired performance guarantee of $1 - \frac{1}{e}$. □

B Mean Value Model

The *mean value problem* for the PFPP, a deterministic model that replaces the random parameters in PFPP with their mean values, can be formulated as follows:

$$\begin{aligned}
\textbf{MVP}: \quad \max. \quad & \sum_{i \in \mathcal{I}} \sum_{j=1}^{J+1} e_i u_{ij} x_{ij} && \text{(22a)} \\
\text{s.t.} \quad & \sum_{i \in \mathcal{I}} e_i x_{ij} \leq \sum_{k \in \mathcal{K}} \mathbb{E}\left[\tilde{a}_j | y_{jk} = 1\right] y_{jk} \quad \forall j \in \mathcal{J}, && \text{(22b)} \\
& \sum_{j=1}^{J+1} x_{ij} = 1 \quad \forall i \in \mathcal{I}, && \text{(22c)} \\
& x_{ij} \geq 0 \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, && \text{(22d)} \\
& (8b)\text{–}(8d), && \text{(22e)}
\end{aligned}$$

where $\mathbb{E}\left[\tilde{a}_j | y_{jk} = 1\right] = \sum_{\ell \in \mathcal{L}} \mathbb{E}_{\tilde{\psi}_j} \left[\mathbb{P}_{j,\ell,\tilde{\psi}_j,k}\right] a_{j\ell}$.

Table 2: Runtime and solution quality for greedy algorithm and two-stage stochastic programming model.

Dataset-J-K-L-b					Half-Exposed				All-Exposed				Conditional			
					Run time SP	Run time G	Gap	$\mathcal{Q}(\boldsymbol{y}^G)/\mathcal{Q}(\boldsymbol{y}^{lb})$	Run time SP	Run time G	Gap	$\mathcal{Q}(\boldsymbol{y}^G)/\mathcal{Q}(\boldsymbol{y}^{lb})$	Run time SP	Run time G	Gap	$\mathcal{Q}(\boldsymbol{y}^G)/\mathcal{Q}(\boldsymbol{y}^{lb})$
d49	6	3	3	4	40	3	0	0.996	38	3	0	0.991	26	4	0	1
d49	6	3	3	8	138	3	0	0.998	131	4	0	1	72	4	0	1
d49	6	3	4	4	62	8	0	0.997	59	8	0	0.993	117	14	0	1
d49	6	3	4	8	216	10	0	0.998	213	10	0	1	310	17	0	1
d49	6	4	3	6	177	4	0	0.964	158	4	0	1	91	6	0	1
d49	6	4	3	12	991	5	0	0.996	1064	7	0	0.999	286	7	0	0.999
d49	6	4	4	6	335	14	0	0.963	267	14	0	1	438	25	0	1
d49	6	4	4	12	1760	17	0	1	1658	17	0	1	1429	30	0	0.999
d49	9	3	3	6	3602	50	0.04	1.01	3602	50	0.04	1.021	3613	84	0.04	1.001
d49	9	3	3	12	3601	60	0.17	1.139	3600	59	0.1	1.074	3607	104	0.01	1
d49	9	3	4	6	←mem.→				3619	746	0.33	1.316	←mem.→			
d49	9	3	4	12	←mem.→											
d49	9	4	3	9	3601	87	0.03	1	3601	89	0.03	1	3613	152	0.03	0.999
d49	9	4	3	18	3602	105	0.35	1.314	3602	107	0.22	1.201	3602	187	0.01	1.001
d49	9	4	4	9	3610	1117	0.13	1.074	←mem.→							
d49	9	4	4	18	←mem.→											
d88	6	3	3	4	39	4	0	1	38	4	0	1	28	5	0	1
d88	6	3	3	8	141	5	0	0.996	133	5	0	1	70	6	0	1
d88	6	3	4	4	63	12	0	0.999	60	12	0	1	118	18	0	1
d88	6	3	4	8	223	14	0	0.996	215	13	0	1	304	20	0	1
d88	6	4	3	6	173	7	0	1	156	7	0	1	93	8	0	1
d88	6	4	3	12	997	7	0	1	1017	7	0	1	280	9	0	1
d88	6	4	4	6	338	20	0	0.954	271	20	0	1	449	31	0	1
d88	6	4	4	12	←mem.→				1569	22	0	1	1444	35	0	1
d88	9	3	3	6	3602	77	0.03	1.016	3600	76	0.02	1.01	3613	109	0.04	1.004
d88	9	3	3	12	3604	82	0.11	1.084	3602	83	0.09	1.077	3607	136	0.01	1
d88	9	3	4	6	←mem.→											
d88	9	3	4	12	←mem.→											
d88	9	4	3	9	3601	127	0.02	0.978	3601	124	0.02	1	3604	187	0.05	1.013
d88	9	4	3	18	3603	142	0.06	1.037	3602	143	0.23	1.211	3603	223	0.02	1.013
d88	9	4	4	9	←mem.→											
d88	9	4	4	18	←mem.→											
All				min.	39	3	0.00	0.954	38	3	0.00	0.991	26	4	0.00	0.999
				max.	3610	1117	0.35	1.314	3619	746	0.33	1.316	3613	223	0.05	1.013
				avg.	1588	83	0.04	1.021	1579	65	0.04	1.036	1434	59	0.01	1.001
				stdev.	1639	225	0.081	0.074	1610	148	0.088	0.083	1611	70	0.015	0.004
0% Gap				min.				0.954				0.991				0.999
				max.				1.000				1.000				1.000
				avg.				0.990				0.999				1.000
				stdev.				0.016				0.003				0.000

Table 3: Value of the stochastic solution for several problem instances.

					Half-Exposed			All-Exposed			Conditional		
Dataset-*J*-*K*-*L*-*b*					SP	*VSSR*	*UBR*	SP	*VSSR*	UBR	SP	*VSSR*	*UBR*
					Opt.			Opt.			Opt.		
					Gap			Gap			Gap		
d49	6	3	3	4	0	0.996	1.041	0	1.000	1.030	0	1.000	1.037
d49	6	3	3	8	0	1.000	1.044	0	1.000	1.033	0	0.994	1.016
d49	6	3	4	4	0	0.997	1.028	0	1.000	1.021	0	1.000	1.032
d49	6	3	4	8	0	1.000	1.030	0	1.000	1.021	0	0.995	1.013
d49	6	4	3	6	0	1.000	1.052	0	1.000	1.038	0	1.000	1.035
d49	6	4	3	12	0	1.000	1.038	0	0.999	1.037	0	0.985	1.011
d49	6	4	4	6	0	1.000	1.035	0	1.000	1.024	0	1.000	1.030
d49	6	4	4	12	0	1.000	1.025	0	1.000	1.025	0	0.986	1.008
d49	9	3	3	6	0.04	1.011	1.036	0.04	1.022	1.038	0.04	1.005	1.039
d49	9	3	3	12	0.17	1.139	1.171	0.1	1.074	1.095	0.01	0.996	1.011
d49	9	3	4	6		←mem.→		0.33	1.316	1.333		←mem.→	
d49	9	3	4	12					←mem.→				
d49	9	4	3	9	0.03	1.000	1.032	0.03	1.000	1.025	0.03	1.006	1.033
d49	9	4	3	18	0.35	1.318	1.354	0.22	1.202	1.225	0.01	0.996	1.008
d49	9	4	4	9	0.13	1.105	1.128			←mem.→			
d49	9	4	4	18					←mem.→				
d88	6	3	3	4	0	0.999	1.022	0	0.998	1.016	0	1.000	1.038
d88	6	3	3	8	0	1.000	1.029	0	1.000	1.025	0	0.999	1.016
d88	6	3	4	4	0	1.000	1.014	0	1.000	1.012	0	1.000	1.034
d88	6	3	4	8	0	1.000	1.019	0	1.000	1.016	0	0.999	1.012
d88	6	4	3	6	0	1.000	1.031	0	1.000	1.028	0	1.000	1.036
d88	6	4	3	12	0	1.000	1.028	0	1.000	1.028	0	0.994	1.011
d88	6	4	4	6	0	1.000	1.020	0	1.000	1.018	0	1.000	1.032
d88	6	4	4	12		←mem.→		0	1.000	1.018	0	0.995	1.008
d88	9	3	3	6	0.03	1.016	1.031	0.02	1.009	1.023	0.04	1.004	1.043
d88	9	3	3	12	0.11	1.084	1.106	0.09	1.076	1.095	0.01	0.978	1.012
d88	9	3	4	6					←mem.→				
d88	9	3	4	12					←mem.→				
d88	9	4	3	9	0.02	1.000	1.025	0.02	1.000	1.019	0.05	1.013	1.047
d88	9	4	3	18	0.06	1.040	1.061	0.23	1.211	1.233	0.02	0.990	1.021
d88	9	4	4	9					←mem.→				
d88	9	4	4	18					←mem.→				
			All	min.	0.000	0.996	1.014	0.000	0.998	1.012	0.000	0.978	1.008
				max.	0.350	1.318	1.354	0.330	1.316	1.333	0.050	1.013	1.047
				avg.	0.04	1.03	1.06	0.04	1.04	1.06	0.01	1.00	1.02
				stdev.	0.08	0.07	0.07	0.09	0.08	0.08	0.02	0.01	0.01
			Opt.	min.		0.996			0.998			0.985	
				max.		1.000			1.000			1.000	
				avg.		0.999			1.000			0.997	
				stdev.		0.507			0.508			0.506	