UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

ROBUST RESOURCE ALLOCATION TO INTERDEPENDENT NETWORKS UNDER MULTIPLE DISRUPTION SCENARIOS

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

In partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

HANNAH LOBBAN
Norman, Oklahoma
2017

ROBUST RESOURCE ALLOCATION TO INTERDEPENDENT NETWORKS
UNDER MULTIPLE DISRUPTION SCENARIOS

A THESIS APPROVED FOR THE
SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

BY

_____
Dr. Kash Barker, Chair

_____
Dr. Charles Nicholson

_____
Dr. Theodore Trafalis

To my friends and family,

without whom I would not be where I am.

Acknowledgements

First, I owe so much gratitude to Yasser Almoghathawi, for his invaluable insights on interdependent networks and Python and Gurobi coding. It is only with his knowledge and guidance that the work in this thesis was accomplished.

Second, I have to thank Nazanin Morshedlou for being the most encouraging, kindest office-mate ever. She always has a smile and words of advice, and kept me from stressing too much over the last semester.

I also, of course, must thank the professors and staff in the ISE department, especially my advisor Dr. Kash Barker. I have learned and grown so much not only in graduate school but over the past five years and I look forward to applying those lessons in the future.

Last, but never least, thank you to my family and friends for their constant support and general wonderfulness. They have each impacted who I am as a person and I endeavor to live up to their expectations.

Table of Contents

List of Tables

List of Figures

Abstract

In recent years, the threat of international attacks on critical infrastructure networks has grown; as defined by a 1998 Presidential Decision Directive, these are networks necessary to society's functionality and include water, power, communication, transportation, and more. Due to existing interdependencies, damage to a small area of one of the networks could have far-reaching effects on the ability to meet demand across the entire system. In similar work, common scenarios for malevolent attacks include degree- and capacity-based disruptions. However, attackers targeting a network may consider some components more desirable than others for qualitative reasons such as religious or governmental significance. Additionally, the concept of social vulnerability, which describes an area's ability to prepare for and respond to a disruption, must be included. This should promote not only the protection of the most at-risk components but also ensure that socially vulnerable communities are given adequate resources. This work attempts to determine the allocation of defensive resources that accounts for all these factors while minimizing both costs and the unmet demand in the disrupted network.

Chapter 1: Introduction and Motivation

Over the past 50 years, more than 2,500 foreign and domestic terrorist attacks have occurred in the United States. Understandably, high-fatality attacks on visible public targets such as the Alfred P. Murrah Federal Building bombing in Oklahoma City, the attacks on 9/11, and the mass shooting at Pulse nightclub in Orlando garner more attention and make a larger memorable impact on society. However, these types of attacks only account for 10-20% of the total; the majority are non-lethal but nevertheless can cause significant economic and psychological damage to the nation. In a recent report for the Department of Homeland Security (DHS), the National Consortium for the Study of Terrorism and Responses to Terrorism classifies attack by target type and shows that 75% of attacks focus on the critical infrastructure sector (Miller, 2016).

From this report, it is clear that there is significant risk of disruptive events affecting U.S. infrastructure. As defined by the Presidential Decision Directives regarding Critical Infrastructure Protection, critical infrastructures are networks are service and utility systems that are considered necessary for society to function. Critical infrastructure includes energy, water, transportation, and communication systems, among many others on which all aspects of our society—from the public, to the government and businesses—depend. With the growth of technology in recent years, the definition of critical infrastructure networks has expanded to include cyber-based systems. While these advances, and the desire to make such networks more efficient, have improved the overall system functionality, they come at a cost and "have created new vulnerabilities to equipment failure, human error, weather and other natural causes,

and physical and cyber attacks. Addressing these vulnerabilities will necessarily require flexible, evolutionary approaches" to ensure that both the infrastructure itself and the population in potentially at-risk areas are protected (PDD 63, 1998). This problem stated in the Directive nearly 20 years ago is even more pressing today, as higher degrees of connectivity and communication have made networks interdependent. Interdependency is defined in Rinaldi et al. (2001) as a "bidirectional relationship between two infrastructures through which the state of each infrastructure influences or is correlated to the state of the other;" the relationship between components of different networks increases the complexity of the network system as a whole.

Due to the existing interdependencies, damage to a section of one infrastructure network could have far-reaching effects on other networks in the system; high vulnerability in even a small number of components has significant negative potential. To combat this vulnerability, a defensive strategy that concentrates on susceptible components can be implemented, increasing the overall resilience of the system of networks.

This research aims to provide a framework that decision makers can use to determine the allocation of defensive resources that is robust to a variety of disruption scenarios. It considers a system of multiple interdependent networks subjected to the disruptions and optimizes a multi-objective mathematical model to determine allocation options that reduce both vulnerability (represented by unmet demand) and cost.

The composition of this is paper is as follows. Chapter 2 discusses the background of the problem, citing previous works that contributed to the work presented here. Chapter 3 details the methodology and model formulation, while

Chapter 4 demonstrated how the model can be applied to an example network system under disruptions. Lastly, Chapter 5 provides concluding remarks on the results from the example and offers recommendations for future work.

Chapter 2: Background

This section discusses some of the prior work on topics related to those presented in this paper.

2.1 Interdependent Networks

As the prevalence and importance of interdependent infrastructure networks grows, more and more researchers are concentrating on increasing system resilience in the face of significant disruptive events. Almoghathawi et al. (2016) describes a system of interdependent critical infrastructure subjected to a range of disruption scenarios: random failure, spatial failure, and malevolent attacks, which are either degree- or capacity-based. The goal of this work is to determine a strategy to restore the networks in a post-disruption period dependent on the availability of resources such as work crews and time. The multi-objective model minimizes flow, disruption, and restoration costs while maximizing resilience. For simplification, the ϵ-constraint method is used for the latter objective; the model is then solved for the cost objective for multiple values of $\epsilon \in [0,1]$. In addition to the resource constraints, the system of networks is also subject to flow, capacity, and interdependency constraints (Almoghathawi et al., 2016).

González et al. (2015) uses a similar formulation for the interdependent network system but a different approach to finding the optimal restoration strategy. The model still incorporates constraints regulating costs, operations, flow, and interdependence, but also takes advantage of the interdependencies to allow for simultaneous restoration processes. The Interdependent Network Design Problem (INDP) developed is then used as input for an iterative algorithm to generate reconstruction solutions at the minimum

cost (González et al., 2015). This research also considers various types of interdependencies that could be present in the critical infrastructure: physical, cyber, geographical, and logical. Understanding the nature in which the individual networks are connected is valuable for deeper contextual analysis of the problem and how the different interdependence types may affect the recovery process.

2.2 Network Allocation

In both works discussed above, the focus is on the period after the given disruption and restoring system functionality. However, the period prior to a disruption is also a key point in disaster planning; implementing accurate preventative measures can reduce the performance loss experienced in the network(s). One instance of this problem is addressed in Qiao et al (2005); the work examines the scenario of a water supply network under a physical, cyber, or biological attack. In this case, the desired output is the optimal allocation of a security budget across the single network. Assuming the attacker has knowledge of the network and will target components whose disruption will cause the most damage, the defender allocates resources so that the attack will be more costly to carry out. That is, the defenders aim to maximize the resilience of the system where a component is defined as resilient if the "consequences that result from the attack are small in comparison to the attack cost" (Qiao et al 2005). They achieve this by employing a linear programming model whose objective function maximizes the minimum network resilience. This research is tailored to the water network, as it incorporates hydraulic modeling and constraints specifically for water flow and pressure and across pipes (links) in the system. However, the general ideas behind the methodology can be applied to a broader range of network allocation problems.

McCarter et al. (2017) expands the allocation problem presented in Qiao et al by looking at multi-commodity networks (as opposed to the single-commodity water network). Five disruption scenarios were randomly generated and applied to the network; for each scenario $k \in [1,5]$, link (i,j) is subjected to e_{ij}^{k} units of attack resource. and the decision variables are the components of defense strategy $\boldsymbol{h}^{l}$. The set of strategies that minimize allocation cost and vulnerability (or maximize network performance) is found with the Non-dominated Sorting Genetic Algorithm II. The solutions in the Pareto set are then evaluated with the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) using as criteria the cost and the network performance in each scenario.

2.3 Contest Functions

As McCarter et al. (2017) demonstrates, the vulnerability of a component in a network subjected to some disruption scenario can be found using a contest function. In general, contest functions are used when multiple players exert effort to win a prize. This can include events such as elections, sports games, and military combat (Baik, 1998). The outcome of the function $p \in (0,1)$ can either represent the probability that a player wins the prize or the percentage of the prize a player wins, depending on the nature of the prize and event for which the function is employed (Hirshleifer, 1989). Multiple forms of contest functions have been developed over the years and research has shown some to be more applicable to some circumstances over others. Considering malevolent attacks, the two most common contest functions are the ratio form and difference form (Levitin and Hausken, 2010). Both forms take as input the amount of attack and defense resources assigned to a given component. The value of the output is dependent on how

the relation between the attack and defense amounts is defined in the function; it is either based on the ratio of the two inputs or the magnitude of difference between them.

2.4 Terrorist Target Selection

The works above have shown multiple different ways to consider disruption scenarios. However, studies in terrorist target selection suggest that attackers do not always choose what defenders typically think of as the logical choice; that is, the components of a system that are most important to network throughput. In allocating resources planners must be aware not only of which components are most critical to the functionality of the infrastructure as a whole but also which components might be more attractive to an attacker. The likelihood that a given component will be included in an intentional attack is based on several factors, some more qualitative than others. Toft et al. (2010) conducted a study into the risk of terrorist attacks on energy networks and created a framework to analyze and categorize motivations from attack data collected in the Global Terrorism Database and the Worldwide Incidents Tracking System. The study suggests that there is a wide range of objectives and strategic value may be less important that other factors such as symbolism, political or religious ideology, feasibility, and intimidation.

On the other hand, an examination into the motivations behind the 2005 attacks on the London underground transportation system revealed that the stations targeted were those associated with the highest flow capacity in terms of passengers (Jordán 2008). Jordán does acknowledge that causing panic and casualties or targeting locations of symbolic value are also motivations, and the goals vary across attacks and terrorist

groups. In any case, the defenders need to weigh the functional value against the attackers' values to gain a fuller understanding of which components are at-risk.

2.5 Social Vulnerability

Another aspect less frequently discussed in disaster planning and resilience is that of social vulnerability. Social vulnerability describes how social factors and inequalities, such as economic disparities, access to emergency services, and political representation, affect a community's ability to respond to and recover from some disruptive event (Cutter et al., 2003). Within the contest of critical infrastructure systems, socially vulnerable areas are those that will be most negatively impacted by a lack of critical services and resources over the disruption time. However, until recently it has been largely excluded from resilience and recovery research partially because of the greater focus on physical and cyber vulnerabilities. Additionally, the numerous factors that contribute to social vulnerability make quantification complicated.

To address this, the Cutter et al. (2003) and University of South Carolina's Hazards and Vulnerability Research Institute developed a method to assign a Social Vulnerability Index (SoVI) score across the United States by county using demographic, housing, and economic data publicly available through the U.S. Census (Cutter et al., 2003). While the most recent version of the SoVI model implements Principal Component Analysis to determine the influence of 29 variables, the research presented in this paper uses a simplified method called SoVI-Lite developed for the United States Army Corps of Engineers (USACE) as a quicker, less technical method for hazard planners (Cutter el al., 2011). SoVI-Lite also allows for the index to be more easily scaled down to determine scores within smaller geographical units for an

individual county and uses a smaller set of variables determined to be the most relevant to the specific USACE region of study. Resulting SoVI scores can then be used as objectives or weights in resilience models to ensure that more vulnerable areas are allocated the necessary prevention or restoration resources.

Chapter 3: Methodology

The research presented in this paper, incorporating and combining concepts from the works in the previous section, fills a gap in the current literature by applying the vulnerability-reducing resource allocation formulation from McCarter et al. (2017) to an interdependent network system as presented in Almoghathawi et al. (2016). Furthermore, this work also considers the intentional attack scenario based on qualitative motivators and includes community resilience metrics to lower the impact of a disruption on society. The following section explains the types disruption scenarios, how concepts of resilience and social vulnerability are incorporated, the variables and parameters for the interdependent networks, and the mathematical model that aims to minimize the effects of the disruption while simultaneously minimizing costs.

3.1 Resilience and Vulnerability

In the field of disaster planning and recovery, the term resilience can take on a variety of meanings. This research employs the framework adapted from Barker et al. (2013) and Henry and Ramirez-Marquez (2012), shown below in Figure 1. The figure shows the state of the network in the time before, during, and after some disruptive event e^k. Here resilience is a function of vulnerability, the amount the network performance $\varphi(t)$ decreases in the disruption period, and of recoverability, how long it takes for the system to return to acceptable performance $\varphi(t_f)$ after experiencing the disrupted state. The model developed in this paper focuses on decreasing vulnerability by enacting preemptive defensive measures prior to the disruptive event (between t_0 and t_e) and is therefore independent of time.

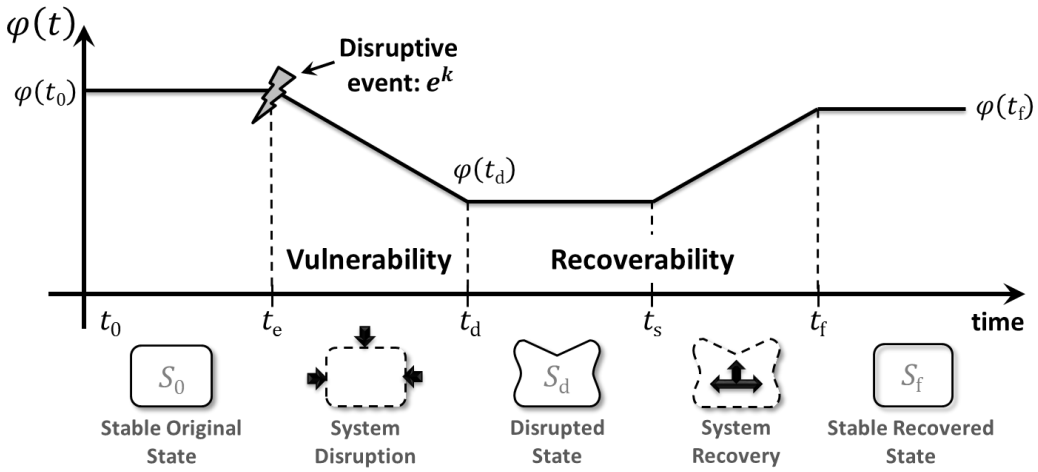

Figure 1: Network performance over time

3.2 Proposed Disruption Scenarios

Wang et al. (2013) categorized disruptive events into three types: natural disasters, malevolent attacks, and random failure. This work proposes three disruption scenarios that fall under the second category and represent different possible target selection motivations from the perspective of the attacker: degree-based, capacity-based, and desirability-based. In degree-based attacks, the defender assumes the attacker will focus on components with a high degree of connections to other nodes or links in the network. The degree of an individual node i is simply the number of connections (number of links in and out of i) and the degree of a link (i, j) is the average of the degrees of nodes i and j. In capacity-based attacks, the attacker targets components with greater capacities. For links, this quantity is the flow capacity across the link. The capacity of a node its supply/demand value; for transshipment nodes whose given capacity is zero we use the minimum between the sum of the node's outgoing links' capacities and its incoming links' capacities ($c_i^k = \min\{\sum_{(j,i)\in L^k} c_{ji}^k, \sum_{(i,l)\in L^k} c_{il}^k\}$). For the last scenario, desirability-based attacks, components are assigned a rank of external importance to represent how desirable they are to the attacker, supposing that the attacker may value

some targets over others for more qualitative, symbolic significance. Due to limited knowledge about the network regarding which components are associated with greater symbolic and ideological values, the components are assigned a rank to approximate the likelihood of attack (high, moderate or low). The majority of nodes selected for the third scenario were in the "high likelihood" category, though some were also chosen from the "moderate" and "low" categories to represent the high degree of unpredictability in attacker motivation. In each of the three scenarios, 25% of the nodes and 20% of the links were disrupted.

3.3 Definitions and Notations

The mathematical model below defines components for *K* interdependent networks. N^k and L^k are the total sets of all the nodes and links, respectively, in each individual network. Within the sets of nodes and links, N^{ks} and N^{kd} are the subsets of supply and demand nodes and $N^{k\prime}$ and $L^{k\prime}$ are the subsets of disrupted nodes and links. The flow on each link from any node i to any node j in network k is defined as x_{ij}^k and has a capacity of c_{ij}^k. For all nodes $i \in N^{ks}$, the available supply, and subsequently the maximum possible flow from node $i \in N^{ks}$ to node $i \in N^{kd}$ is b_i^k. Using this definition, b_i^k is a maximum performance amount representing the desired amount of demand to be met at node $i \in N^{kd}$. For all nodes $i \in N^{kd}$, the reduction from this maximum flow, or slack, is s_i^k and has an associated unit cost of p_i^k; $s_i^k = 0$ when the flows into the demand node sum to b_i^k. The unit cost of allocation resources for a node is q_i^k and the same quantity for a link is r_{ij}^k.

Considering that some of the networks may be more critical to a community's functionality, μ_k is used as a network importance factor; all $i \in N^k$ have the same μ_k. In

the example network system used for this research, the values of μ_k have been assigned to each of the networks randomly, but a more in-depth technique drawing from detailed knowledge of the networks and (such as Analytical Hierarchy Process) could determine weights that accurately reflect the situation and community needs. There are also node attributes that describe the importance of each $i \in N^k$ to attackers and defenders. The SoVI rank (high, moderate, or low social vulnerability) v_i^k for each node is determined for the node's geographical location in the area of study. Lastly, the attribute that describes the attacker's attitude toward and desirability of each potential target is unknown by the defenders but is estimated with γ_i^k.

3.4 Contest Function Simplification

As previously stated, the difference form of contest functions, show in Eq. (1)-(2), is commonly applied to scenarios in which a defender is determining resource amounts to combat an intentional attack. The vulnerability of a node or link (u_i^k and w_{ij}^k, respectively) is calculated for some assumed attack resources g_i^k and h_{ij}^k and allocation of defense resources d_i^k and f_{ij}^k if the component is included in that attack scenario (if the value for the attack resources is greater than zero); if the component is not disrupted, its vulnerability is zero. The expected value for a component's functionality in the disruption scenario can be estimated by multiplying its pre-disrupted value by 1-u_i^k (for nodes) or by 1-w_{ij}^k (for links).

$$u_i^k\left(g_i^k, d_i^k\right) = \begin{cases} \dfrac{e^{g_i^k}}{e^{g_i^k} + e^{d_i^k}} \, if \; g_i^k > 0 \\ 0 \, if \; g_i^k = 0 \end{cases} \tag{1}$$

$$w_{ij}^k(h_{ij}^k, f_{ij}^k) = \begin{cases} \frac{e^{h_{ij}^k}}{e^{h_{ij}^k} + e^{f_{ij}^k}} if\ h_{ij}^k > 0 \\ 0\ if\ h_{ij}^k = 0 \end{cases} \quad (2)$$

This format, though, cannot be used in linear models; to keep the computation simple, this research uses a modified contest function that still calculates vulnerability as a function of the difference between attacker and defender resources. As seen in Eq. (3)-(4), if a defender does not allocate any resources to a component it is completely disrupted and its capacity is reduced to zero. If equal amounts of attack and defense resources are assigned to a component, the vulnerability is zero and the component's capacity is unaffected. The condition still holds that if the amount of attack resources on a given component is zero, that component's vulnerability is zero.

$$u_i^k(g_i^k, d_i^k) = \begin{cases} \frac{g_i^k - d_i^k}{g_i^k} if\ g_i^k > 0 \\ 0\ if\ g_i^k = 0 \end{cases} \quad (3)$$

$$w_{ij}^k(h_{ij}^k, f_{ij}^k) = \begin{cases} \frac{h_{ij}^k - f_{ij}^k}{h_{ij}^k} if\ h_{ij}^k > 0 \\ 0\ if\ h_{ij}^k = 0 \end{cases} \quad (4)$$

3.5 Mathematical Model

$$\min \frac{\sum_{k\in K}\sum_{i\in N^{kd}} \mu^k v_i^k s_i^k}{S} \quad (5)$$

$$\min \sum_{k\in K}\sum_{i\in N^k} q_i^k d_i^k + \sum_{k\in K}\sum_{i,j\in L^k} r_{ij}^k f_{ij}^k \quad (6)$$

Subject to:

$$\sum_{i,j\in L^k} x_{ij}^k \le b_i^k\ , \forall i \in N^{ks}, k\ \in K \quad (7)$$

$$\sum_{i,j\in L^k} x_{ij}^k - \sum_{j,i\in L^k} x_{ji}^k = 0, \forall i \in N^k \backslash \{N^{ks}, N^{kd}\}, k \in K \quad (8)$$

$$\sum_{i,j\in L^k} x_{ji}^k + s_i^k = b_i^k, \forall i \in N^{kd}, k \in K \quad (9)$$

$$x_{ij}^k - c_{ij}^k \leq 0\ , \forall (i,j) \in L^k, k \in K \quad (10)$$

$$x_{ij}^k - (1 - u_i^k)c_{ij}^k \leq 0\ , \forall (i,j) \in L^k, i \in N^{k\prime}, k \in K \quad (11)$$

$$x_{ij}^k - (1 - u_j^k)c_{ij}^k \leq 0\ , \forall (i,j) \in L^k, j \in N^{k\prime}, k \in K \quad (12)$$

$$x_{ij}^k - (1 - w_{ij}^k)c_{ij}^k \leq 0\ , \forall (i,j) \in L^{k\prime}, k \in K \quad (13)$$

$$(1 - u_{\bar{i}}^{\bar{k}}) - (1 - u_i^k) \leq 0\ , \forall ((i,k), (\bar{i},\bar{k}))\ \in \Psi \quad (14)$$

Alternatively (14) can be written as $u_i^k - u_{\bar{i}}^{\bar{k}} \leq 0\ , \forall ((i,k), (\bar{i},\bar{k}))\ \in \Psi$

$$d_i^k \geq 0 \quad (15)$$

$$f_{ij}^k \geq 0 \quad (16)$$

Objective (5) minimizes the total amount of weighted proportional slack across all networks, accounting for demand nodes in networks are more important to the decision makers and those that have less ability to recover from disruption. These weights encourage defensive resources to be first allocated to susceptible components whose removal would cause the most harm, functionally and socially. The quantity S represents the total amount of weighted slack that exists in the network system when no defensive resources have been allocated and the network performance had deteriorated its lowest point. The conflicting objective shown in Eq. (6) minimizes the total cost of the resource allocation strategy $(\boldsymbol{d}, \boldsymbol{f})$, which is comprised of the amount of defensive resources selected to each of the nodes and links in the disruption scenario and the unit cost of the resource at each component. The unit cost is included in the objective, as opposed to looking at just the amount of resource used, to account for the fact that it may be more expensive to assign resources to one area over another.

Constraints (7)-(9) are the flow conservation constraints at each node: the flows x_{ij}^k out of node i cannot exceed the maximum possible flow b_i^k; the sum of the flows out of node i $\left(x_{ij}^k\right)$ must be equal to the sum of the flows into node i (x_{ji}^k); and the total flow into a demand node i (x_{ji}^k) combined with the amount of slack (s_i^k) at that node must be equal to the demand (maximum performance). Constraint (10) ensures that the flow across any link is no more than the link capacity. Constraints (11) -(13) consider link capacity for disrupted components whose capacity has been reduced by a factor related to its vulnerability. The flow between nodes i and j cannot be greater than the disrupted performance level of either node, or the disrupted capacity of the link itself. The interdependency of the network system is described with constraint (14): if node $\bar{\imath}$ in network $\bar{k}$ is dependent on node i in network k, node $\bar{\imath}$ must be at least as vulnerable as node i. Finally, constraints (15) and (16) are nonnegativity constraints for the decision variables d_i^k and f_{ij}^k; the amount of resources assigned to each component must be positive, though non-integer values are allowed.

To simplify the computational aspects of the model, the ϵ-constraint method developed by Haimes et al. (1971) is used; the first objective is constrained by a given value of ϵ representing the maximum allowable vulnerability, as seen in Eq. (17).

$$\frac{\sum_{k\in K}\sum_{i\in N^{kd}}\mu^k v_i^k s_i^k}{S} \leq \epsilon \tag{17}$$

The second objective is still minimized and becomes the single-objective in the mathematical model so that the lowest cost strategy for a determined amount of weighted slack is found. Running the model with several values of ϵ across each of the disruption scenarios gives the set of Pareto-optimal solutions. In this research, nine

values of $\epsilon \in [0,1]$ were run for each disruption scenario for a total of 27 potential solutions.

3.6 Solution Robustness

Once the set of Pareto-optimal solutions has been found, each solution is evaluated to determine which is the most robust with respect to all three scenarios. Robustness in this instance is defined by how well a solution meets the objectives (specifically the first objective, as cost will remain the same for a given solution) when applied to the disruption scenarios for which it was not initially solved. To do this, each of the solutions y_{mn} in the Pareto-optimal set is run on each of the scenarios and the resulting values for vulnerability (weighted proportional slack) and cost are recorded. The data was then evaluated and ranked with TOPSIS, using the vulnerability associated with each scenario and the strategy cost as its four criteria.

Because the criteria have different units, the data are first standardized using the formula shown in Eq. (18).

$$r_{nm}(y) = \frac{y_{nm} - \min_n y_{nm}}{\max_n y_{nm} - \min_n y_{nm}} \tag{18}$$

Next, weights are applied to the standardized values, seen in Eq. (19), so that more important criteria have greater influence on the solution ranking. Criteria can have equal weights, or the weights can be determined by stakeholders' beliefs and preferences.

$$v_{nm}(y) = w_m r_{nm}(y) \tag{19}$$

From the weighted scores the Positive Ideal Solution (PIS) and Negative Ideal Solution (NIS) are found; these represent the best-case (minimum value) scenario and the worst-

case (maximum value) scenario, respectively, in each of the criteria. The formulas for calculating these two values are shown in Eq. (20) and Eq. (21).

$$PIS = A^{+} = \{v_1^{+}(y), \dots, v_m^{+}(y), \dots, v_M^{+}(y)\} \tag{20}$$

$$NIS = A^{-} = \{v_1^{-}(y), \dots, v_m^{-}(y), \dots, v_M^{-}(y)\} \tag{21}$$

The distance of each candidate solution to the PIS and NIS are found using the Euclidian distance function in Eq. (22) and (23).

$$D_n^{+} = \sqrt{\sum_{m=1}^{M} [v_{nm}(y) - v_m^{+}(y)]^2} \; \forall \; n = 1 \dots N \tag{22}$$

$$D_n^{-} = \sqrt{\sum_{m=1}^{M} [v_{nm}(y) - v_m^{-}(y)]^2} \; \forall \; n = 1 \dots N \tag{23}$$

The last step in TOPSIS combines the two distance measures into a single closeness coefficient using Eq. (24).

$$S_n^{+} = \frac{D_n^{-}}{D_n^{+} + D_n^{-}} \tag{24}$$

The coefficients are ranked to determine which defensive strategy is most similar to the ideal solution and least similar to the worst solution. (Hwang et al., 1993). Larger coefficients represent strategies that are closer, based on Euclidean distance, to the PIS and furthest from the NIS.

Chapter 4: Illustrative Example: Shelby County, Tennessee

This section details an application of the above methodology on a critical infrastructure system in Tennessee and discusses the results of the model.

4.1 Critical Infrastructure System

The data used in this research is from a set of three interdependent networks in Shelby County, TN, taken from González et al. (2015). The water, gas, and power networks are shown below in Figures 2-4, respectively. The system is comprised of three interdependent networks with a total of 125 nodes and 328 links. The number of nodes and links in each network, and how many of these nodes are dependent on a node in a different network, are shown in Table 1.

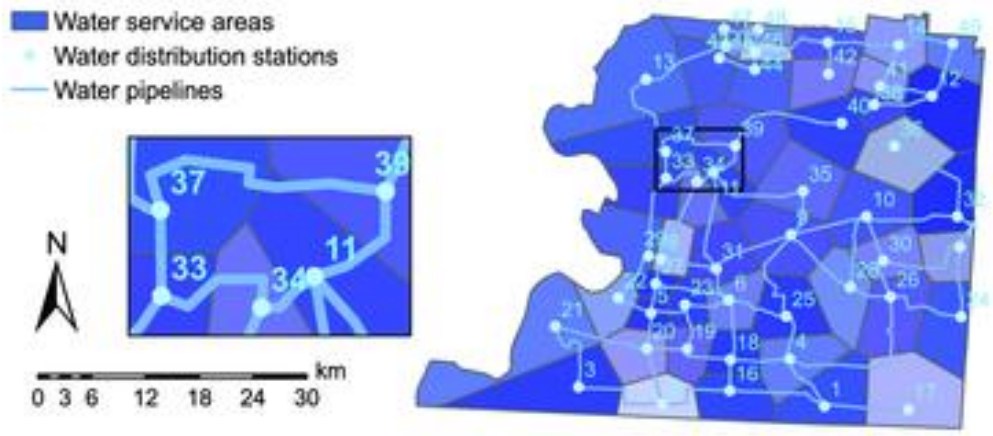

Figure 2: Water network in Shelby County

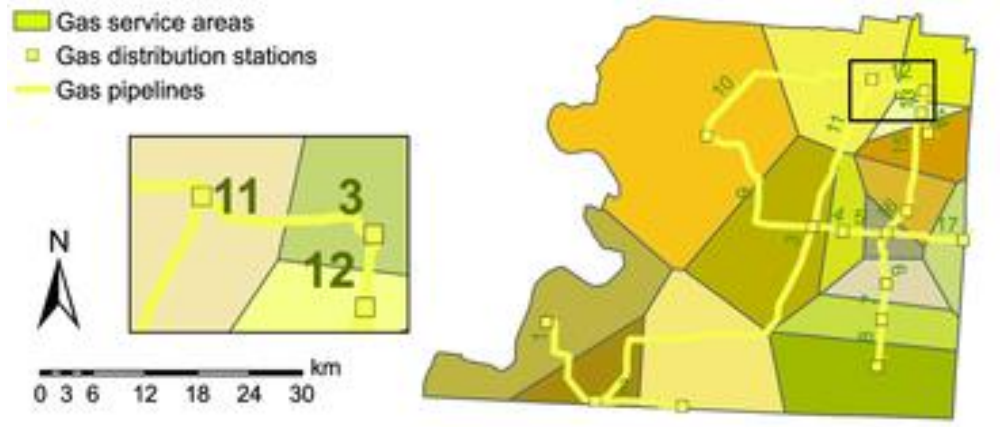

Figure 3: Gas network in Shelby County

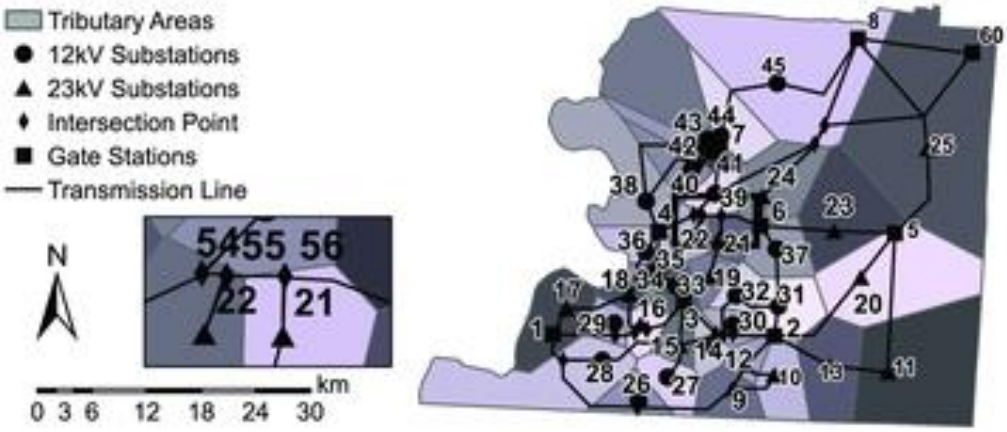

Figure 4: Power network in Shelby County

Table 1: Network properties of the Shelby County system

Network	Total Nodes	Supply Nodes	Demand Nodes	Total Links	Interdependent Links
Water	49	34	15	142	46
Gas	16	2	13	34	0
Power	60	37	23	152	46

4.2 Social Vulnerability by Block Group

The SoVI scores are calculated at the block group level for Shelby County using the SoVI-Lite methodology developed for the Mississippi Valley Region, in which the county resides. Due to the high margin of error in US Census data at the block group

level, the exact SoVI scores have a low degree of accuracy the therefore generalized categories of Low, Moderate, and High social vulnerability (ranks of 1,2 and 3, respectively), are used instead. These are found by standardizing the scores and assigning a rank of 1 to block groups for which $z \leq -0.5$, a rank of 2 to block groups for which $-0.5 < z < 0.5$ and a rank of 3 to block groups for which $z \geq 0.5$. The final distribution of social vulnerability in the county is 31% low, 47% moderate, and 22% high social vulnerability.

4.3 Numerical and Graphical Results

To be able to calculate weighted proportional slack, the model was run for each scenario with ϵ=1 so total vulnerability was allowed and no resources would be allocated. Table 2 shows this "no allocation" state for each scenario, where Total Cost is the sum of allocation costs (which have been forced to zero in this run) and of the costs of unmet demands, as shown in Eq. (25).

$$T = \sum_{k \in K} \sum_{i \in N^k} p_i^k s_i^k + \sum_{k \in K} \sum_{i \in N^k} q_i^k d_i^k + \sum_{k \in K} \sum_{i,j \in L^k} r_{ij}^k f_{ij}^k \tag{25}$$

Understandably, the desirability-based attacks results in the lowest total cost and slack because the components targeted may not have contributed significantly to meeting demand but instead are associated with costs of a less quantifiable nature. The total cost and slack for the capacity- and degree-based attacks are similar.

Table 2: Cost and slack in disrupted networks with no allocation.

Scenario	Total Cost	Total Slack	Slack: Water Network	Slack: Gas Network	Slack: Power Network
Desirability	$1,072,500	2145	357	1000	788
Capacity	$1,384,000	2768	546	1000	1222
Degree	$1,357,694	2715	921	586	1208

The Pareto-optimal frontier in Figure 5 shows the competing objective functions' values for each of the nine solutions found for each disruption scenario. As expected, the more money that goes toward allocation the lower the slack (lower vulnerability). While the capacity-based scenario has higher costs at the lowest values of weighted proportional slack, the costs are similar for values over approximately 0.25. Furthermore, the overall shape of the curves is very similar for each scenario; going from a slack of 1.0 to 0.50 does not require huge expenditure but from 0.10 to 0.0 the cost jump, indicating that the additional funds required may not be worth the slight decrease in slack.

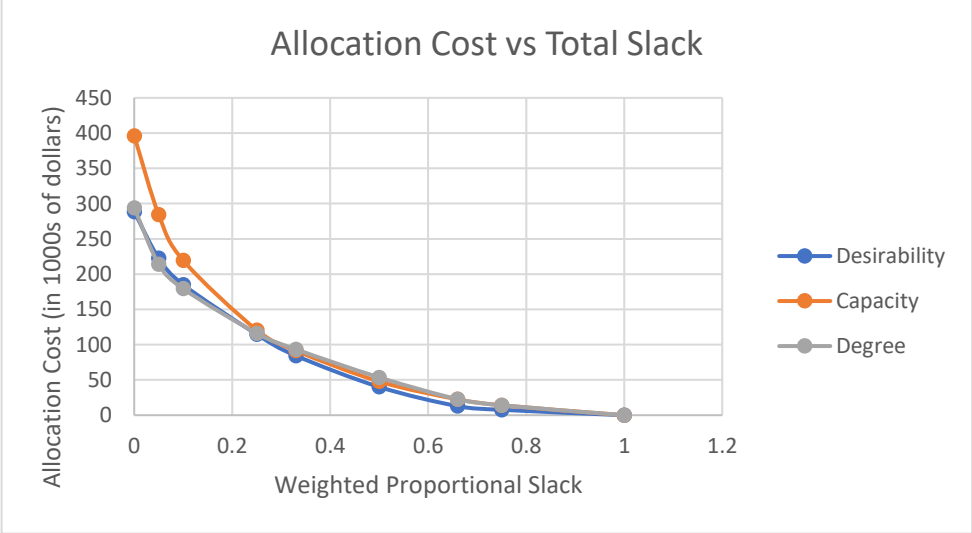

Figure 5: Pareto-optimal front for cost and vulnerability objectives

Once the model was run for each value of ϵ on each of the disruption scenarios, it was slightly reformatted so the solution values—the allocation amount at each component—were treated as inputs for the other two scenarios. This yielded the data seen in Figures 6-8. The desirability-based solutions appear to be largely ineffective in the capacity- and degree-based disruptions; except for the initial jump from "no allocation" to "some

allocation" between slack values of 1.0 and 0.85, the increase in allocation cost does not noticeably decrease the system vulnerability.

The capacity-based solutions offer a slight improvement over those for the desirability-based scenario. The weighted proportional slack for the degree-based scenario decreases more than that of the desirability-based scenario as costs increase, but the system remains between 40%-60% vulnerable. With the degree-based solutions, the desirability-based scenario is practically entirely unaffected, though the capacity-based scenario responds to the solutions almost as well as the degree-based scenario did with the capacity-based solutions. This indicates that there may be more overlap in the set of disrupted components between the capacity- and degree-based disruption scenarios than with the degree-based disruption scenarios.

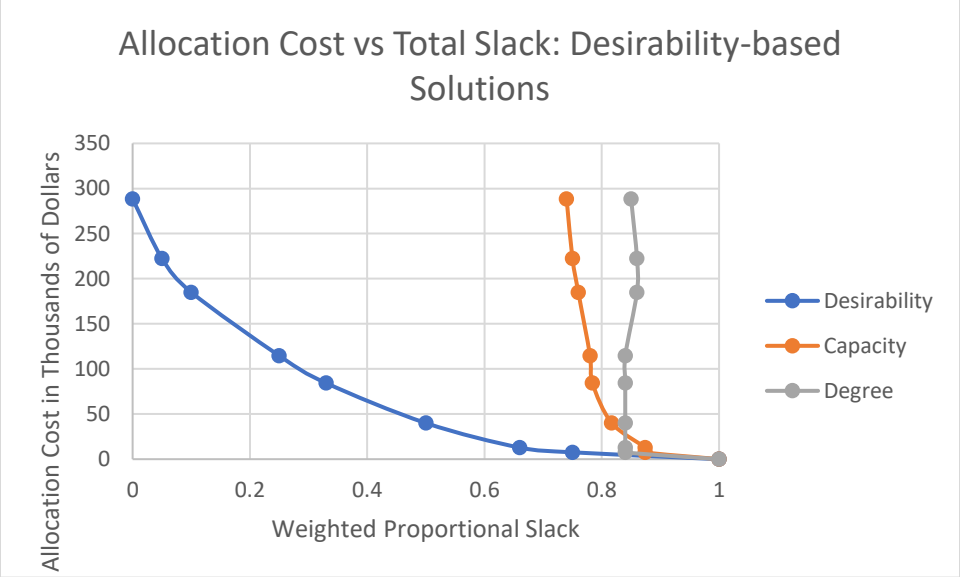

Figure 6: Robustness of desirability-based solutions on cost and vulnerability objectives

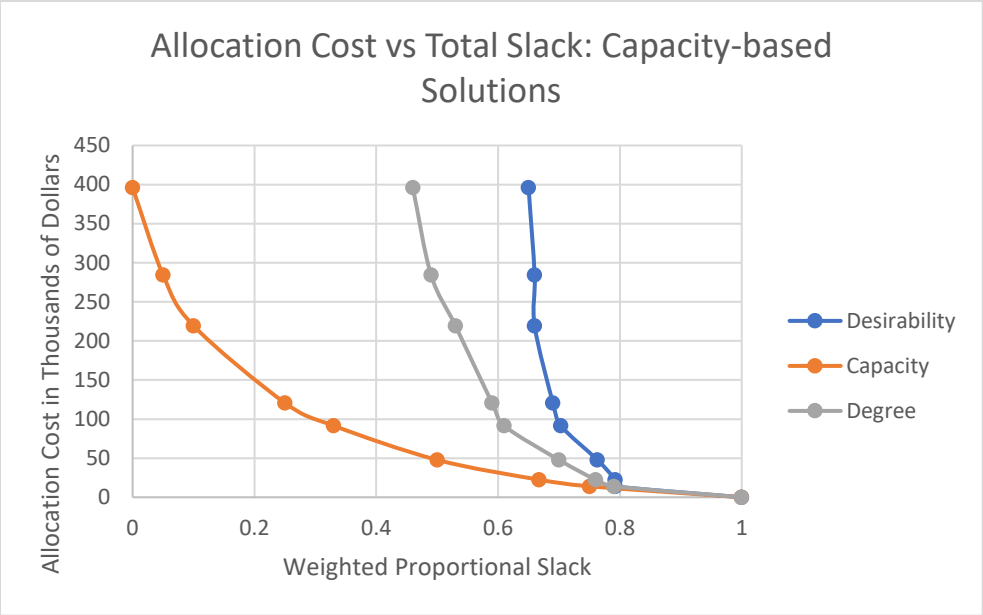

Figure 7: Robustness of capacity-based solutions on cost and vulnerability objectives

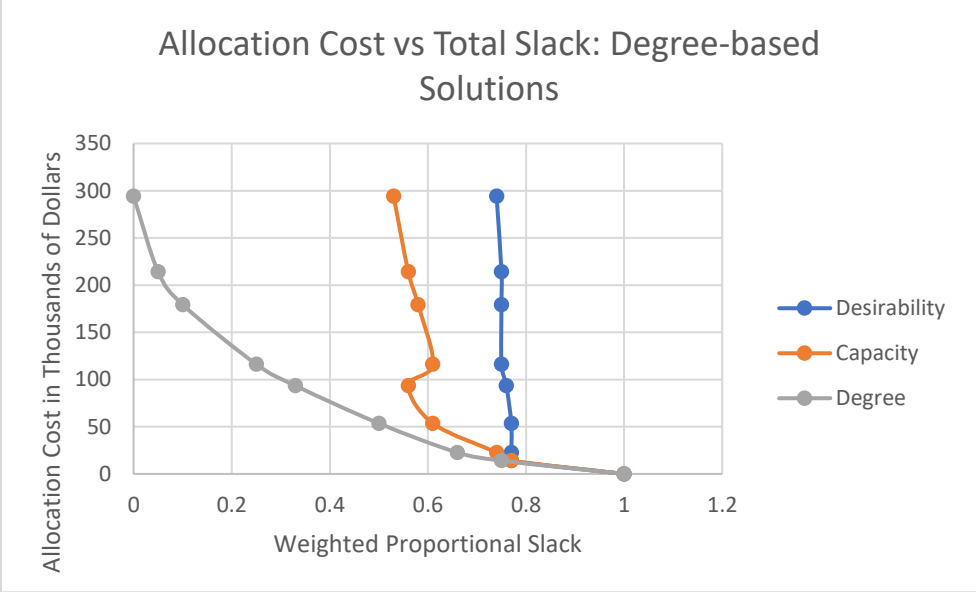

Figure 8: Robustness of degree-based solutions on cost and vulnerability objectives

4.4 Robustness Ranking

From the graphs above, capacity-based solutions seem to outperform degree-based solutions, and both are far better than the desirability-based solutions. This conclusion is confirmed in the TOPSIS evaluation in Table 3, as all the highest ranked defensive

strategies are from those two solution sets. However, none of the strategies are able to adequately decrease vulnerability across more than one disruption scenario. This can likely be attributed to high distinction between the sets of components targeted; even when the vulnerability is reduced to 0.10 or 0.05 for one scenario, indicating that nearly all the disrupted components have sufficient resources, the disrupted components for another scenario remain highly vulnerable.

In the TOPSIS results, the trade-off between reducing slack and reducing cost is clearly seen. For example, the Cap-6 strategy is ranked higher than Cap-7 strategy even though the slack is lower for both the capacity-based and degree-based scenarios. The allocation cost in the latter is too high to justify the relatively small decreases.

Table 3: Top eight solutions based on TOPSIS rankings; strategies are named by the scenario for and the run order in which they were originally solved

Strategy	**S+**	**Rank**	**Desirability-based Scenario**	**Capacity-based Scenario**	**Degree-based Scenario**	**Allocation Cost**
Cap – 6	0.458	1	0.66	0.10	0.53	$219,412
Cap – 5	0.443	2	0.69	0.25	0.59	$120,742
Cap – 7	0.439	3	0.66	0.05	0.49	$284,316
Deg – 6	0.437	4	0.75	0.58	0.10	$179,294
Deg – 7	0.437	5	0.75	0.56	0.05	$214,064
Cap – 4	0.421	6	0.70	0.33	0.61	$91,527
Deg – 4	0.418	7	0.76	0.56	0.33	$93,442
Deg – 5	0.415	8	0.75	0.61	0.25	$116,208

Chapter 5: Concluding Remarks

5.1 Discussion

In summary, the differences between the disruption scenarios made finding a truly robust solution difficult. Nevertheless, the methodology discussed in this work does provide a framework for comparing the effectiveness of allocation strategies for attacks on an interdependent network system. Additionally, it offers one way to include social vulnerability into disaster planning, a crucial component for which many previous works do not account. The resilience of both the infrastructure and of the community must be incorporated in disaster planning so that if a disruption should occur the system will be optimally fortified at its weakest points. While protecting critical infrastructure networks is undeniably important, it does little good if they are still susceptible to attack in the areas in which people most depend on them.

5.2 Limitations and Future Work

In the course of this research, multiple limitations and opportunities for continuing improvements and expansions were identified. First, the overall accuracy of the model can be increased by gaining a better understanding of the component attributes, such as the cost of resource allocation, understanding which factors are more important to attacker target selection, and the degree to which the network components have these factors. The latter two are obviously more complicated to achieve as they require deeper understanding into attacker psychology and motivation.

Secondly, using a non-linear contest function may have different results. A comparison of methods using the simplified contest function from this work and using the more common difference or ratio forms could help determine if the added complication from the non-linear function contributed to significantly better results.

Thirdly, weights based of off the likelihood of each attack scenario or the stakeholders' preferences for one criterion over another could be incorporated into the TOPSIS calculations using the Analytical Hierarchy Process or similar technique. For example, if defenders have information prior to an intentional attack that a degree-based attack is more likely than a desirability- or capacity-based attack, the weights used in TOPSIS could reflect this information and influence the higher ranked strategies to include those that best protected the network in degree-based disruption scenarios. Defenders may also consider vulnerability reduction more important than cost, or vice-versa. Even if the probability of a disruption is relatively low, there are non-monetary costs, like decreased sense of security in the public or loss of trust in the government, that would occur in the event of a successful malevolent attack. If defenders are more concerned with these qualitative costs than the quantitative costs of resource allocation, the vulnerability criteria can be weighted higher than the cost criterion to allow strategies with high costs but less damage to the networks to be ranked high. These changes would help represent the reality of the situation more accurately and could incorporate opinions of the community, governmental bodies, and infrastructure workers.

References

Almoghathawi, Y., Barker, K., & McLay, L. A. (2016). Resilience-driven restoration model for interdependent infrastructure networks. Submitted to the *European Journal of Operations Research*.

Baik, K.H. (1998). Difference-form contest success functions and effort levels in contests. *European Journal of Political Economy, 14*(4), 685-701.

Barker, K., J.E. Ramirez-Marquez, and C.M. Rocco. (2013). Resilience-based network component importance measures. *Reliability Engineering and System Safety*, 117(1): 89-97.

Cutter, S.L., Boruff, B.J., & Shirley, W.L. (2003). Social Vulnerability to environmental hazards. *Social Science Quarterly, 84*(2), 242-261.

Cutter, S. L., C. T. Emrich & D. Morath. 2011. Social vulnerability and place vulnerability analysis methods and application for Corps planning: Technical analyses. In Social Vulnerability Analysis Methods for Corps Planning, eds. C. M. Dunning & S. Durden, 74-88. Institute for Water Resources: U.S. Army Corps of Engineers.

González, A.D., Dueñas-Osorio, L., Sánchez-Silva, M., & Medaglia, A.L. (2015). The Interdependent Network Design Problem for Optimal Infrastructure System Restoration. *Computer Aided Civil and Infrastructure Engineering 31*(5), 334-350.

Haimes, Y.Y., L.S. Ladson, and D.A. Wismer. 1971. Bicriterion formulation of problems of integrated system identification and system optimization. *IEEE Transactions on Systems Man and Cybernetics*, **1**(3): 296-297.

Henry, D., and J.E. Ramirez-Marquez. 2012. Generic Metrics and Quantitative Approaches for System Resilience as a Function of Time. *Reliability Engineering and System Safety*, 99(1): 114-122.

Hirshleifer, J. (1989). Conflict and Rent-Seeking Success Functions: Ratio vs. Difference Models of Relative Success. *Public Choice, 63*(2), 101-112.

Hwang, C.-L., Lai, Y.-J., & Liu, T.-Y. (1993). A New Approach for Multiple Objective Decision Making. *Computers and Operations Research, 20*, 889-899.

Levitin, G. & Hausken, K. (2010). Resource Distribution in Multiple Attacks Against a Single Target. *Risk Analysis 30*(8), 1231-1239.

McCarter, M., Barker, K., Johansson, J., & Ramirez-Marquez, J.E. (2017). A Multi-objective formulation for robust defense strategies in multi-commodity networks. In first revision in *Reliability Engineering and System Safety*.

Miller, Erin. Terrorist Attacks Targeting Critical Infrastructure in the United States, 1970-2015. College Park, MD: START, 2016.

PDD 63 (Presidential Decision Directive 63). 1998. *The Clinton administration's policy on critical infrastructure protection: Presidential decision directive 63* available at http://www.terrorism.com/homeland/pdd63.htm.

Rinaldi, S.M., J.P. Peerenboom, and T.K. Kelly. 2001. Identifying, Understanding and Analyzing Critical Infrastructure Interdependencies. *IEEE Control Systems Magazine*, 21(6): 1125.

Toft, P., Duero, A., & Bieliauskas, A. (2010). Terrorist targeting and energy security. *Energy Policy*, *38*(8), 4411-4421.

Wang, S., Hong, L., Ouyang, M., Zhang, J., & Chen, X. (2013). Vulnerability Analysis of Interdependent Infrastructure Systems under Edge Attack Strategies. *Safety Science*, *51*(1), 328-337.

Appendix A Network Allocation Model

This section shows the code used to define the network attributes and solve the network allocation mathematical model. The data was read in from Excel and the model was written and solved with Python 2.7 and Gurobi 7.5.1.

```
# Resilience-Driven Allocation Model for Interdependent Infrastructure Networks
# Adapted from Almoghathawi et al, 2016
#Packages
import pandas as pd
import numpy as np
import math
from math import exp
from gurobipy import *
import xlsxwriter

# DATA:
nodes  = pd.DataFrame(pd.read_csv('Shelby full - nodes - Copy7.csv',header='infer'))
supply = pd.DataFrame(pd.read_csv('Shelby full - supply - Copy3.csv',header='infer'))
demand = pd.DataFrame(pd.read_csv('Shelby full - demand - Copy3.csv',header='infer'))
links  = pd.DataFrame(pd.read_csv('Shelby full - links - Copy7.csv', header='infer'))

dis_n=pd.DataFrame(pd.read_csv('Shelby full - nodes - Copy4 - dis.10.csv',header='inf
er') dis_l=pd.DataFrame(pd.read_csv('Shelby full - links - Copy4 - dis.10.csv',header
='infer'))depend
=pd.DataFrame(pd.read_csv('Shelby full - dependency - Copy1.csv',header='infer'))

#List of nodes
node = nodes['node'].values.ravel().tolist()
node = list(map(int,node))

#List of nodes and networks
IK =  list(zip(nodes['node'], nodes['net.n']))
nodes['IK']  = IK
Shelby_n_dic = nodes.set_index('IK').to_dict()
IK = [t for t in IK if not any(isinstance(n, float) and math.isnan(n) for n in t)] #n
ote to self: what does this line mean
IK = tuplelist(IK)

#List of links and networks
IJK =  list(zip(links['from'], links['to'], links['net.l']))
links['IJK'] = IJK
Shelby_l_dic = links.set_index('IJK').to_dict()
IJK = [t for t in IJK if not any(isinstance(n, float) and math.isnan(n) for n in t)]

IJK = tuplelist(IJK)

#List if interdependent nodes
IKJL =  list(zip(depend['node1'], depend['net1'], depend['node2'], depend['net2']))

depend['IKJL'] = IKJL
Shelby_d_dic   = depend.set_index('IKJL').to_dict()
IKJL = [t for t in IKJL if not any(isinstance(n, float) and math.isnan(n) for n in t)
]
IKJL = tuplelist(IKJL)
```

```
# SUPPLY AND DEMAND: ------------------------------------------------------------
#Supply nodes
IK_S =  list(zip(supply['node'], supply['net.n']))
supply['IK_S']  = IK_S
Shelby_sup_dic = supply.set_index('IK_S').to_dict()
IK_S = [t for t in IK_S if not any(isinstance(n, float)and math.isnan(n) for n in t)]
IK_S = tuplelist(IK_S)

#Demand nodes
IK_D =  list(zip(demand['node'], demand['net.n']))
demand['IK_D']  = IK_D
Shelby_dem_dic = demand.set_index('IK_D').to_dict()
IK_D = [t for t in IK_D if not any(isinstance(n, float)and math.isnan(n) for n in t)]
IK_D = tuplelist(IK_D)

# DISRUPTED COMPONENTS:----------------------------------------------------------
#Disrupted nodes
DIK =  list(zip(dis_n['node'], dis_n['net.n']))
dis_n['DIK']  = DIK
Shelby_dn_dic = dis_n.set_index('DIK').to_dict()
DIK = [t for t in DIK if not any(isinstance(n, float) and math.isnan(n) for n in t)]

DIK = tuplelist(DIK)

#Disrupted links
DIJK =  list(zip(dis_l['from'], dis_l['to'], dis_l['net.l']))  #list of links with th
eir network
dis_l['DIJK'] = DIJK
Shelby_dl_dic = dis_l.set_index('DIJK').to_dict()
DIJK = [t for t in DIJK if not any(isinstance(n, float)and math.isnan(n) for n in t)]
DIJK = tuplelist(DIJK)

#NODE AND LINK ATTRIBUTES:------------------------------------------------------
sup   = {} #Supply
dem   = {} #Demand
sovi  = {} #Social vulnerability rank
ext_n = {} #Target desirability of node
ext_l = {} #Target desirability of link
imp   = {} #Network weight
rc_n = {} #Unit restoration cost of node
rc_l = {} #Unit resoration cost of link
p     = {} #Unit cost of unmet demand
enode = {} #Attack resources dedicated to node
elink = {} #Attack resources dedicated to link
cap  = {} #Capacity of link

for i,k in IK_S:
      sup[i,k]  = Shelby_sup_dic['supply'][i,k]

for i,k in IK_D:
      dem[i,k]  = Shelby_dem_dic['demand'][i,k]
      imp[i,k]  = Shelby_dem_dic['imp'][i,k]
      sovi[i,k] = Shelby_dem_dic['sovi'][i,k]

for i,k in IK:
      p[i,k]     = Shelby_n_dic['p'][i,k]
      #fn[i,k]   = Shelby_n_dic['fn'][i,k]
      ext_n[i,k] = Shelby_n_dic['ext_n'][i,k]
      enode[i,k] = Shelby_n_dic['enode10'][i,k]
      rc_n[i,k] = Shelby_n_dic['rc_n'][i,k]
```

```
for i,j,k in IJK:
      cap[i,j,k]   = Shelby_l_dic['cap'][i,j,k]
      #cf[i,j,k]   = Shelby_l_dic['cf'][i,j,k]
      #fl[i,j,k]   = Shelby_l_dic['fl'][i,j,k]
      ext_l[i,j,k] = Shelby_l_dic['ext_l'][i,j,k]
      elink[i,j,k] = Shelby_l_dic['elink10'][i,j,k]
      rc_l[i,j,k] = Shelby_l_dic['rc_l'][i,j,k]

#Interdependent networks:
network = (1,2,3)

#Weighted slack in each disruption scenario with no allocation
TotWSlack=1444.56 #Desirability-based scenario
#TotWSlack=1837.44 #Capacity-based scenario
#TotWSlack=1956.03 #Degree-based scenario

m = Model('Resilience Optimization')

# VARIABLES:-----------------------------------------------------------------------
hnode={} #Defensive resource allocation for nodes - decision variable
for i,k in IK:
      hnode[i,k]=m.addVar(lb=0.0,vtype=GRB.CONTINUOUS,name='hnode_%s_%s'%(i,k))

hlink = {} #Defensive resource allocation for links - decision variable
for i,j,k in IJK:
      hlink[i,j,k] = m.addVar(lb=0.0,vtype=GRB.CONTINUOUS, name='hlink_%s_%s_%s'%(i,j
,k))

v_n = {} #Vulnerability of node given the proposed attack/defense scenario
for i,k in IK:
      v_n[i,k] = m.addVar(vtype=GRB.CONTINUOUS,name='v_n_%s_%s'%(i,k))

v_l = {} #Vulnerability of link given the proposed attack/defense scenario
for i,j,k in IJK:
      v_l[i,j,k] = m.addVar(vtype=GRB.CONTINUOUS,name='v_l_%s_%s_%s'%(i,j,k))

F = {} #Flow
for i,j,k  in IJK:
      F[i,j,k] = m.addVar(ub=cap[i,j,k], name='F_%s_%s_%s'%(i,j,k))

SU = {} #Unmet demand
for i,k in IK_D:
      SU[i,k] = m.addVar(ub=dem[i,k], name='SU_%s_%s'%(i,k))

SSU = {} #Total slack in each network
for k in network:
      SSU[k] = m.addVar(name='SSU_%s'%(k))

WS = {} #Total weighted slack in each network
for k in network:
      WS[k] = m.addVar(name='WS_%s'%(k))

m.update()

# CONSTRAINTS:---------------------------------------------------------------------
# 1. CONSERVATION CONSTRAINTS OF FLOW AT NODE i:
for i,k in IK_S:
      m.addConstr(quicksum(F[i,j,k] for i,j,k in IJK.select(i,'*',k)) -
                  quicksum(F[j,i,k] for j,i,k in IJK.select('*', i,k)) <= sup[i,k])
```

```
for i,k in IK_D:
      m.addConstr(quicksum(F[i,j,k] for i,j,k in IJK.select(i,'*',k)) -
            quicksum(F[j,i,k] for j,i,k in IJK.select('*', i,k)) - SU[i,k] == -
dem[i,k])

# 2. CAPACITY CONSTRAINTS ON LINK (I,J):
for i,k in DIK:
      m.addConstr(v_n[i,k] == ((enode[i,k]-hnode[i,k])/enode[i,k]))

for i,j,k in DIJK:
      m.addConstr(v_l[i,j,k] == ((elink[i,j,k]-hlink[i,j,k])/elink[i,j,k]))

for i,j,k in IJK:
    m.addConstr(F[i,j,k] <= cap[i,j,k] * (1-v_n[i,k]))

for i,j,k in IJK:
    m.addConstr(F[i,j,k] <= cap[i,j,k] * (1-v_n[j,k]))

for i,j,k in DIJK:
    m.addConstr(F[i,j,k] <= cap[i,j,k] * (1-v_l[i,j,k]))


# 3. INTERDEPENDENCE CONSTRAINST:
for i,k,j,l in IKJL:
      m.addConstr(v_n[i,k] >= v_n[j,l])

#Total slack in each network
for k in network:
      m.addConstr(SSU[k] == quicksum(SU[i,k] for i,k in IK_D.select('*',k)))

#Weighted slack in each network
for k in network:
      WS[k] = quicksum(sovi[i,k]*imp[i,k]*SU[i,k] for i,k in IK_D.select('*',k))

# OBJECTIVE FUNCTION:  ------------------------------------------------------
#OBJECTIVE I: MINIMIZE SLACK:
S_total=(quicksum(sovi[i,k]*imp[i,k]*SU[i,k] for i,k in IK_D))/TotWSlack #Proportiona
l weighted slack
epsilon=1
m.addConstr(S_total <= epsilon) #Epsilon constraint

m.update()

# OBJECTIVE II: MINIMIZE THE ALLOCATION COST:
NRC = quicksum( rc_n[i,k]*hnode[i,k] for i,k in DIK)   #Node resource allocation cost
LRC = quicksum( rc_l[i,j,k]*hlink[i,j,k] for i,j,k in DIJK) #Link resource allocation
 cost
UDC = quicksum( p[i,k] * SU[i,k] for i,k in IK_D)  #Unmet demand cost at demand nodes
Alloc=NRC+LRC  #Total allocation costs
COST=NRC+LRC+UDC #Total allocation and slack costs

m.update()

m.setObjective(Alloc, GRB.MINIMIZE)

m.modelSense = GRB.MINIMIZE
m.optimize()
status = m.status

m.write('Resilience Optimization.lp')
```

```
if m.status == GRB.Status.OPTIMAL:
    print('Optimal cost objective: %g' % m.objVal)
elif m.status == GRB.Status.INF_OR_UNBD:
    print('Model is infeasible or unbounded')
    exit(0)
elif m.status == GRB.Status.INFEASIBLE:
    print('Model is infeasible')
    exit(0)
elif m.status == GRB.Status.UNBOUNDED:
    print('Model is unbounded')
    exit(0)
else:
    print('Optimization ended with status %d' % m.status)
    exit(0)

solution  = m.getAttr('x',SSU)

for k in network:
        if solution[k]>0:
              print('SSU[%s] = %g'%(k,solution[k]))

print 'The runtime is'
print m.Runtime
print 'The total cost is'
print COST.getValue()
print 'The total allocation cost is'
print Alloc.getValue()
print 'The total weighted proportional slack is'
print S_total.getValue()
print 'The total unmet demand is'
print SSU[1]
print SSU[2]
print SSU[3]
print 'The weighted slack in each network is'
print WS[1].getValue()
print WS[2].getValue()
print WS[3].getValue()

#Write out current solution to file
node_i=[]
node_k=[]
node_h=[]
for i,k in IK:
      node_i.append(i)
      node_k.append(k)
      node_h.append(hnode[i,k].X)
nodedata={'Node':node_i,'Network':node_k,'hnode':node_h}
dataframe=pd.DataFrame(nodedata,columns=['Node','Network','hnode'])
writer=pd.ExcelWriter('hnode.xlsx',engine='xlsxwriter')
dataframe.to_excel(writer,sheet_name='Sheet1')
writer.save()

link_i=[]
link_j=[]
link_k=[]
link_h=[]
for i,j,k in IJK:
      link_i.append(i)
      link_j.append(j)
      link_k.append(k)
      link_h.append(hlink[i,j,k].X)
```

```
linkdata={'Node1':link_i,'Node2':link_j,'Network':link_k,'hlink':link_h}
dataframe=pd.DataFrame(linkdata,columns=['Node1','Node2','Network','hlink'])
writer=pd.ExcelWriter('hlink.xlsx',engine='xlsxwriter')
dataframe.to_excel(writer,sheet_name='Sheet1')
writer.save()

print('The optimal objective is %g' % m.objVal)
```

Appendix B Network Allocation Model for Robustness

This section shows the code from Appendix A rewritten to apply the found solutions to the other two scenarios for which it was not originally solved. Only the sections of code that required changes are shown here.

```
# NODE AND LINK ATTRIBUTES:-------------------------------------------------------
sup   = {} #supply at node i
dem   = {} #demand at node i
sovi  = {} #social vulnerability score at node i
ext_n = {} #external importance at node i (importance to attacker)
ext_l = {} #external importance at link i,j (importance to attacker)
imp   = {} #network importance for node i (importance to defender)
rc_n = {} #unit restoration cost for nodes
rc_l = {} #unit resoration cost for links
p     = {} #unit cost of unmet demand, slack cost
enode = {} #attack resources dedicated to node i
elink = {} #attack resources dedicated to link i
cap  = {} #Capacity of the links

#hnode and hlink are read in as attributes and are not solved as decision variables
hnode = {} #defense resources dedicated to node i
hlink = {} #defense resources dedicated to link i

for i,k in IK_S:
      sup[i,k]  = Shelby_sup_dic['supply'][i,k]

for i,k in IK_D:
      dem[i,k]  = Shelby_dem_dic['demand'][i,k]
      imp[i,k]  = Shelby_dem_dic['imp'][i,k]
      sovi[i,k] = Shelby_dem_dic['sovi'][i,k]

for i,k in IK:
      p[i,k]     = Shelby_n_dic['p'][i,k]
      ext_n[i,k] = Shelby_n_dic['ext_n'][i,k]
      enode[i,k] = Shelby_n_dic['enode11'][i,k] #change based on dis#
      hnode[i,k] = Shelby_n_dic['hnode128'][i,k] #change based on dis#
      rc_n[i,k] = Shelby_n_dic['rc_n'][i,k]

for i,j,k in IJK:
      cap[i,j,k]   = Shelby_l_dic['cap'][i,j,k]
      ext_l[i,j,k] = Shelby_l_dic['ext_l'][i,j,k]
      elink[i,j,k] = Shelby_l_dic['elink11'][i,j,k] #change based on dis#
      hlink[i,j,k] = Shelby_l_dic['hlink128'][i,j,k] #change based on dis#
      rc_l[i,j,k] = Shelby_l_dic['rc_l'][i,j,k]

#Interdependent networks:
network = (1,2,3)

#Weighted slack in each disruption scenario with no allocation
TotWSlack=1444.56 #Desirability-based scenario
#TotWSlack=1837.44 #Capacity-based scenario
#TotWSlack=1956.03 #Degree-based scenario

m = Model('Resilience Optimization')

# VARIABLES:----------------------------------------------------------------------
```

```
v_n = {} #Vulnerability of node given the proposed attack/defense scenario
for i,k in IK:
      v_n[i,k] = m.addVar(vtype=GRB.CONTINUOUS,lb=0,ub=1,name='v_n_%s_%s'%(i,k))

v_l = {} #Vulnerability of link given the proposed attack/defense scenario
for i,j,k in IJK:
      v_l[i,j,k]=m.addVar(vtype=GRB.CONTINUOUS,lb=0,ub=1,name='v_l_%s_%s_%s'%(i,j,k))

F = {} #Flow
for i,j,k  in IJK:
      F[i,j,k] = m.addVar(ub=cap[i,j,k], name='F_%s_%s_%s'%(i,j,k))

SU = {} #Unmet demand
for i,k in IK_D:
      SU[i,k] = m.addVar(ub=dem[i,k], name='SU_%s_%s'%(i,k))

SSU = {} #Total slack in each network
for k in network:
      SSU[k] = m.addVar(name='SSU_%s'%(k))

WS = {} #Total weighted slack in each network
for k in network:
      WS[k] = m.addVar(name='WS_%s'%(k))

m.update()

#CONSTRAINTS:----------------------------------------------------------------------
#1. CONSERVATION CONSTRAINTS OF FLOW AT NODE i:
for i,k in IK_S:
      m.addConstr(quicksum(F[i,j,k] for i,j,k in IJK.select(i,'*',k)) -
                  quicksum(F[j,i,k] for j,i,k in IJK.select('*', i,k)) <= sup[i,k])

for i,k in IK_D:
      m.addConstr(quicksum(F[i,j,k] for i,j,k in IJK.select(i,'*',k)) -
            quicksum(F[j,i,k] for j,i,k in IJK.select('*', i,k)) - SU[i,k] == -
dem[i,k])

# 2. CAPACITY CONSTRAINTS ON LINK (I,J):
for i,j,k in IJK:
      m.addConstr(F[i,j,k]<= cap[i,j,k])

for i,k in DIK:
      m.addConstr(v_n[i,k] >= ((enode[i,k]-hnode[i,k])/enode[i,k]))

for i,j,k in DIJK:
      m.addConstr(v_l[i,j,k] >= ((elink[i,j,k]-hlink[i,j,k])/elink[i,j,k]))

for i,j,k in IJK:
    m.addConstr(F[i,j,k] <= cap[i,j,k] * (1-v_n[i,k])) #flow on link ij< (13-2)


for i,j,k in IJK:
    m.addConstr(F[i,j,k] <= cap[i,j,k] * (1-v_n[j,k])) #flow on link ij<(14-2)


for i,j,k in DIJK:
    m.addConstr(F[i,j,k] <= cap[i,j,k] * (1-v_l[i,j,k])) #(15-2)

# 3. INTERDEPENDENCE CONSTRAINST:
for i,k,j,l in IKJL:
      m.addConstr(v_n[i,k] >= v_n[j,l])
```

```
#Total slack in each network
for k in network:
      m.addConstr(SSU[k] == quicksum(SU[i,k] for i,k in IK_D.select('*',k)))

#Weighted slack in each network
for k in network:
      WS[k] = quicksum(sovi[i,k]*imp[i,k]*SU[i,k] for i,k in IK_D.select('*',k))

#OBJECTIVE FUNCTION: MINIMIZE SLACK:
#Proportional weighted slack
S_total=(quicksum(sovi[i,k]*imp[i,k]*SU[i,k] for i,k in IK_D))/TotWSlack
m.update()

NRC = quicksum( rc_n[i,k]*hnode[i,k] for i,k in IK)  #Node resource allocation cost
LRC = quicksum( rc_l[i,j,k]*hlink[i,j,k]
for i,j,k in IJK) #Link resource allocation cost
UDC = quicksum( p[i,k] * SU[i,k] for i,k in IK_D)  #Unmet demand cost at demand nodes
Alloc=NRC+LRC #Total allocation costs
COST=NRC+LRC+UDC #Total allocation and slack costs

m.update()

m.setObjective(S_total, GRB.MINIMIZE)

m.modelSense = GRB.MINIMIZE
m.optimize()
status = m.status

m.write('Resilience Optimization.lp')

if m.status == GRB.Status.OPTIMAL:
    print('Optimal cost objective: %g' % m.objVal)
elif m.status == GRB.Status.INF_OR_UNBD:
    print('Model is infeasible or unbounded')
    exit(0)
elif m.status == GRB.Status.INFEASIBLE:
    print('Model is infeasible')
    exit(0)
elif m.status == GRB.Status.UNBOUNDED:
    print('Model is unbounded')
    exit(0)
else:
    print('Optimization ended with status %d' % m.status)
    exit(0)

solution  = m.getAttr('x',SSU)

for k in network:
        if solution[k]>0:
              print('SSU[%s] = %g'%(k,solution[k]))

print 'The runtime is'
print m.Runtime
print 'the total cost is'
print COST.getValue()
print 'the total allocation cost is'
print Alloc
print 'The total weighted proportional slack is'
print S_total.getValue()
print 'The total unmet demand is'
```

```
print SSU[1]
print SSU[2]
print SSU[3]
print 'The weighted slack in each network is'
print WS[1].getValue()
print WS[2].getValue()
print WS[3].getValue()
print('The optimal objective is %g' % m.objVal)
```