UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

# OPTIMAL FACILITY LOCATION FOR INTERDEPENDENT INFRASTRUCTURE NETWORK RECOVERY

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

ELYSSA MOONEY
Norman, Oklahoma
2017

OPTIMAL FACILITY LOCATION FOR INTERDEPENDENT INFRASTRUCTURE NETWORK RECOVERY

A THESIS APPROVED FOR THE
SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

BY

______________________________
Dr. Kash Barker, Chair

______________________________
Dr. Charles Nicholson

______________________________
Dr. Shivakumar Raman

To my family and friends,
For all your love and support.

## Acknowledgements

First and foremost, I would like to thank Yasser Almoghathawi for not only providing me with the data and initial model used in my thesis, but also for your expert guidance throughout the last two semesters. I consider Yasser to be a significant mentor through this process and could always count on him to bounce ideas off of. I cannot express how much I truly appreciate his help.

Second, I would like to thank Nazanin Morshedlou for always sharing your stress snacks with me and encouraging me throughout this year. Nazanin is also a non-linearity queen, so thank you for helping me transform non-linear functions to be linear again.

I would also like to thank my advisor, Dr. Kash Barker for not only encouraging me to attend graduate school and write a thesis in the first place, but for always being available when I felt like I had no idea what I was doing. If Kash couldn't answer a question or resolve issue I had throughout this process, he was always able to direct me to someone who could.

Finally, I would like the thank all the other professors in the School of Industrial and Systems Engineering at OU and friends I've made in the department. Thank you to my professors for always keeping my best interest in mind and encouraging me to perform to the best of my ability. Thank you to all of my friends in the ISE department; I truly could not have done this without your loving support.

## Table of Contents


Acknowledgements ........ iv

List of Figures ........ vii

Abstract ........ viii

Chapter 1.0 Introduction ........ 1

1.1 Motivation ........ 1

1.2 Critical Infrastructure Networks ........ 1

Chapter 2.0 Literature Review ........ 4

2.1 Interdependent Infrastructure Restoration ........ 4

2.2 Facilities Location ........ 6

Chapter 3.0 Methodological Background ........ 8

3.1 Network Resilience ........ 8

3.2 Disruption Scenarios ........ 9

Chapter 4.0 Proposed Model ........ 11

4.1 Resilience-Driven Recovery of Interdependent Infrastructure Networks ........ 11

4.1.1 Mathematical Model ........ 13

4.2 Facilities Location and Work Crew Assignment ........ 17

4.2.1 Addressing Non-Linearity of Cost Objective ........ 19

4.2.2 Model Simplification ........ 19

Chapter 5.0 Illustrative Example ........ 21

5.1 Data ........ 21

5.2 Example ........ 23

5.2.1 Pareto-Optimal Solutions ........ 23

5.2.2 Assessment of Varying the Number of Established Facilities ........ 24

Chapter 6.0 Concluding Remarks ........ 30

6.1 Summary .......... 30

6.2 Future work .......... 30

References .......... 32

Appendix A Network Generation .......... 34

Appendix B Network Restoration Model with Location Selection .......... 42

Appendix C Locations Selected .......... 47

Appendix C.1 Capacity-based Disruption Scenario .......... 47

Appendix C.1.1 1 Facility .......... 47

Appendix C.1.2 2 Facilities .......... 48

Appendix C.1.3 3 Facilities .......... 48

Appendix C.2 Degree-based Disruption Scenario .......... 49

Appendix C.2.1 1 Facility .......... 49

Appendix C.2.2 2 Facilities .......... 50

Appendix C.2.3 3 Facilities .......... 50

Appendix C.3 Random Disruption Scenario .......... 51

Appendix C.3.1 1 Facility .......... 51

Appendix C.3.2 2 Facilities .......... 52

Appendix C.3.3 3 Facilities .......... 52

Appendix C.4 Spatial Disruption Scenario .......... 53

Appendix C.4.1 1 Facility .......... 53

Appendix C.4.2 2 Facilities .......... 54

Appendix C.4.3 3 Facilities .......... 54

## List of Figures

Figure 1: Phases of network resilience as a function of time .................... 9
Figure 2: Interdependent Network Graph with 25 Candidate sites .................... 22
Figure 3: Pareto-optimal frontier for competing objectives cost and resilience .................... 24
Figure 4: Resilience vs. time for a capacity-based disruption for (a) power network and (b) water network considering a different number of established facilities .................... 25
Figure 5: Resilience vs. time for a degree-based disruption for (a) power network and (b) water network considering a different number of established facilities .................... 26
Figure 6: Resilience vs. time for a spatial disruption for (a) power network and (b) water network considering a different number of established facilities .................... 27
Figure 7: Resilience vs. time for a random disruption for (a) power network and (b) water network considering a different number of established facilities .................... 28

# Abstract

Modern communities heavily depend on critical infrastructure networks such as power, water, transportation, telecommunications, gas, etc. Since daily life requires these networks to be operational, it is important that they are able to withstand or recover quickly from a disruption, a term known as resilience. These infrastructure networks are often dependent on each other for operation. The interdependency of infrastructure networks makes them more vulnerable to disruptive events such as malevolent attacks, natural disasters, and random failures. The operability of these networks may be compromised following a disruptive event such that demand in any given network is not met. To return the networks to some desired level of resilience, work crews must be scheduled to restore certain disrupted elements. The proposed model is a multi-objective mixed-integer programming model that seeks to minimize the total cost of restoration while maximizing the combined resilience of interdependent infrastructure networks. The model may be used to determine where each work crew should originate from following a disruptive event as well as schedule the work crews to restore disrupted network elements over a finite time horizon. This work demonstrates the use of the model through an illustrative example of two interdependent infrastructure networks. Considering four disruption scenarios, this illustrative example shows how recovery may change by varying the number of facilities established for work crews in each network.

# Chapter 1.0 Introduction

## 1.1 Motivation

Following the attack on the World Trade Center in New York City on September 11, 2001, government officials began focusing on the assistance provided following a disastrous event. In August 2006, President George W. Bush issued an executive order to improve the quality of response to non-routine disasters including malevolent attacks, natural disasters, and other failures (Exec. Order No. 13411, 2006). Since 9/11 there have been several natural disasters and terrorist threats that have compromised our nation's infrastructure. These events have evoked the issuing of the Presidential Policy Directive (PPD) on Critical Infrastructure Security and Resilience in 2013 and other similar governmental actions. The PPD recognizes the federal responsibility toward maintaining "secure, functioning, and resilient critical infrastructure (Executive Office of the President, 2013)."

## 1.2 Critical Infrastructure Networks

Critical infrastructure networks, as defined by the USA PATRIOT Act of 2001, are "systems and assets, whether physical or virtual, so vital to the United States that the incapacity or destruction of such systems and assets would have a debilitating impact on security, national economic security, national public health or safety, or any combination of those matters (United States. Cong., 2001)." More explicitly, a critical infrastructure network is used to distribute resources that are necessary for our daily lives. Common examples of critical infrastructure networks are water, electricity, gas, communications, and transportation. Anything that causes these networks to be inoperable impacts the health and safety of the affected community.

As daily life has become more dependent on critical infrastructure, it is increasingly important to not only protect current infrastructure networks, but to be able to rebuild portions of a network that have been disrupted. Infrastructure resilience, specifically, "depends upon its ability to anticipate, absorb, adapt to, and/or rapidly recover from a potentially disruptive event (National Infrastructure Advisory Council, 2009)." The Department of Homeland Security has focused largely on resilience in terms of withstanding and recovering from deliberate attacks that may affect multiple critical infrastructures networks.

Not only do critical infrastructure networks influence society, but also affect the operation and resilience of other critical infrastructure networks. Interdependency is defined as "a bidirectional relationship between two infrastructures through which the state of each infrastructure influences or is correlated to the state of the other (Rinaldi et al., 2001)." The interconnectedness of critical infrastructures is becoming increasingly prevalent and complex. Interdependencies of infrastructure networks may cause them to be more vulnerable to a disruptive event. If a disruption compromises the operability of a certain network, the functionality of any dependent network may also be affected. Decision makers must take this into consideration when recovering after an extreme event affecting multiple critical infrastructures. An increase in interdependency also increases the complexity of planning for recovery.

The goal of this work is to help decision makers plan for recovery after a disruptive event; it addresses the restoration of interdependent infrastructure networks by solving a facilities location model to determine where work crews should be stationed following a disruption and scheduling those work crews to repair disrupted elements to attain a desired level of resilience.

The remainder of this paper is organized as follows: Chapter 2 considers literature relevant to the problem of facilities location and interdependent infrastructure recovery. Chapter 3 gives background to resilience-driven methodology. Chapter 4 details the proposed model. The proposed model will be illustrated in an example using a set of two interdependent infrastructure networks in Chapter 5. Concluding remarks and future work will be addressed in Chapter 6.

## Chapter 2.0 Literature Review

This chapter discusses the literature critical to developing the methodology addressed in the remainder of the paper.

### 2.1 Interdependent Infrastructure Restoration

There has been significant research in the area of critical infrastructure restoration. This work focuses primarily on the restoration of interdependent critical infrastructure networks. Lee et al. (2007) recognizes the complexities involved in the interdependency of critical infrastructure networks. The authors also define multiple types of interdependence including mutual dependence. Mutual dependence may be described by a scenario where all networks in a set of critical infrastructure networks require the output of another network to be operational. This work primarily proposes a model to restore disrupted elements in a set of interdependent infrastructure networks by minimizing the cost associated with unmet demand. The model, however, does not consider the cost associated with the restoration process and it is not time dependent. As a result, there is no fixed restoration time associated with disrupted components. Moreover, the model does not associate work crews with the restoration process and, therefore, cannot schedule specific work crews to restore the disrupted elements.

Gong et al. (2009) on the other hand, proposes a multi-objective optimization model to schedule emergency work crews for restoration of interdependent networks. This model assumes that all restoration tasks have a defined due time and is thus a time-dependent model. The objectives are to minimize the cost, time to restoration, and delay in restoration time. The main purpose of this work is to consider when each task should be completed and create a schedule for restoration. The model also assigns available work crews to each restoration task. Although this work does not actually restore the disrupted elements, the scheduling of work crews and associating time with restoration tasks are important contributions.

Cavdaroglu et al. (2013) combine the work of Lee et al. (2007) and Gong et al. (2009) by specifically accounting for the interdependencies that exist between critical infrastructure networks. An important consideration is that the operability of one network is dependent on the functionality of certain elements in another; further, any change to one network may affect another, whether positive or negative. This work specifically uses a network flow model to determine which disrupted elements should be restored, create a schedule for restoration, and assign restoration to tasks to available work crews. The objective is to minimize the total cost including flow cost, restoration cost, and cost of unmet demand.

Almoghathawi et al. (2016) propose a recovery model for interdependent infrastructure much like Cavdaoglu et al. However, instead of a single-objective mixed-integer programming model, the authors propose a multi-objective model. Here the objectives are to minimize the total cost of restoration (i.e. fixed restoration costs, flow cost, and cost of unmet demand) while also maximizing the combined resilience of the interdependent infrastructure network system. It is recognized by Almoghathawi et al. that the resilience of one network is dependent on another due to the interdependencies between them; these interdependencies are considered bi-directional, meaning the output of each network is dependent on the output of another. In this work 4 disruption scenarios are modeled: 2 malevolent attack scenarios, a random failure scenario, and a spatial disruption scenario. For each scenario, disrupted elements must be scheduled to be restored. Restoration processes are time-dependent and require a work crew for completion. Thus, the model also model accounts for the availability of work crews during each time period and schedules them to restoration processes.

**2.2 Facilities Location**

Part of the proposed model to be discussed in Chapter 4 is to determine where work crews in each network should dispatch from. As a result, the following literature discusses facilities location models in the context of emergency response.

Batta and Mannur (1990) propose a model for emergency response related to the service coverage of a given set of demand points. The objective of the mdoel is to maximize the coverage provided by $M$ facilites. In this model, $M$ is a parameter that must be defined by decision makers. The value of $M$ may also be defined by the number of work crews available to service a given network. An important finding of this work is that positioning more than one work crew at any given facility does not improve the objective of maximizing service coverage for a given set of demand points.

Jia et al. (2007) address the problem of facilities location for emergency response to large-scale problems. Specifically, the authors propose a facility location that may be used in the event of a terrorist attack or natural disaster: events that most first responders are not regularly accustomed to. An important contribution from this work is that quality of service from a facility is dependent on its proximity to a demand point. Thus, the closer a facility is to a demand location, the better that facility may be.

Afacan and McLay (2016) propose a model for emergency response specific to the context of critical infrastructure recovery. This work discusses the interdependency of critical infrastructures and emergency responders. Following a disruptive event, portions of critical infrastructure networks become unusable, which makes the job of first responders much more difficult. The model thus accounts for work crews restoring disrupted network elements and solves a P-median facilities location model for the dispatch of those work crews. In addition to dispatching work crews, the model also schedules network recovery by assigning work crews to restoration tasks over a finite time horizon. Multiple case studies are examined in the work to fully assess the usability of the model. Interdependency is considered in this work, but in the context of work crews and first responders rather than the interdependency of infrastructure.

Although there has been significant work done in the areas of interdependent infrastructure restoration and facilities location models for emergency response, there is a gap in literature to combine to the two topics. This work develops a model to determine where work crews should be stationed in an emergency such that a set of critical infrastructure networks are able to return to full resilience.

# Chapter 3.0 Methodological Background

This section gives a background on the framework of resilience and describes how resilience is quantified. There are 4 disruption scenarios (malevolent attacks--degree-based and capacity-based, random failures, and spatial failures) introduced in this section as well.

## 3.1 Network Resilience

Resilience in literature has been quantified in a variety of ways. In this work, resilience is considered the performance of a system of networks before, during, and after a disruptive event. The framework of network resilience is adapted from Henry and Ramirez-Marquez (2012) and shown in Figure 1. This framework focuses on two phases of system performance: vulnerability and recoverability. Vulnerability is considered the susceptibility of system components to a disruptive event (Jönsson et al., 2008). Initially, the system exists at some stable performance level. At the time of a disruption, though, the system becomes vulnerable to failures, leading to a loss in system performance. At this point, the system exists at a disrupted state until it can be recovered. The recoverability of a system is defined by how quickly it can be restored to some desired level of resilience after a disruption (Rose, 2007). As the system is recovered, its performance is improved until it reaches a desired state of recovery. These phases of system performance are depicted in Figure 1.

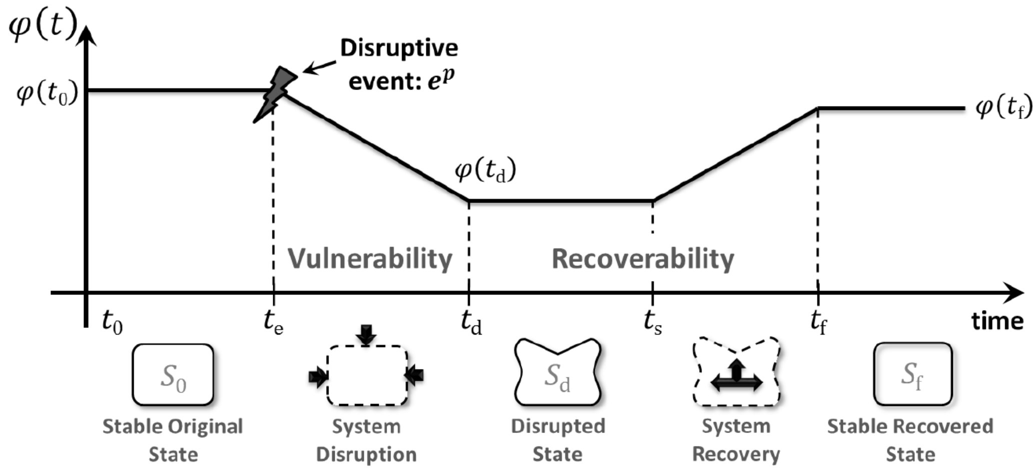



Figure 1: Network performance, $\varphi(t)$, as a function of time

System performance as a function of time, $\varphi(t)$, may be used to mathematically represent resilience. Network resilience, Я, may be described as the ratio of time-dependent recovery to loss (i.e. $Я(t) = \frac{Recovery(t)}{Loss(t)}$) (Henry and Ramirez-Marquez, 2012). This relationship is defined more explicitly in Equation 1.

$$Я_{\varphi}(t|e^j) = \frac{\varphi(t|e^j) - \varphi(t_d|e^j)}{\varphi(t_0) - \varphi(t_d|e^j)}, \quad \forall\, t \in (t_s, t_f) \tag{1}$$

$\varphi(t|e^j)$ is the system performance at time $t$ following disruptive event $e^j$, $\varphi(t_d|e^j)$ is the system performance immediately following a disruption, and $\varphi(t_0)$ is the system performance prior to a disruption. $Я_{\varphi}(t|e^j)$ may range between 0 and 1, where 1 means the system is fully resilient.

### 3.2 Disruption Scenarios

Interdependent infrastructure networks are susceptible to several different types of disruptions. These disruptions may be divided in to 3 distinct groups: malevolent attacks, random failures, and spatial failures (Wang et al., 2013). Malevolent attacks are intentional disruptions caused as an act of terrorism against specific infrastructure components. These attacks can be separated into 2 groups: capacity-based attacks and degree-based attacks. In a capacity-based attack, network elements with higher capacity are targeted. A link's capacity is defined as a network parameter, $u_{ij}^k$, and indicates the maximum flow across each link. The capacity of a node is defined by the minimum of the sum of incoming and outgoing links associated with that node ($u_i^k = \min\,(\sum_{(j,i)\epsilon L^k} u_{ji}^k, \sum_{(i,l)\epsilon L^k} u_{il}^k)$. In a degree-based attack, network elements with the greatest connectivity to other network elements (i.e. highest degree) are targeted. The degree of a node is defined by the number of bi-directional links that are connected to it; the degree of a link is the average of the degree of the two nodes it connects (i.e. $degree_{ij} = \frac{1}{2}(degree_i + degree_j)$). A random disruption may include any man-made failure, a failure due to the age of a network component, etc. In this work, all network components are considered to have an equal probability of failure for random disruptions. Finally, all natural disasters such as hurricanes, earthquakes, or any other failure related to the physical location of network components are captured by the spatial disruption scenario (Almoghathawi et al., 2016).

# Chapter 4.0 Proposed Model

In this section, the methodology for the proposed model will be discussed. Previous work by Almoghathawi et al. (2016) will first be described, followed by the proposed additions to define work crew placement.

## 4.1 Resilience-Driven Recovery of Interdependent Infrastructure Networks

Almoghathawi et al. (2016) describes a model that may be used to restore a set of disrupted elements in interdependent infrastructure networks. The model is formulated as a mixed-integer programming (MIP) model. It is a multi-objective optimization model with competing objectives. Generally, the objectives are to maximize resilience while minimizing the total restoration cost. There are three sets of constraints: network flow constraints, interdependency constraints, and assignment and scheduling constraints for work crews.

This model contains a set of networks, $K$, and a set of time periods, $T$. In each network $k \in K$ there is a set of nodes, $N^k$, and a set of links between nodes, $L^k$. Each network $k \in K$ has a set of source nodes, $N_s^k \subseteq N^k$, and a set of demand nodes, $N_d^k \subseteq N^k$. There is a set of disrupted nodes $N'^k \subseteq N^k$ and disrupted links $L'^k \subseteq L'^k$ for each network $k \in K$ following a disruption.

Supply for each node $i \in N_s^k$ in network $k \in K$ is denoted by $b_i^k$. Supply $b_i^k$ is considered to be the maximum flow from node $i \in N_s^k$ to node $i \in N_d^k$ and is considered independent of time. Unmet demand for network $k \in K$ during time $t \in T$ in node $i \in N_d^k$ is represented as slack, $s_{it}^k$. Slack may be described as the extent to which demand is not being met; thus, it is, in part, used to represent resilience (Almoghathawi et al., 2016).

Resilience can be described as the loss of maximum flow in a network. Equation (2) quantifies resilience as the proportion of slack in the time periods following a disruption to the original slack in the set of interdependent networks. Each network $k \in K$ has a weight $\mu^k$ such that $\sum_{k=1}^{K} \mu^k = 1$. The total slack prior to a disruption and immediately following a disruption in network $k \in K$ are represented by $S_0^k$ and $S_d^k$, respectively.

$$\sum_{k \in K} \mu^k \left[ \frac{\sum_{t \in T} \left[ t \left( S_d^k - \sum_{i \in N_d^k} s_{it}^k \right) - (t-1)(S_d^k - \sum_{i \in N_d^k} s_{i(t-1)}^k) \right]}{T(S_d^k - S_0^k)} \right] \tag{2}$$

In addition to maximizing the resilience in the set of interdependent networks, a competing objective is to minimize the total cost of restoration. Total cost of restoration is a function of fixed restoration costs, unitary flow cost, and the cost of unmet demand; it is captured by Equation (3). In this work, $fn_i^k$ and $fl_{ij}^k$ are the fixed restoration costs for node $i \in N'^k$ and link $(i,j) \in L'^k$ for network $k \in K$, respectively. The decision variable $z_i^k$ is a binary variable that equals 1 if node $i \in N'^k$ in network $k \in K$ is chosen for restoration and 0 otherwise. Similarly, $y_{ij}^k$ is a binary decision variable that equals 1 if link $(i,j) \in L'^k$ in network $k \in K$ is chosen for restoration and 0 otherwise. There are also per-unit costs associated with flow and unmet demand. Let $c_{ij}^k$ be the unitary cost of flow on link $(i,j) \in L^k$ and let $p_i^k$ be the cost of unmet demand at node $i \in N_d^k$ in network $k \in K$. Flow across link $(i,j) \in L^k$ in network $k \in K$ in period $t \in T$ is represented by the continuous decision variable $x_{ijt}^k$. As before, unmet demand is equated to slack and represented by $s_{it}^k$ for node $i \in N'^k$ in network $k \in K$ in period $t \in T$.

$$\min \sum_{k \in K} \left( \sum_{i \in N'^k} fn_i^k z_i^k + \sum_{(i,j) \in L'^k} fl_{ij}^k y_{ij}^k + \sum_{t \in T} \left[ \sum_{(i,j) \in L^k} c_{ij}^k x_{ijt}^k + \sum_{i \in N^k} p_i^k s_{it}^k \right] \right) \tag{3}$$

Each disrupted element has a fixed restoration time that must be elapsed before an element may be considered operational. The restoration times for node $i \in N'^k$ and link

$(i,j) \in L'^k$ are denoted by $dn_i^k$ and $dl_{ij}^k$, respectively. Once a disrupted element has been selected for restoration and has completed its restoration time, it becomes operational. Two binary decision variables--$\beta_{it}^k$ and $\alpha_{ijt}^k$—indicate the status of node $i \in N'^k$ and link $(i,j) \in L'^k$, respectively. $\beta_{it}^k$ equals 1 if node $i \in N'^k$ in network $k \in K$ is operational during time $t \in T$ and is 0 otherwise. Similarly, $\alpha_{ijt}^k$ equals 1 if link $(i,j) \in L'^k$ in network $k \in K$ is operational during time $t \in T$ and is 0 otherwise. Each network $k \in K$ has a set of work crews $R^k$ dedicated to restoring its disrupted elements. There are two decision variables associated with scheduling work crews to restore disrupted elements, $\gamma_{it}^{kr}$ and $\delta_{ijt}^{kr}$. $\gamma_{it}^{kr}$ equals 1 if work crew $r \in R^k$ in network $k \in K$ is selected to restore node $i \in N'^k$ during time $t \in T$ and 0 otherwise. In the same way, if work crew $r \in R^k$ in network $k \in K$ is chosen to restore link $(i,j) \in L'^k$ during time $t \in T$, $\delta_{ijt}^{kr}$ equals 1 and is 0 otherwise. Finally, each network contains nodes that are dependent on specific nodes from another network being operational. In this work, $\Psi$ is used to represent interdependence such that $\left((i,k),(\bar{\imath},\bar{k})\right) \in \Psi$ indicates that node $\bar{\imath} \in N^{\bar{k}}$ in network $\bar{k} \in K$ requires node $i \in N^k$ in network $k \in K$ to be operational.

*4.1.1 Mathematical Model*

$$\max \sum_{k \in K} \mu^k \left[ \frac{\sum_{t \in T} \left[ t\left(S_d^k - \sum_{i \in N_d^k} s_{it}^k\right) - (t-1)(S_d^k - \sum_{i \in N_d^k} s_{i(t-1)}^k) \right]}{T(S_d^k - S_0^k)} \right] \tag{4}$$

$$\min \sum_{k \in K} \left( \sum_{i \in N'^k} fn_i^k z_i^k + \sum_{(i,j) \in L'^k} fl_{ij}^k y_{ij}^k + \sum_{t \in T} \left[ \sum_{(i,j) \in L^k} c_{ij}^k x_{ijt}^k + \sum_{i \in N^k} p_i^k s_{it}^k \right] \right) \tag{5}$$

Subject to:

$$\sum_{(i,j)\epsilon L^k} x_{ijt}^k \leq b_i^k, \quad \forall i \,\epsilon\, N_s^k, k \,\epsilon\, K, t \,\epsilon\, T \tag{6}$$

$$\sum_{(i,j)\,\epsilon\, L^k} x_{ijt}^k - \sum_{(j,i)\,\epsilon\, L^k} x_{jit}^k = 0, \quad \forall i \,\epsilon\, N^k\{N_s^k, N_d^k\}, k \,\epsilon\, K, t \,\epsilon\, T \tag{7}$$

$$\sum_{(j,i)\,\epsilon\, L^k} x_{jit}^k + s_{it}^k = b_i^k, \quad \forall\, i \,\epsilon\, N_d^k, k \,\epsilon\, K, t \,\epsilon\, T \tag{8}$$

$$x_{ijt}^k - u_{ij}^k \leq 0, \quad \forall (i,j) \,\epsilon\, L^k, k \,\epsilon\, K, t \,\epsilon\, T \tag{9}$$

$$x_{ijt}^k - u_{ij}^k \beta_{it}^k \leq 0, \quad \forall (i,j) \,\epsilon\, L^k, i \,\epsilon\, N'^k, k \,\epsilon\, K, t \,\epsilon\, T \tag{10}$$

$$x_{ijt}^k - u_{ij}^k \alpha_{ijt}^k \leq 0, \quad \forall (i,j) \,\epsilon\, L'^k, k \,\epsilon\, K, t \,\epsilon\, T \tag{11}$$

$$\beta_{\bar{\imath}t}^{\bar{k}} - \beta_{it}^k \leq 0, \quad \forall \left((i,k),(\bar{\imath},\bar{k})\right) \,\epsilon\, \Psi, t \,\epsilon\, T \tag{12}$$

$$y_{ij}^k = \sum_{r\,\epsilon\, R^k} \sum_{t\,\epsilon\, T} \delta_{ijt}^{kr}, \quad \forall (i,j) \,\epsilon\, L'^k, k \,\epsilon\, K \tag{13}$$

$$z_i^k = \sum_{r\,\epsilon\, R} \sum_{t\,\epsilon\, T} \gamma_{it}^{kr}, \quad \forall i \,\epsilon\, N'^k, k \,\epsilon\, K \tag{14}$$

$$\sum_{(i,j)\,\epsilon\, L'^k} \sum_{l=t}^{\min\{T,t+dl_{ij}^k-1\}} \delta_{ijl}^{kr} + \sum_{(i,j)\,\epsilon\, N'^k} \sum_{l=t}^{\min\{T,t+dn_i^k-1\}} \gamma_{il}^{kr} \leq 1, \quad \forall\, k \,\epsilon\, K, r \,\epsilon\, R^k \tag{15}$$

$$\alpha_{ijt}^k \leq \sum_{r\,\epsilon\, R^k} \sum_{t'=1}^{t} \delta_{ijt'}^{kr}, \quad \forall (i,j) \,\epsilon\, L'^k, k \,\epsilon\, K, t \,\epsilon\, T \tag{16}$$

$$\beta_{it}^k \leq \sum_{r\,\epsilon\, R^k} \sum_{t'=1}^{t} \gamma_{it'}^{kr}, \quad \forall i \,\epsilon\, N'^k, k \,\epsilon\, K, t \,\epsilon\, T \tag{17}$$

$$\sum_{t=1}^{dl_{ij}^k-1} \alpha_{ijt}^k = 0, \quad \forall (i,j) \in L'^k, k \in K \tag{18}$$

$$\sum_{t=1}^{dn_i^k-1} \beta_{it}^k = 0, \quad \forall i \in N'^k, k \in K \tag{19}$$

$$\sum_{r \in R^k} \sum_{t=1}^{dl_{ij}^k-1} \delta_{ijt}^{kr} = 0, \quad \forall (i,j) \in L'^k, k \in K \tag{20}$$

$$\sum_{r \in R^k} \sum_{t=1}^{dn_i^k-1} \gamma_{it}^{kr} = 0, \quad \forall i \in N'^k, k \in K \tag{21}$$

$$s_{it}^k \geq 0, \quad \forall i \in N^k, k \in K, t \in T \tag{22}$$

$$x_{ijt}^k \geq 0, \quad \forall (i,j) \in L^k, k \in K, t \in T \tag{23}$$

$$y_{ij}^k \in \{0,1\}, \quad \forall (i,j) \in L'^k, k \in K \tag{24}$$

$$z_i^k \in \{0,1\}, \quad \forall i \in N^k, k \in K \tag{25}$$

$$\alpha_{ijt}^k \in \{0,1\}, \quad \forall (i,j) \in L'^k, k \in K, t \in T \tag{26}$$

$$\beta_{it}^k \in \{0,1\}, \quad \forall i \in N^k, k \in K, t \in T \tag{27}$$

$$\delta_{ijt}^{kr} \in \{0,1\}, \quad \forall (i,j) \in L'^k, k \in K, t \in T, r \in R^k \tag{28}$$

$$\gamma_{it}^{kr} \in \{0,1\}, \quad \forall i \in N'^k, k \in K, t \in T, r \in R^k \tag{29}$$

The objectives of this model are to maximize resilience of the interdependent infrastructure networks $K$ while minimizing the total cost of restoration. These objectives are represented mathematically in Equations (4) and (5). The rest of the model is broken into three sets of constraints which help describe network flow, the interdependency of networks, and work group scheduling for restoration.

Constraints (6) – (11) are dedicated to network flow. Specifically, constraints (6) specify that each source node $i \in N_s^k$ in network $k \in K$ for each time $t \in T$ cannot output more than its supply, $b_i^k$. Flow must be conserved into and out of node $i \in N^k$. This is accounted for by constraints (7). The combination of flow into all demand nodes $i \in N_d^k$ and the unmet demand at those nodes must equal the demand in network $k \in K$ during time $t \in T$, as described in constraints (8). Each link $(i,j) \in L^k$ in network $k \in K$ has a capacity that is described by $u_{ij}^k$. As such, the flow across each link, $x_{ij}^k$, for all $(i,j) \in L^k$ in network $k \in K$ cannot exceed that link's capacity. This is dictated by constraints (9)-(11). Interdependency of networks is described by constraints (12) such that node $\bar{\imath} \in N^{\bar{k}}$ in network $\bar{k} \in K$ may not be operational unless its interdependent node, $i \in N^k$ in network $k \in K,$ is also operational.

Constraints (13)-(23) are in place for the scheduling of work crews to restore disrupted elements. Constraints (13) and (14) state that a work crew must be assigned to repair all elements that have been selected for restoration. Constraints (15) ensure that work crew $r \in R^k$ for network $k \in K$ can only restore one disrupted element during time $t \in T.$ A disrupted element cannot be operational unless a work crew has been assigned to it; thus constraints (16) and (17) are put in place. Constraints (18)-(21) exist so that a disrupted element cannot be considered operational until it has completed its restoration time. The remaining constraints, (22)-(29) are used to describe the nature of all decision variables i.e. whether they are continuous or binary.

**4.2 Facilities Location and Work Crew Assignment**

Each work crew $r \epsilon R^k$ must dispatch from an assigned location. Work crews must be assigned to origin locations such that the fixed cost of establishment and the distance travelled is minimized. Thus, in addition to the model presented in Section 4.1, there is a set of candidate sites, $M,$ where work crews must be assigned prior to a disruptive event. Before a work crew can be assigned to a facility, the facility must be established in a candidate location. Binary decision variable $v_m$ equals 1 if candidate site $m \epsilon M$ is established and is 0 otherwise. There is a fixed cost associated with establishing a resource facility; this cost is represented by $cs_m$ for site $m \epsilon M$. Further, if work crew $r \epsilon R^k$ in network $k \epsilon K$ is stationed at site $m \epsilon M$, the binary decision variable $w_{mr}^k$ equals 1 and is 0 otherwise. Of course, each candidate site is positioned some distance from all disrupted elements in network $k \epsilon K$. For nodes, this distance is the Euclidean distance from candidate site $m \epsilon M$ to node $i \epsilon N'^k$ in network $k \epsilon K$ and is represented by $ns_{im}^k$. For links, the Euclidean distance from candidate site $m \epsilon M$ to the midpoint of link $(i, j) \epsilon L'^k$ in network $k \epsilon K$ is represented by $ls_{ijm}^k$. There is also a unitary cost associated with the distance a work group must travel from each candidate site to a disrupted element. This cost is captured by $DC_m$ for site $m \epsilon M$. Both $cs_m$ and $DC_m$ are incorporated into the cost objective as shown in Equation (30).

$$\min \sum_{k \in K} \left( \sum_{i \in N'^k} fn_i^k z_i^k + \sum_{(i,j) \in L'^k} fl_{ij}^k y_{ij}^k + \sum_{t \in T} \left[ \sum_{(i,j) \in L^k} c_{ij}^k x_{ijt}^k + \sum_{i \in N^k} p_i^k s_{it}^k \right] + \sum_{m \in M} \left[ cs_m v_m + \sum_{t \in T} \sum_{r \in R^k} \left[ \sum_{i \in N'^k} ns_{im}^k DC_m w_{rm} \gamma_m^{kr} + \sum_{(i,j) \in L'^k} ls_{ijm}^k DC_m w_{rm} \delta_{ijt}^{kr} \right] \right] \right) \tag{30}$$

When deciding where work groups should originate from, there must be constraints added to those described by Almoghathawi et al. (2016). Constraints (31)-(40) are defined as follows:

$$\sum_{k \in K} \sum_{r \in R^k} w_{mr}^k \leq 1, \quad \forall\, m \in M \tag{31}$$

$$v_m \geq w_{mr}^k, \quad \forall\, m \in M, r \in R^k, k \in K \tag{32}$$

$$\sum_{m \in M} w_{mr}^k = 1, \quad \forall\, r \in R^k, k \in K \tag{33}$$

$$v_m \in \{0,1\}, \quad \forall\, m \in M \tag{34}$$

$$w_{mr}^k \in \{0,1\}, \quad \forall\, m \in M, r \in R^k, k \in K \tag{35}$$

Constraints (31) state that candidate site $m \in M$ may be assigned at most one work crew $r \in R^k$ in network $k \in K$. This also implies that there may not be work crews from different networks assigned the same site. It is also important to consider the fact that a workgroup cannot be assigned to a site that is not established; thus, constraints (32) are in place so that work crew $r \in R^k$ for network $k \in K$ may not be assigned to site $m \in M$ unless it has been selected to be established. It should be noted that a work crew may not change locations. Therefore, each work group $r \in R^k$ in network $k \in K$ may only be assigned to one selected site, as described by constraints (33). Constraints (34) and (35) define $v_m$ and $w_{mr}^k$ as binary variables.

*4.2.1 Addressing Non-Linearity of Cost Objective*

Because two decision variables are multiplied together in the objective presented in Equation (30), the model is non-linear. To continue using a linear solver, Equation (36) is included as a substitute for objective function (30).

$$\min \sum_{k \in K} \left( \sum_{i \in N'^k} fn_i^k z_i^k + \sum_{(i,j) \in L'^k} fl_{ij}^k y_{ij}^k + \sum_{t \in T} \left[ \sum_{(i,j) \in L^k} c_{ij}^k x_{ijt}^k + \sum_{i \in N^k} p_i^k s_{it}^k \right] + \sum_{m \in M} \left[ cs_m v_m + \sum_{t \in T} \sum_{r \in R^k} \left[ \sum_{i \in N'^k} G_{imt}^{kr} + \sum_{(i,j) \in L'^k} H_{ijmt}^{kr} \right] \right] \right) \tag{36}$$

$$G_{imt}^{kr} \geq ns_{im}^k DC_m \left( \gamma_{it}^{kr} + w_{mr} - 1 \right), \quad \forall\, i \in N'^k, t \in T, m \in M, r \in R^k, k \in K \tag{37}$$

$$H_{ijmt}^{kr} \geq ns_{im}^k DC_m \left( \delta_{ijt}^{kr} + w_{mr} - 1 \right), \quad \forall\, (i,j) \in L'^k, t \in T, m \in M, r \in R^k, k \in K \tag{38}$$

$$G_{imt}^{kr} \geq 0, \quad \forall\, i \in N'^k, t \in T, m \in M, r \in R^k, k \in K \tag{39}$$

$$H_{ijmt}^{kr} \geq 0, \quad \forall\, (i,j) \in L'^k, t \in T, m \in M, r \in R^k, k \in K \tag{40}$$

Constraints (37)-(40) must also be added to the model to maintain the significance of objective function (30).

*4.2.2 Model Simplification*

As previously noted, the proposed model is a multi-objective optimization problem. As such, there are several trade-off solutions to be considered. To simplify the solution process, the $\varepsilon$-constraint method is used in this work (Haimes et al., 1971). The $\varepsilon$-constraint method treats one objective as the primary objective while the other is constrained to a specified target value. Using this method, objective function (36) is minimized and objective function (4) is substituted for constraint (41). Constraint (41) is bound by a minimum level of resilience as determined by decision makers of the interdependent networks. Since resilience is continuous between 0 and 1, $\varepsilon \in [0,1]$.

$$\sum_{k \in K} \mu^k \left[ \frac{\sum_{t \in T} \left[ t \left( S_d^k - \sum_{i \in N_d^k} s_{it}^k \right) - (t-1)(S_d^k - \sum_{i \in N_d^k} s_{i(t-1)}^k) \right]}{T(S_d^k - S_0^k)} \right] \geq \varepsilon \qquad (41)$$

# Chapter 5.0 Illustrative Example

## 5.1 Data

Real data related to existing infrastructure networks is often difficult to find to protect against the risk of malevolent attacks. As such, to test the proposed model, a network with randomly generated components is used. This set of simulated interdependent infrastructure networks is created in R Studio using the method originally described by Casey (2005). The network itself is generated by first establishing the random networks then creating interdependencies between them. Once the network is in place, the candidate sites for work groups is added to the graph.

First, the independent networks must be established. The coordinates of the nodes in each network are random and uniformly distributed between 0 and 1. Source nodes are the first to be added for each network. At this point, they are considered independent of each other, so there are no links established between them. As each demand node is added to the network, it is connected to the nearest existing node from the same network in the graph. Each link is considered undirected (i.e. there may be flow in either direction). "Nearness" is determined by smallest Euclidean distance.

Next, the interdependencies between networks must be established. The source and demand nodes from each network are considered to be dependent on nodes from the other network. Thus, a link is established between each supply and demand node and the nearest source or demand node from another network. Again, nearness is defined as the smallest Euclidean distance.

Finally, candidate sites must be incorporated into the graph. In this work, candidate sites are placed on the graph in a 5x5 grid resulting in a total of 25 candidate sites. The candidate locations are equally spaced in the vertical and horizontal directions. Figure 2 shows the final graph used in this illustration.

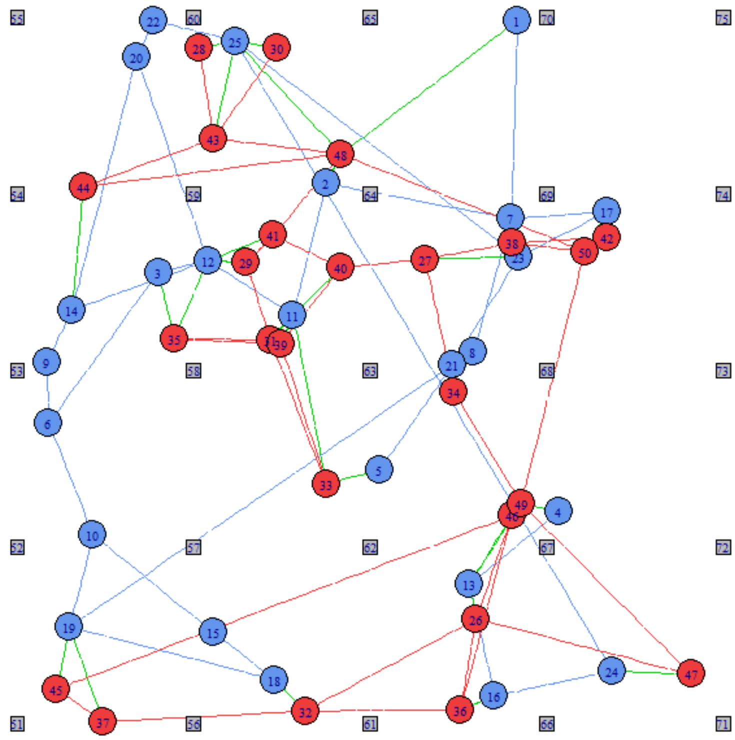


Figure 2: Interdependent Network Graph with 25 Candidate sites

As described by Almoghathawi et al. (2016), the two interdependent infrastructure networks present in this work simulate a water (red nodes and links) and a power network (blue nodes and links). The water network depends on the power network to pump and distribute water; the power network depends on the water network for cooling and to reduce emissions. In Figure 2, these interdependencies are shown by green arcs. Power generators and substations act as supply and demand nodes, respectively, in the power network; power lines act as the links between them. In the water network, supply nodes represent water pumps and demand nodes represent storage tanks. The links between them represent pipelines. Candidate locations for work group facilities are indicated by gray squares in Figure 2.

### 5.2 Example

For each disruption scenario described in Chapter 3, there is a certain number of elements removed to simulate a disruption. For random, capacity-based, and degree-based disruptions, 21% of components are disrupted (5 nodes and 7 links from each network). For spatial disruption scenarios, however, the disrupted elements are confined to a specific area. As a result, demand in other nodes may still be met through other channels. To combat this phenomenon, the special disruption scenario requires a greater number of elements to be disrupted for the same loss in network flow efficiency as the other disruption scenarios. For this scenario, 32% of network components are disrupted (9 nodes and 12 links from the power network; 6 nodes and 9 links from the water network) (Almoghathawi et al., 2016).

#### *5.2.1 Pareto-Optimal Solutions*

The experiment is performed using LINGO 16.0. The problem is solved for varying values of $\varepsilon$ to assess how the total cost of restoration changes for different levels of resilience and create the Pareto frontier of non-dominated solutions for each disruption scenario. Pareto-optimal solutions are only available when the cost of restoration, $fn_i^k$, is greater than the cost of unmet demand, $p_i^k$, for node $i \in N'^k$. Because both objectives are focused on minimizing the unmet demand, if $fn_i^k < p_i^k$, given there is enough time to restore essential disrupted elements, resilience will always reach 1 (Almoghathawi et al., 2016). Therefore, to create the Pareto frontier, unmet demand is under-penalized. Figure 3 shows the set of Pareto optimal solutions for the capacity-based, degree-based, random, and spatial disruption scenarios with $\varepsilon = [0.5,1]$.

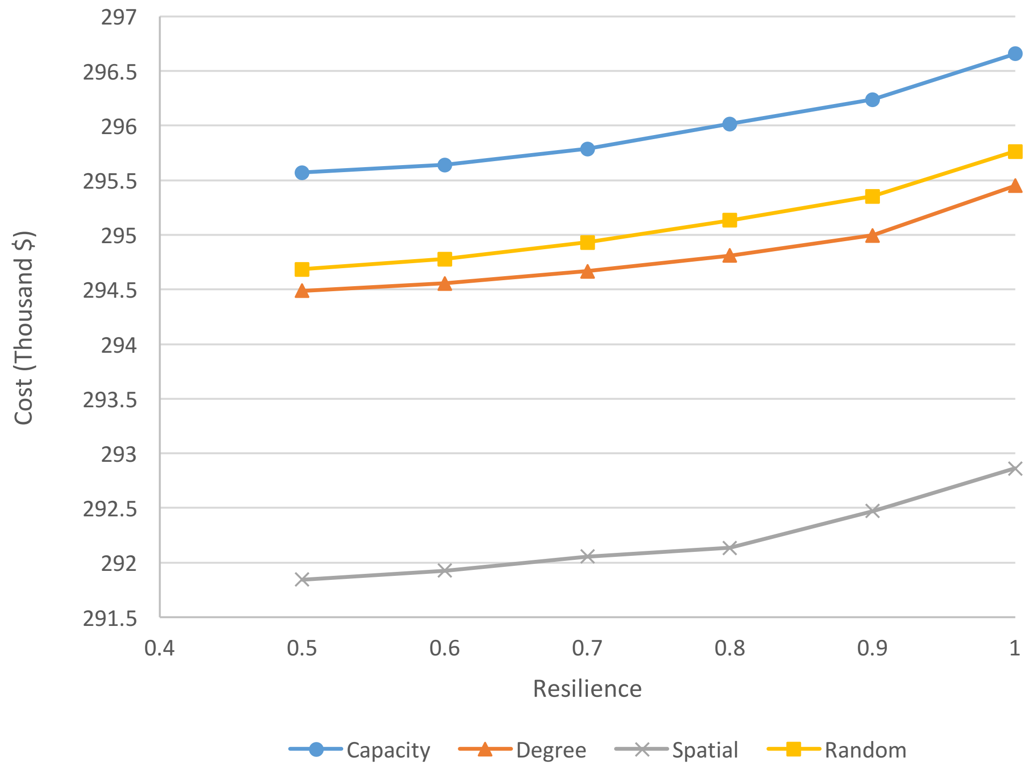
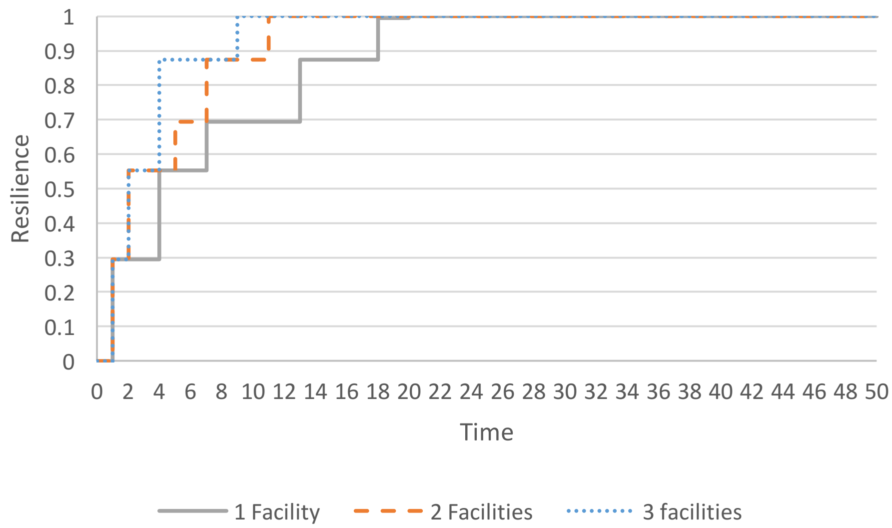



Figure 3: Pareto-optimal frontier for competing objectives cost and resilience

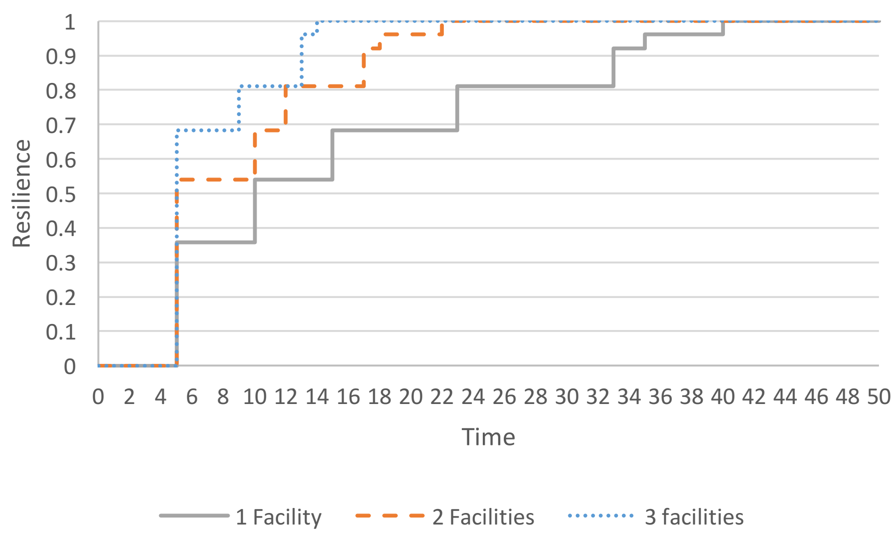
The lowest cost is observed when resilience = 0.5 and the highest cost occurs for resilience = 1 for all disruption scenarios. The capacity-based disruption scenario results in the highest total restoration cost while the spatial disruption scenario has the lowest total restoration cost, regardless of level of resilience.

*5.2.2 Assessment of Varying the Number of Established Facilities*

For the remainder of the experiment, the parameters are set as follows: $\mu^k = \frac{1}{K}$, $T = 50$, $fn_i^k, fl_{ij}^k, u_{ij}^k, cs_m \sim U(20,50)$, $c_{ij}^k, DC_m \sim U\ (1,10)$, $p_i^k = 60$ and $dn_i^k, dl_{ij}^k \sim U(1,5)$. Figures 4, 5, 6 and 7 show the resilience across the available time periods for restoration given the selection of 1, 2, and 3 sites for work groups in each network.

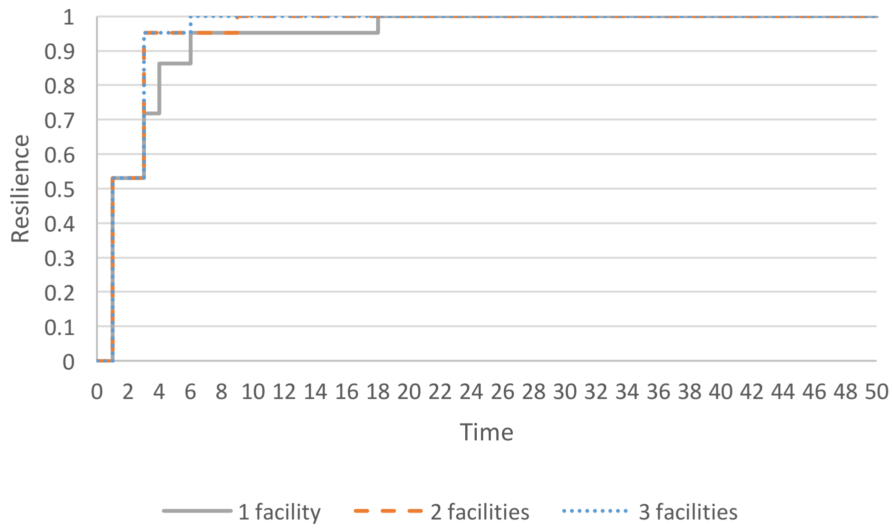



(a)

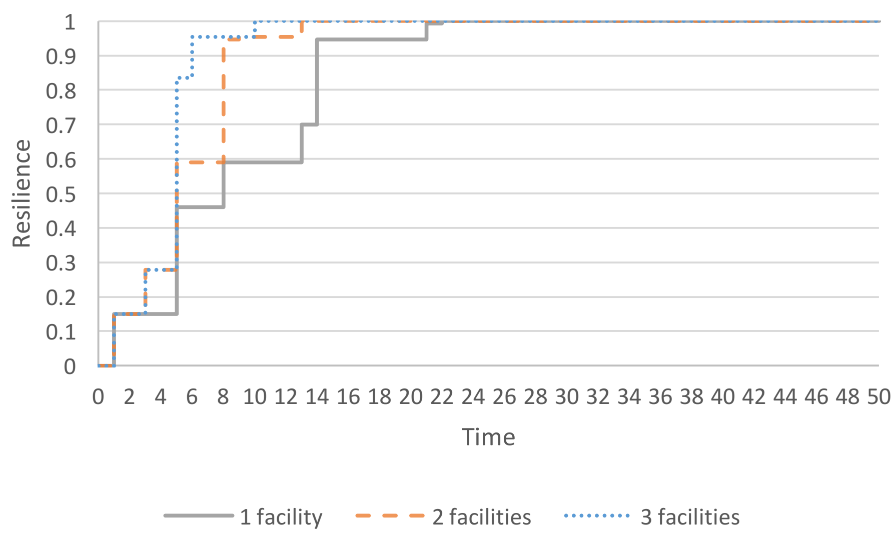



(b)

Figure 4: Resilience vs. time for a capacity-based disruption for (a) power network and (b) water network considering a different number of established facilities

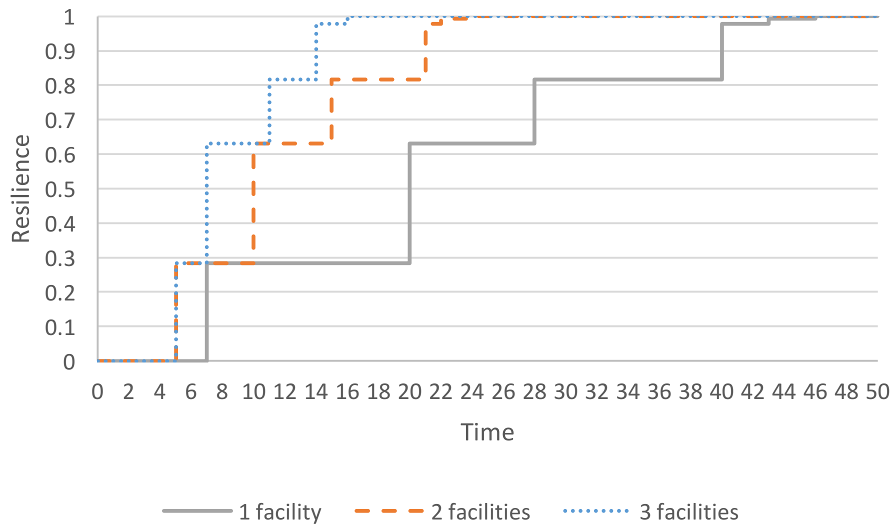



(a)

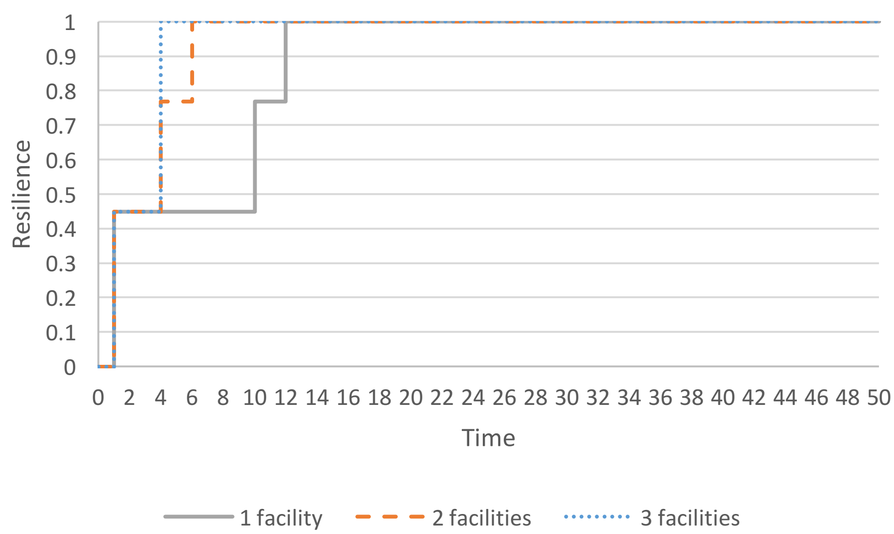



(b)

Figure 5: Resilience vs. time for a degree-based disruption for (a) power network and (b) water network considering a different number of established facilities

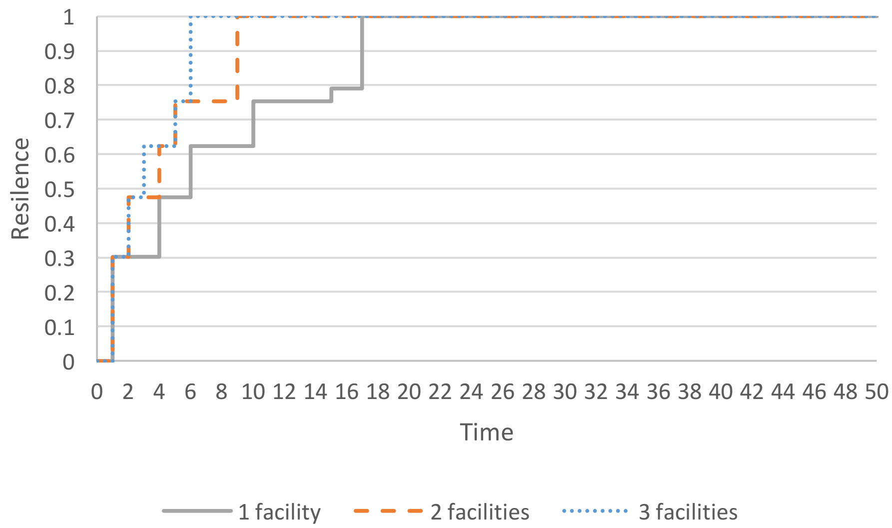



(a)

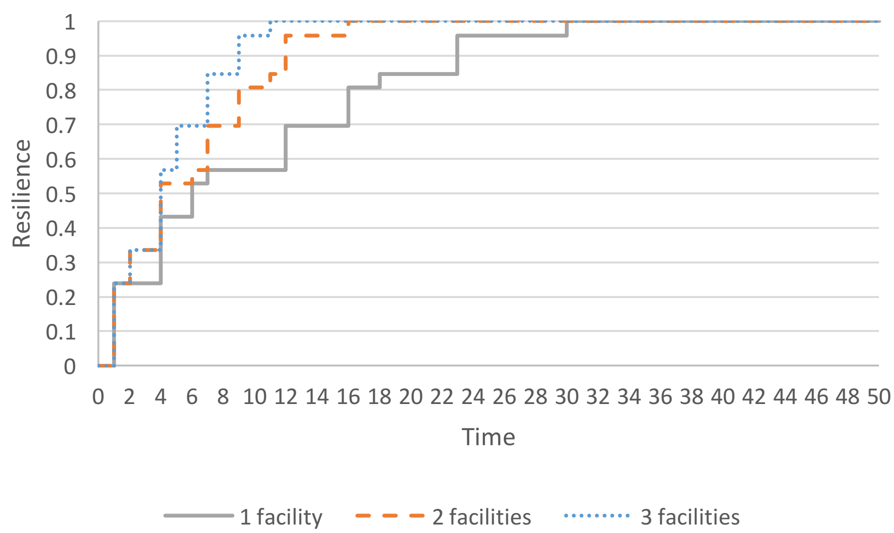



(b)

Figure 6: Resilience vs. time for a spatial disruption for (a) power network and (b) water network considering a different number of established facilities




(a)




(b)

Figure 7: Resilience vs. time for a random disruption for (a) power network and (b) water network considering a different number of established facilities

As shown in Figures 3-6 above, increasing the number of facilities and, in turn, increasing the number of work crews in each network, reduces the time to full resilience. It should also be noted that because of the interdependencies between the two networks, one network may reach full resilience before the other. The model inherently prioritizes the recovery of interdependent nodes, but one network may take longer to reach full resilience.

There is a tradeoff that occurs when increasing the number of facilities. As the number of established facilities increases, additional fixed facilities costs are incurred. There may exist some benefits to increasing the number of facilities, though. By establising additional facilities, the distance each work crew must travel to repair a disrupted element and the number of time periods with unmet demand are decreased. So, determining the number of facilities to establish is dependent on the cost parameters associated with these decision variables. In the case of this experiment, increasing the number of established facilities decreases the total cost of restoration because the fixed cost of establishing a facility is small compared to the cost of unmet demand.

# Chapter 6.0 Concluding Remarks

## 6.1 Summary

Modern society heavily depends on critical infrastructure networks, such as electricity, water, transportation, and telecommunications, for everyday activities. Just as we are dependent on these networks, these networks also depend on each other for operation. There exist several complex relationships between each of these critical infrastructures that make them highly vulnerable in the event of a malevolent attack, natural disaster, or random failure. As such, it has become increasingly important to not only protect these networks, but also create a plan for restoring them.

This work proposes a model that can be used following a disruptive event to restore interdependent infrastructure networks to some desired level of resilience while minimizing the total cost of restoration. The model not only schedules work crews to restore disrupted elements, but also determines where work crews should originate from, given a set of candidate locations.

## 6.2 Future work

This model should be applied to a larger scale to illustrate its usability in a real-world scenario. Affiliates of Rice University have created a graphical representation of supply and demand nodes associated with the interdependent gas, power, and water networks of Shelby County, Tennessee. Shelby County is highly susceptible to earthquakes, so the model could be primarily used to determine where recovery crews should be stationed prior to a spatial disruption. However, the data could also be used to demonstrate the other disruption scenarios discussed in this work.

To improve the solution of this model, it may be important to introduce a clustering of network nodes. In so doing, candidate sites could be located at the center of these clusters to minimize the distance a work crew would need to travel to restore disrupted elements. It is important to note that although candidate sites may exist at the center of these clusters, it may not be optimal to position a work crew in those locations. Additionally, the proposed model assumes that only one network may use an established facility. However, it may be more economical to allow work crews from different networks to be stationed at the same facility.

A robust decision approach to account for several possible disruptions is also an important extension of this work. Since the model is so dependent on a specific disruption scenario, by considering the probability of different spatial disruptions a more intelligent solution may be available with this approach. Similarly, incorporating game theory to try to predict the malevolent actions could result in a better solution.

Finally, infrastructure resilience is not the only component of community resilience. As such, in future applications of this work, it will be important to consider the vulnerability of different groups of people affected by a disruption.

## 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.

Barker, K., Lambert, J.H., Zobel, C.W., Tapia, A.H., Ramirez-Marquez, J.E., McLay, L.A., Nicholson, C.D., & Caragea, C. (2017). *Defining resilience analytics for interdependent cyber-physical-social networks.* Taylor & Francis.

Batta, R., & Mannur, N. R. (1990). Covering-location models for emergency situations that require multiple response units. *Management science*, *36*(1), 16-23.

Casey, M.J. (2005). *Self-organization and topology control of infrastructure sensor networks* (Doctoral Dissertation). University of Maryland, College Park.

Cavdaroglu, B., Hammel, E., Mitchell, J. E., Sharkey, T. C., & Wallace, W. A. (2013). Integrating restoration and scheduling decisions for disrupted interdependent infrastructure systems. *Annals of Operations Research*, 1-16.

Executive Office of the President. (Feb. 2013). *Critical Infrastructure Security and Resilience.* Presidential Policy Directive/PPD-21. Washington, DC.

Exec. Order No. 13411, 3 C.F.R. 1527-1528 (2006).

Gong, J., Lee, E. E., Mitchell, J. E., & Wallace, W. A. (2009). Logic-based multiobjective optimization for restoration planning. In Optimization and Logistics Challenges in the Enterprise (pp. 305-324). Springer US.

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

Henry, D., & Ramirez-Marquez, J. E. (2012). Generic metrics and quantitative approaches for system resilience as a function of time. Reliability Engineering & System Safety, 99, 114-122.

Jia, H., Ordóñez, F., & Dessouky, M. (2007). A modeling framework for facility location of medical services for large-scale emergencies. *IIE transactions*, *39*(1), 41-55.

Jönsson, H., Johansson, J., & Johansson, H. (2008). Identifying critical components in technical infrastructure networks. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 222(2), 235-243.

Lee II, E. E., Mitchell, J. E., & Wallace, W. A. (2007). Restoration of services in interdependent infrastructure systems: A network flows approach. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 37(6), 1303-1317.

National Infrastructure Advisory Council. (2009). *Critical Infrastructure Resilience: Final Report and Recommendations*. Department of Homeland Security.

The Report of the President's Commission on Critical Infrastructure Protection. 1997. *Critical Foundation: Protecting America's Infrastructure*. USA.

Rinaldi, S.M., Peerenboom, J.P., and Kelly, T.K. (2001). Identifying, understanding, and analyzing critical infrastructure interdependencies. *IEEE Control Systems Magazine,* **21**(6): 11 25.

Rose, A. (2007). Economic resilience to natural and man-made disasters: Multidisciplinary origins and contextual dimensions. *Environmental Hazards*, 7(4), 383-398.

United States. Cong. *USA PATRIOT Act of 2001*. 107 Cong., 1st sess. Cong 3162. Washington, D.C.: n.p., 2001

Wang, S., Hong, L., & Chen, X. (2012). Vulnerability analysis of interdependent infrastructure systems: A methodological framework. *Physica A: Statistical Mechanics and its applications*, *391*(11), 3323-3335

# Appendix A Network Generation

This section includes all code generated in R Studio used to generate the two interdependent infrastructure networks used in the illustration of this model. The following code is adapted from the network generation proposed by Almoghathawi et al., 2016.

```
library(igraph)
library(gdata)
library(hierarchicalDS)

#n = total nodes in network 1 and network 2
#n1 = nodes in network 1
#n2 = nodes in network 2
#s1 = source nodes in network 1
#s2 = source nodes in network 2
#N  = overall number of nodes including all candidate sites (n + 25)

#Network 1 = Power
#Network 2 = Water

two.graph.CS <- function(n, n1, s1, s2) {

  #Number of actual nodes in the graph
  n2=n1-n
  #Number of nodes including all candidate sites
#  N=n+100
  N=n+25

  #to fix the values of the random coordinates unless n is changed
  set.seed(n)

  #generate random coordinates and candidate sites
  xy <- rbind(cbind(runif(n), runif(n)),
              cbind(0,seq(0,1,.25)),
              cbind(.25,seq(0,1,.25)),
              cbind(.5,seq(0,1,.25)),
              cbind(.75,seq(0,1,.25)),
              cbind(1,seq(0,1,.25)))
```

```
  distMat <- as.matrix(dist(xy, method = 'euclidean', upper = T, diag = T))
  distmat <- distMat[1:n,1:n]

  #set a large value for the lower trainagle so they won't be selected as min distance
b/w nodes
  lowerTriangle(distMat[1:n1,1:n1], diag=TRUE) <- 10
  lowerTriangle(distMat[(n1+1):n,(n1+1):n], diag=TRUE) <- 10

  #create the adjacency matrix
  adjMat <- matrix(0, nr = N, nc = N)



  #ADD LINKS OF NETWORK 1:
  #-----------------------

  #min of each column
  for (j in 1:n1){
    for(i in 1:n1){
      #ifelse(distMat[i,j] == min(distMat[1:n1,j]), adjMat[i,j] <- 1 , adjMat[i,j] <- 0)
      ifelse(distMat[i,j] == min(distMat[1:n1,j]), adjMat[i,j] <- 1 , adjMat[i,j])
    } }

  #min of each row
  for (j in 1:n1){
    for(i in 1:n1){
      ifelse(distMat[i,j] == min(distMat[i,1:n1]) , adjMat[i,j] <- 1, adjMat[i,j])
    } }

  #connect each source to the nearest node (not source)
  for (j in (s1+1):n1){
    for(i in 1:s1){
      ifelse(distMat[i,j] == min(distMat[i,(s1+1):n1]), adjMat[i,j] <- 1 , adjMat[i,j])
    } }

  #ADD LINKS OF NETWORK 2:
  #-----------------------

  #min of each column
  for (j in (n1+1):n){
    for(i in (n1+1):n){
```

```
      ifelse(distMat[i,j] == min(distMat[(n1+1):n,j]), adjMat[i,j] <- 1 , adjMat[i,j] <-
0)
    } }

  #min of each row
  for (j in (n1+1):n){
    for(i in (n1+1):n){
      ifelse(distMat[i,j] == min(distMat[i,(n1+1):n]) , adjMat[i,j] <- 1 , adjMat[i,j])
    } }


  #connect each source to the nearest node (not source)
  for (j in (n1+s2+1):n){
    for(i in (n1+1):(n1+s2)){
      ifelse(distMat[i,j] == min(distMat[i,(n1+s2+1):n]), adjMat[i,j] <- 1 ,
adjMat[i,j])
    } }

  #ADD DEPENDENCY LINKS:
  #---------------------
  #from water to power:
  for (j in 1:s1){
    for(i in (n1+s2+1):n){
      ifelse(distMat[i,j] == min(distMat[(n1+s2+1):n,j]), adjMat[i,j] <- 1 ,
adjMat[i,j])
    }  }

  #from power to water:
  for (j in (n1+1):n){
    for(i in (s1+1):n1){
      ifelse(distMat[i,j] == min(distMat[(s1+1):n1,j]), adjMat[i,j] <- 1 , adjMat[i,j]
<- 0)
    }  }

  #Creating the adjacency for the candidate sites
  adjMat[(n+1):(N),(n+1):(N)] <- square_adj(5)

  #set the lower traiangle of the adjacenecy matrix to 0:
  lowerTriangle(adjMat[1:n1,1:n1], diag=TRUE) <- 0
  lowerTriangle(adjMat[(n1+1):n,(n1+1):n], diag=TRUE) <- 0
```

```
  #isolate the source nodes of Network 1 (i.e. no links b/w them):
  adjMat[1:s1, 1:s1] <- 0
  #isolate the source nodes of Network 2 (i.e. no links b/w them):
  adjMat[(n1+1):(n1+s2), (n1+1):(n1+s2)] <- 0

  #return: xy coordinates, distance matrix, and adjacency matrix:
  return(list(xy, distMat, adjMat))

}
#Create a network of 50 total nodes. 25 in each network with 5 source nodes.
g <- two.graph.CS(50, 25, 5, 5)

distances <- g[[2]]



n=50
n1=25
N=n+25

#getting the xy coordinates for all nodes
COORD <- g[[1]]
#just the coordinates for the network
coordNet <- g[[1]][1:n,]
#just the coordinates for the candidate locations
coordGrid <-g[[1]][(n+1):N,]

#recalling the adjacency matrix
mat = g[[3]]

#Create a graph from the adjacency matrix
net = graph.adjacency(mat, mode="undirected")

#labelling each node
V(net)$label <- c(1:N)

#NAME the nodes based on which network they belong to
for (i in 1:N){
  ifelse (0 < V(net)$label[i] & V(net)$label[i] <= n1, V(net)$name[i] <- "Network1",
          ifelse(n1 < V(net)$label[i] & V(net)$label[i] <= n, V(net)$name[i] <-
"Network2",
            V(net)$name[i] <- "CandidateLocations"))
```

```
}

#COLOR the NODES based on their network
V(net)$color <- ifelse(V(net)$name == "Network1", "cornflowerblue",
                       ifelse(V(net)$name == "Network2","brown2",
                              "gray"))
#change the lable's  font size
V(net)$label.cex[V(net)$name == "Network1"] <- 0.9
V(net)$label.cex[V(net)$name == "Network2"] <- 0.9
V(net)$label.cex[V(net)$name == "CandidateLocations"] <- 0.6

#get the two end nodes of all edges
edge.ends <- ends(net, es=E(net), names=T)
k=0
for (i in 1:nrow(edge.ends)){
  if(edge.ends[i,1] == "Network1" & edge.ends[i,2] == "Network1"){
    k=k+1}
  else if(edge.ends[i,1] == "Network2" & edge.ends[i,2] == "Network2"){
    k=k+1}
}

#just the ends for the network nodes
NNedge.ends <- edge.ends[1:k,]

#EDGES:
for (i in 1:nrow(edge.ends)){
  if(edge.ends[i,1] == "Network1" & edge.ends[i,2] == "Network1"){
    E(net)$color[i] <- "cornflowerblue"
    E(net)$lty[i]   <- 1
    E(net)$width[i] <- "1.5"}
  else if(edge.ends[i,1] == "Network2" & edge.ends[i,2] == "Network2"){
    E(net)$color[i] <- "brown2"
    E(net)$lty[i]   <- 1
    E(net)$width[i] <- "1.5"}
  else if(edge.ends[i,1] == "Network1" & edge.ends[i,2] == "Network2"){
    E(net)$color[i] <- "green3"
    E(net)$lty[i]   <- 1
    E(net)$width[i] <- "1.5"}

  else{
    if(edge.ends[i,1] == "CandidateLocations" & edge.ends[i,2] == "CandidateLocations"){
```

```
      E(net)$color[i] <- "white"}
  }}

#NODES:
for (i in 1:max(V(net)$label)){
  if(V(net)$name[i] == "Network1" | V(net)$name[i] == "Network2"){
    V(net)$shape[i] <- "circle"
    V(net)$size[i]  <- 8
    V(net)$label.cex <- 0.9}


  else{
    V(net)$shape[i] <- "square"
    V(net)$size[i]  <- 4
    V(net)$label.cex <- 0.6}
}

#plot(layout_with_kk(net,COORD))
# To make them spaced more evenly

plot(net,layout=COORD)


#Create the distance matrix between all points
N2CS <- as.matrix(dist(COORD))
#Trims the matrix so it only shows the distance between all network nodes and each
candidate location
N2CS <- N2CS[1:n,(n+1):N]
View(N2CS)

#write to csv
write.csv(N2CS,file='NodetoCSv3.csv')


#function to find the midpoint of 2 coordinates
midcoord <- function(x1,x2) {

  mid <- (x1+x2)/2

  return(mid)
```

```
}

#finding the endpoints of each link
edge.ends2 <- ends(net, es=E(net), names=F)
#Trims that matrix to just the links between actual network nodes (excludes candidat
site adjacency)
midpoints <- edge.ends2[1:102,]

#Creating vectors for the midpoints
x <- c(1:nrow(midpoints))
y <- c(1:nrow(midpoints))

#Calculates the x and y components of the midpoint between linked nodes
for (i in 1:nrow(midpoints)){

  a <- midpoints[i,1]
  b <- midpoints[i,2]

  x[i] <- midcoord(COORD[a,1],COORD[b,1])
  y[i] <- midcoord(COORD[a,2],COORD[b,2])

}

#Combines the matrix of endpoints and the two vectors with the coordinates of the
midpoint
midpoints <- cbind(midpoints, x)
midpoints <- cbind(midpoints, y)
View(midpoints)

#Write to csv
write.csv(midpoints,file='Linksv3.csv')

#Creates a matrix of the x and y coordiates for the midpoints of links and the candidate
locations
LinkCSCoords <- rbind(midpoints[,3:4],COORD[(n+1):N,])

#finds the distance between all midpoints and candidate sites
distMatLinkCS <- as.matrix(dist(LinkCSCoords, method = 'euclidean', upper = T, diag =
T))
#trims the matrix to contain the midpoints on the vertical portion and candidate sites
on the horizontal portion
```

```
distMatLinkCS <- distMatLinkCS[1:nrow(midpoints),(nrow(midpoints)+1):nrow(LinkCSCoords)]

#write to csv
write.csv(distMatLinkCS,file='LinktoCSv3.csv')
View(distMatLinkCS)
```

## Appendix B Network Restoration Model with Location Selection

This section details the proposed model presented in Chapter 4. The model is written using LINGO 16.0.

```
!  Resilience-Driven Recovery Model with Origin Selection for WG;
!  Adapted from Almoghathawi et al., 2016;

SETS:
        TIME;  !available time periods;
        WG;    !available Work groups;
        NETWORK; !Interdependent networks;
        NODE;

        CS:      ! Available candidate sites;
          FB,    ! 1 if a facility is built at CS m, 0 otherwise;
          FSC, ! Fixed site selection cost;
          VDC; ! Variable distance cost;

        CNN(NODE,NETWORK): !there is a set of nodes;
          Q,   !unit cost of unmet demand             (parameter);
          B,   !demand-supply (= max flow)            (parameter);
          RN,  !time when node i is operational;
          FN,  !recovery cost for a node              (parameter);
          DN,   !time periods needed to restore node i (parameter);
          VN,  !1 if node i is undisrupted and 0 o.w. (parameter);
          Z;    !1 if node i is selected to be restored and 0 o.w. (DV);

        CLN(NODE,NODE,NETWORK):!there is a set of links;
          CAP, !capacity                                  (parameter);
          CF,  !flow unitary cost                         (parameter);
          RL,  !time when link (i,j) is operational;
          FL,  !recovery cost for a link                  (parameter);
          DL,   !time periods needed to restore link (i,j)(parameter);
          VL,   !1 if link (i,j) is disrupted and 0 o.w.   (parameter);
          Y;    !1 if link (i,j) is selected to be restored and 0 o.w.(DV);

        DEP(CNN,CNN):
          DD;   !Dependency (parameter);

        CNT(CNN, TIME): !combination of node and time period;
          SU,  !unmet demand                              (DV);
          ZT;   !1 if node i is operational and 0 o.w. (DV);

        CLT(CLN, TIME): !combination of link and time period;
          X,   !flow                                      (DV);
          YT;   !1 if link (i,j) is operational and 0 o.w.(DV);

        CNTR(CNN, TIME, WG): !combination of node, time period and work group;
          VT;   !1 if restored at time t by work group r and 0 o.w. (DV);

        CLTR(CLN, TIME, WG): !combination of link, time period and work group;
          WT;   !1 if restored at time t by work group r and 0 o.w. (DV);

        CNCS (CNN, CS):
          DNCS, ! distance between each node and candidate site;
          NS;          ! 1 if candidate site m is chosen to serve disrupted node i, 0
otherwise;

        CLCS (CLN, CS):
          DLCS, ! Distance between each link and candidate site;
          LS;          ! 1 if candidate site m is chosen to serve disrupted link (i,j),
0 otherwise;

        CCSR (CS, WG, NETWORK):
          SR;   ! 1 if WG r is chosen to be stationed at site m, 0 otherwise;
```

```
        NALL (CNN, TIME, CS, WG):
                LIN1;   !decision variable to deal with non-linear nature of objective;

        LALL (CLN, TIME, CS, WG):
                LIN2;   !Second decision variable to deal with non-linear nature of
objective;

ENDSETS


DATA:
NETWORK = 1..2;
NODE = 1..54;

CNN, VN, DN, Q, B, FN       = @OLE('C:\Users\CEC448\Dropbox\Thesis-EM\Example 6 - random
25CS', 'CNN', 'VN', 'DN', 'Q', 'B', 'FN');
DEP, DD                     = @OLE('C:\Users\CEC448\Dropbox\Thesis-EM\Example 6 - random
25CS', 'FromD', 'FROM.NET', 'ToD', 'TO.NET', 'DD');
CLN, CAP, VL, DL, CF, FL    = @OLE('C:\Users\CEC448\Dropbox\Thesis-EM\Example 6 - random
25CS', 'CLN', 'CAP', 'VL', 'DL', 'CF', 'FL');
CS, DNCS, DLCS, FSC, VDC    = @OLE('C:\Users\CEC448\Dropbox\Thesis-EM\Example 6 - random
25CS', 'CS', 'NODE.TO.CS', 'LINK.TO.CS', 'FSC', 'VDC');


FPo  = 190;   !Original Maximum flow for Power Network;
FWo  = 280;   !Original Maximum flow for Water Network;
FPd  = 28;    !Maximum flow after disruption for Power Network (random);
FWd  = 0;     !Maximum flow after disruption for Water Network (random);
!FPd  = 0;    !Maximum flow after disruption for Power Network (capacity);
!FWd  = 0;    !Maximum flow after disruption for Water Network (capacity);
!FPd  = 45;   !Maximum flow after disruption for Power Network (degree);
!FWd  = 0;    !Maximum flow after disruption for Water Network (degree);
!FPd  = 49;   !Maximum flow after disruption for Power Network (spatial);
!FWd  = 211;  !Maximum flow after disruption for Water Network (spatial);
TIME = 1..50; !Available time periods for restoration process;
TT   = 50;    !Max time;
ss1  = 51;    !supersource for network 1;
st1  = 52;    !superterminal for network 1;
ss2  = 53;    !supersource for network 2;
st2  = 54;    !superterminal for network 2;
n1   = 25;    !number of nodes in network 1;
WG   = 1..3;  !available work groups;

ENDDATA


  !OBJECTIVE FUNCTION:
! MAXIMIZE THE SYSTEM RESILIENCE and MINIMIZE COSTS OF (RESTORATION + FLOW +
DISRUPTION);

        MIN = OBJECTIVE2;

        !OBJECTIVE 1: Maximize the resilience;
        OBJECTIVE1 =   (SP / (2*TT*SPd)) + (SW / (2*TT*SWd));

        !OBJECTIVE 2: Minimize the cost;
        OBJECTIVE2 =   @SUM(CLN(i,j,k)| VL(i,j,k)#NE# 1:FL(i,j,k)  * Y(i,j,k)) +
                               @SUM(CNN(i,k)   | VN(i,k)   #NE# 1:  FN(i,k)  *   Z(i,k)) +
                               @SUM(TIME(t):
                                       @SUM(CLN(i,j,k) :CF(i,j,k)* X(i,j,k,t)) +
                                       @SUM(CNN(i,k)   :Q(i,k)*SU(i,k,t)) ) +
                               @SUM(CS(m): FSC(m)*FB(m)) +
                               @SUM(TIME(t):
                                       @SUM(CS(m):
                                               @SUM(WG(r):
                                                       @SUM(CNN(i,k): LIN1(i,k,t,m,r)) +
                                                       @SUM(CLN(i,j,k) : LIN2(i,j,k,t,m,r)
))));


        SP1     = @SUM(TIME(t)| t #EQ# 1:    SPd - @SUM(CNN(i,k)| k #EQ# 1: SU(i,1,1)));
```

```
        SP2      = @SUM(TIME(t)| t #GE# 2: t*(SPd - @SUM(CNN(i,k)| k #EQ# 1: SU(i,1,t)))
- (t-1)*(SPd - @SUM(CNN(i,k)| k #EQ# 1: SU(i,1,t-1))));
        SP       = SP1 + SP2;
        SPd      = FPo - FPd;

        SW1      = @SUM(TIME(t)| t #EQ# 1:    SWd - @SUM(CNN(i,k)| k #EQ# 2: SU(i,2,1)));
        SW2      = @SUM(TIME(t)| t #GE# 2: t*(SWd - @SUM(CNN(i,k)| k #EQ# 2: SU(i,2,t)))
- (t-1)*(SWd - @SUM(CNN(i,k)| k #EQ# 2: SU(i,2,t-1))));
        SW       = SW1 + SW2;
        SWd      = FWo - FWd;


        OBJECTIVE1 >= 1;

  !CONSTRAINTS;:

!A. NETWORK FLOW CONSTRAINTS:

! A.1 CONSERVATION CONSTRAINTS OF FLOW AT NODE I:;
 !Conservation for the super source nodes;
 @FOR(CNN(i,k)|  B(i,k) #GT# 0: @FOR(TIME(t): @SUM(CLN(i,j,k): X(i,j,k,t)) -
@SUM(CLN(j,i,k): X(j,i,k,t))<= B(i,k)));
 !For all in between nodes;
 @FOR(CNN(i,k)|  B(i,k) #EQ# 0: @FOR(TIME(t): @SUM(CLN(i,j,k): X(i,j,k,t)) -
@SUM(CLN(j,i,k): X(j,i,k,t)) = 0));
 !For the super terminal nodes;
 @FOR(CNN(i,k)|  B(i,k) #LT# 0: @FOR(TIME(t): @SUM(CLN(i,j,k): X(i,j,k,t)) -
@SUM(CLN(j,i,k): X(j,i,k,t)) - SU(i,k,t) = B(i,k)));


! A.2 CAPACITY CONSTRAINTS ON LINK (I,J);
! When everything is disrupted;
 @FOR(CLN(i,j,k)|  VL(i,j,k) #EQ# 1 #AND# VN(i,k) #EQ# 1 #AND# VN(j,k) #EQ# 1:
     @FOR(TIME(t): X(i,j,k,t) <= CAP(i,j,k)));
! When only the first node might be disrupted;
  @FOR(CLN(i,j,k)| VN(i,k) #NE# 1:
        @FOR(TIME(t): X(i,j,k,t) <= CAP(i,j,k) * ZT(i,k,t)));
! When only the second node might be disrupted;
  @FOR(CLN(i,j,k)| VN(j,k) #NE# 1:
        @FOR(TIME(t): X(i,j,k,t) <= CAP(i,j,k) * ZT(j,k,t)));
! When only the link is disrupted;
  @FOR(CLN(i,j,k)| VL(i,j,k) #NE# 1:
         @FOR(TIME(t): X(i,j,k,t) <= CAP(i,j,k) * YT(i,j,k,t)));



! B. INTERDEPENDENCY CONSTRAINTS;
  @FOR(TIME(t):
         @FOR(DEP(i,k,j,l)|  DD(i,k,j,l) #EQ# 1: ZT(i,k,t)*DD(i,k,j,l)  >= ZT(j,l,t)));



! C. ASSIGNMENT AND SCHEDULING CONSTRAINTS:

! RESTORATION TIME CONSTRAINTS;
! LINK CONSTRAINTS;

  @FOR(CNN(i,k)| VN(i,k) #NE# 1:
                Z(i,k) - @SUM(WG(r): @SUM(TIME(t): VT(i,k,t,r))) = 0);

  @FOR(CLN(i,j,k)|  VL(i,j,k) #NE# 1:
                Y(i,j,k) -  @SUM(WG(r): @SUM(TIME(t): WT(i,j,k,t,r))) = 0);



!WORK GROUPS CONSTRAINTS;
  @FOR (NETWORK(k) :
        @FOR(WG(r):
                @FOR(TIME(t):
                        @SUM(CNN(i,k)|VN(i,k) #NE# 1:
```

```
                      @SUM(TIME(l)  |l #GE# t #AND# l #LE@  #SMIN)TT,(t+DN(i,k)-
1):( VT(i,k,l,r)))
                      +
                      @SUM(CLN(i,j,k)|VL(i,j,k) #NE# 1:
                      @SUM(TIME(l)  |l #GE# t #AND# l #LE@  #SMIN)TT,(t+DL(i,j,k)-
1):( WT(i,j,k,l,r))) <= 1)));


  @FOR(TIME(t):
       @FOR(CNN(i,k)  | VN(i,k)  #NE# 1:
              ZT(i,k,t) <= @SUM(WG(r): @SUM(TIME(l)  |l #GE# 1 #AND# l #LE# t  :
VT(i,k,l,r)))));

  @FOR(TIME(t):
       @FOR(CLN(i,j,k)|VL(i,j,k)#NE# 1:
              YT(i,j,k,t) <= @SUM(WG(r): @SUM(TIME(l)  |l #GE# 1 #AND# l #LE# t  :
WT(i,j,k,l,r)))));



! A DISRUPTED NETWORK ELEMENT CANNOT BE RESTORED (FUNCTIONAL) BEFORE COMPLETING ITS
RESTORATION TIME;
  @FOR(CNN(i,k)| VN(i,k) #NE# 1:
      @SUM(TIME(t)  |t #GE# 1 #AND# t #LE# (DN(i,k)-1): ZT(i,k,t)) = 0);

  @FOR(CLN(i,j,k)| VL(i,j,k) #NE# 1:
      @SUM(TIME(t)  |t #GE# 1 #AND# t #LE# (DL(i,j,k)-1): YT(i,j,k,t)) = 0);

  @FOR(CNN(i,k)| VN(i,k) #NE# 1:
       @SUM(WG(r):
      @SUM(TIME(t)  |t #GE# 1 #AND# t #LE# (DN(i,k)-1): VT(i,k,t,r))) = 0);

  @FOR(CLN(i,j,k)| VL(i,j,k) #NE# 1:
       @SUM(WG(r):
      @SUM(TIME(t)  |t #GE# 1 #AND# t #LE# (DL(i,j,k)-1): WT(i,j,k,t,r))) = 0);


! TIME WHEN A COMPONENT (NODE OR LINK) IS OPERATIONAL;
  @FOR(CNN(i,k)  | VN(i,k)   #NE# 1: RN(i,k)   = 1 + (@SUM(TIME(t): (1-ZT(i,k,t)))));
  @FOR(CLN(i,j,k)|VL(i,j,k) #NE# 1: RL(i,j,k) = 1 + (@SUM(TIME(t): (1-YT(i,j,k,t)))));

  @FOR(CNN(i,k)  | VN(i,k)   #EQ# 1: RN(i,k)   = 0);
  @FOR(CLN(i,j,k)| VL(i,j,k) #EQ# 1: RL(i,j,k) = 0);

! FACILITY LOCATION CONSTRAINTS;

  ! Each chosen candidate site can be assigned at most 1 workgroup;
  @FOR (CS(m):
       @SUM(NETWORK(k): @SUM(WG(r): SR(m,r,k))) <= 1);

  ! A work group can only be assigned to 1 chosen site;
  @FOR (NETWORK(k):
       @FOR (WG(r): @SUM(CS(m): SR(m,r,k)) = 1));

  ! A workgroup must be assigned to a chosen candidate site;
  @FOR(NETWORK(k):
       @FOR (CS(m):
              @FOR(WG(r): FB(m) >= SR(m,r,k))));

! LINEARITY;
 @FOR (CNN(i,k)|VN(i,k)#NE# 1:
       @FOR(TIME(t):
              @FOR(CS(m):
                      @FOR(WG(r): LIN1(i,k,t,m,r) >=
DNCS(i,k,m)*VDC(m)*(VT(i,k,t,r)+SR(m,r,k)-1)))));

 @FOR (CLN(i,j,k)|VL(i,j,k) #NE# 1:
```

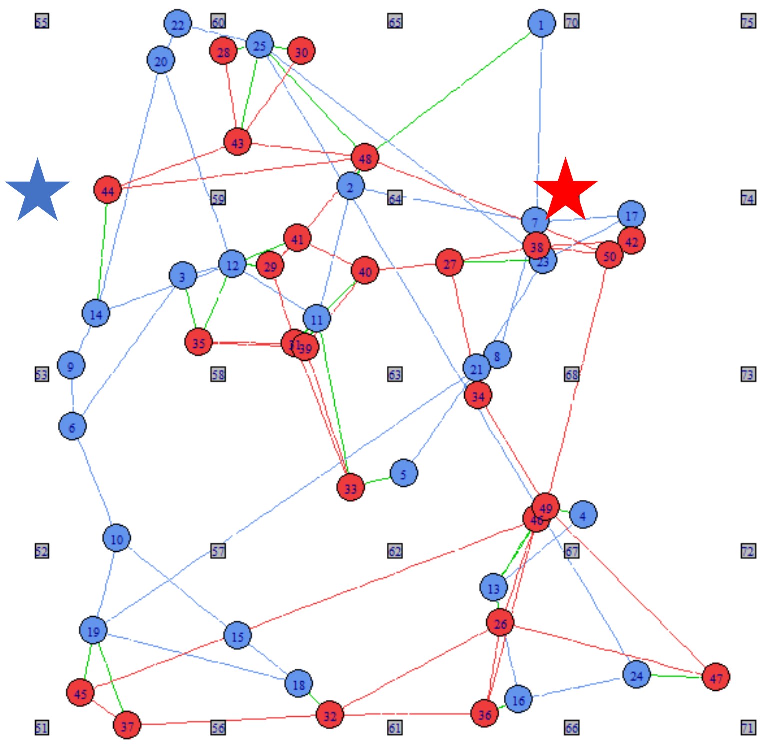
```
        @FOR(TIME(t):
                @FOR (CS(m):
                        @FOR(WG(r): LIN2(i,j,k,t,m,r) >=
DLCS(i,j,k,m)*VDC(m)*(WT(i,j,k,t,r)+SR(m,r,k)-1)))));

! D. DECISION VARIABLES NATURE CONSTRAINTS;
  @FOR(CNT(i,k,t):  @GIN(SU(i,k,t)));
  @FOR(CLT(i,j,k,t):@GIN(X(i,j,k,t)));

  @FOR(CNN(i,k)     |VN(i,k)   #NE# 1: @BIN(Z(i,k)));
  @FOR(CLN(i,j,k)   |VL(i,j,k)#NE# 1: @BIN(Y(i,j,k)));

  @FOR(CNN(i,k)    |VN(i,k)   #EQ# 1:      Z(i,k) = 0);
  @FOR(CLN(i,j,k)    |VL(i,j,k)#EQ# 1:      Y(i,j,k) = 0);


  @FOR(CNT(i,k,t)      |VN(i,k)   #NE# 1: @BIN(ZT(i,k,t)));
  @FOR(CLT(i,j,k,t)    |VL(i,j,k)#NE# 1: @BIN(YT(i,j,k,t)));
  @FOR(CNTR(i,k,t,r)   |VN(i,k)   #NE# 1: @BIN(VT(i,k,t,r)));
  @FOR(CLTR(i,j,k,t,r)|VL(i,j,k)#NE# 1: @BIN(WT(i,j,k,t,r)));

! @FOR(CNT(i,t)      |VN(i)   #EQ# 1:      ZT(i,t) = 0);
 @FOR(CLT(i,j,k,t)    |VL(i,j,k)#EQ# 1:   YT(i,j,k,t) = 0);
 @FOR(CNTR(i,k,t,r)   |VN(i,k)   #EQ# 1:   VT(i,k,t,r) = 0);
 @FOR(CLTR(i,j,k,t,r)|VL(i,j,k)#EQ# 1: WT(i,j,k,t,r) = 0);

  @FOR(CS(m): @BIN(FB(m)));
  @FOR(CCSR(m,r,k): @BIN(SR(m,r,k)));

!  @FOR(CS(m) | FB(m) #NE# 1:
        @FOR(CCSR(m,r): SR(m,r) = 0));


  @FOR(NALL(i,k,t,m,r): LIN1(i,k,t,m,r) >= 0);
  @FOR(LALL(i,j,k,t,m,r): LIN2(i,j,k,t,m,r) >= 0);
```

# Appendix C Locations Selected

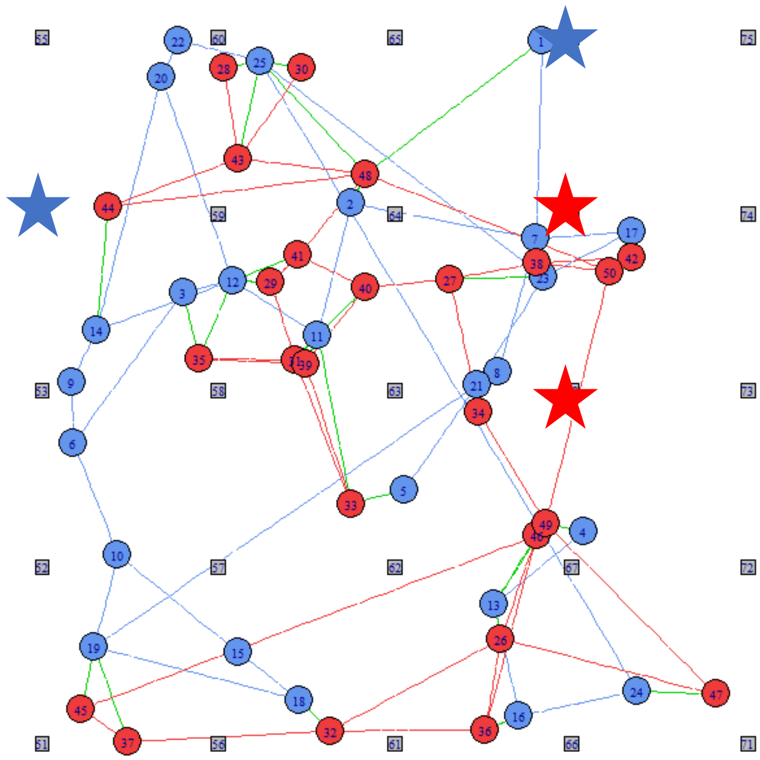
This section shows the locations selected for each disruption scenario. The locations selected for varying number of work crews will be illustrated. Sites selected for the power network will be highlighted by a blue star and sites selected for the water network will be highlighted by a red star.

## Appendix C.1 Capacity-based Disruption Scenario

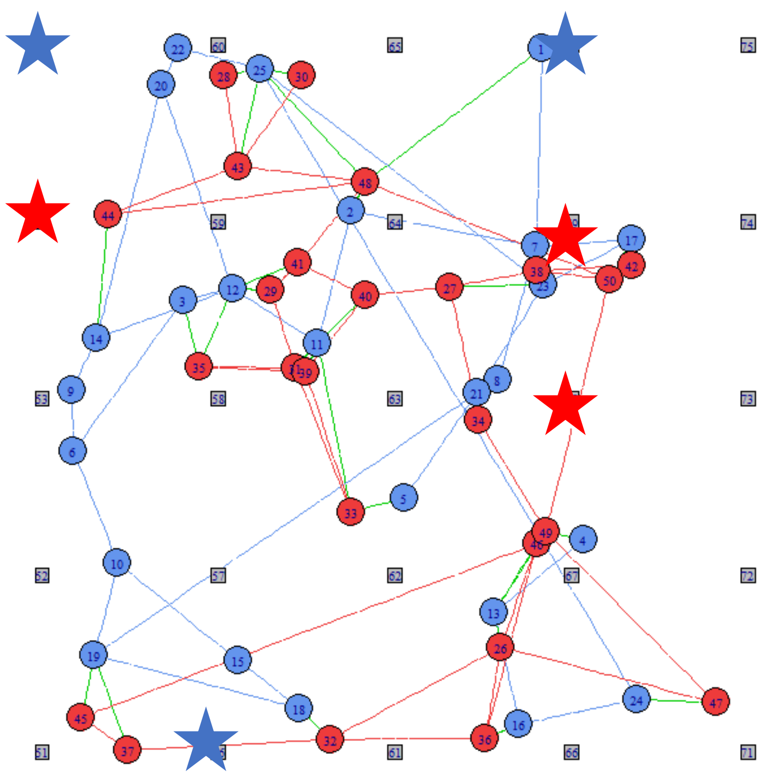
### *Appendix C.1.1 1 Facility*



*Appendix C.1.2 2 Facilities*

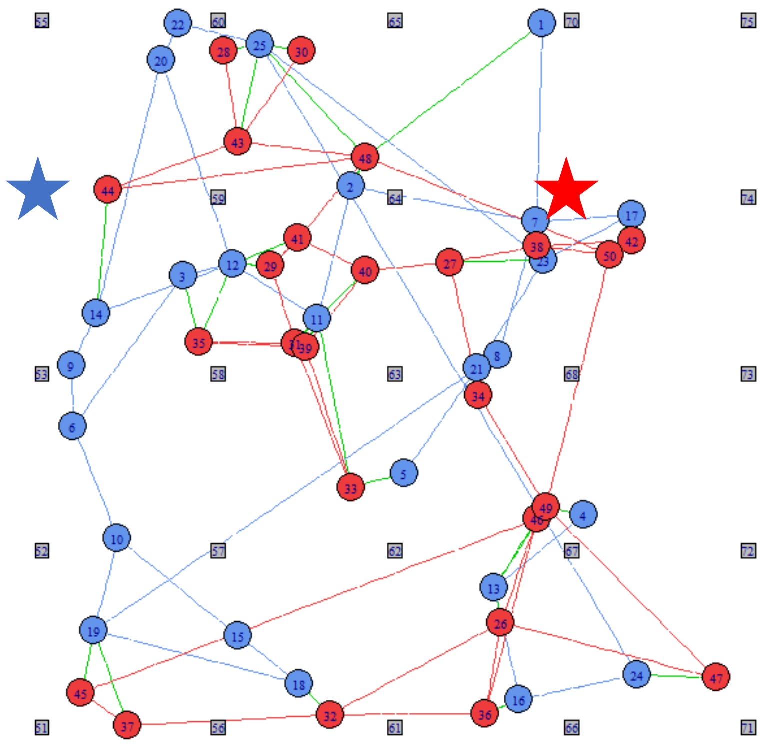


*Appendix C.1.3 3 Facilities*

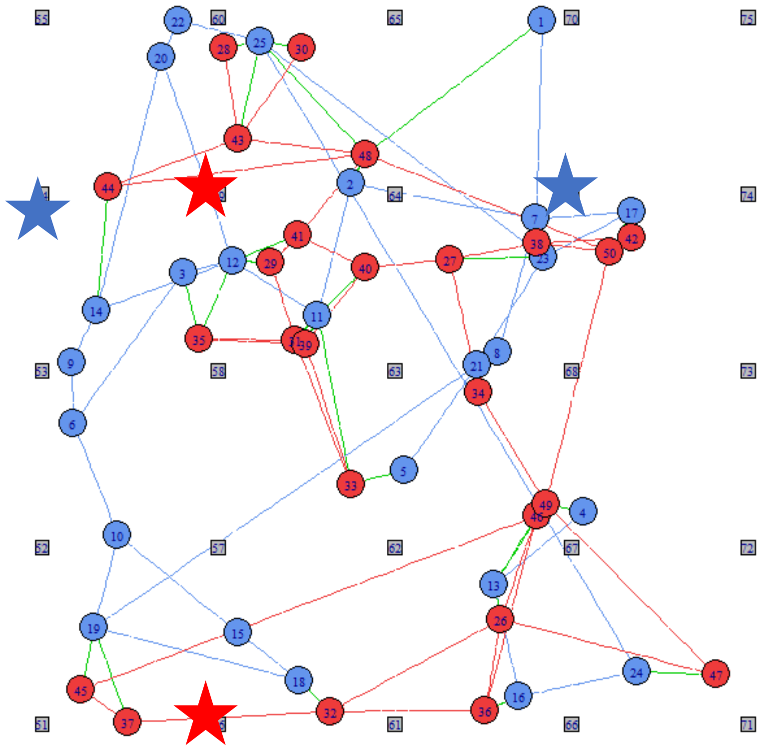


## Appendix C.2 Degree-based Disruption Scenario

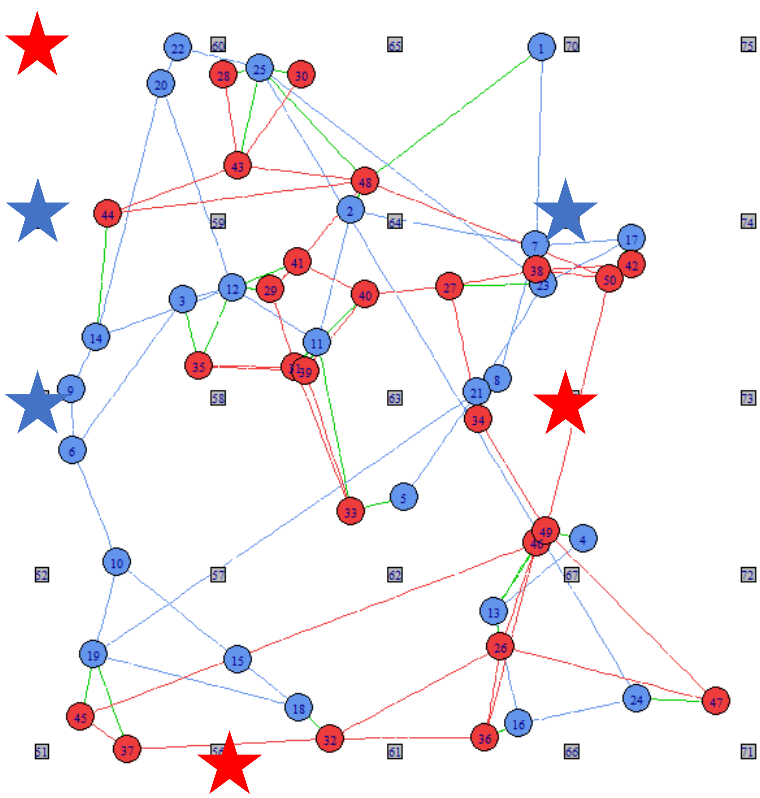
*Appendix C.2.1 1 Facility*



*Appendix C.2.2 2 Facilities*

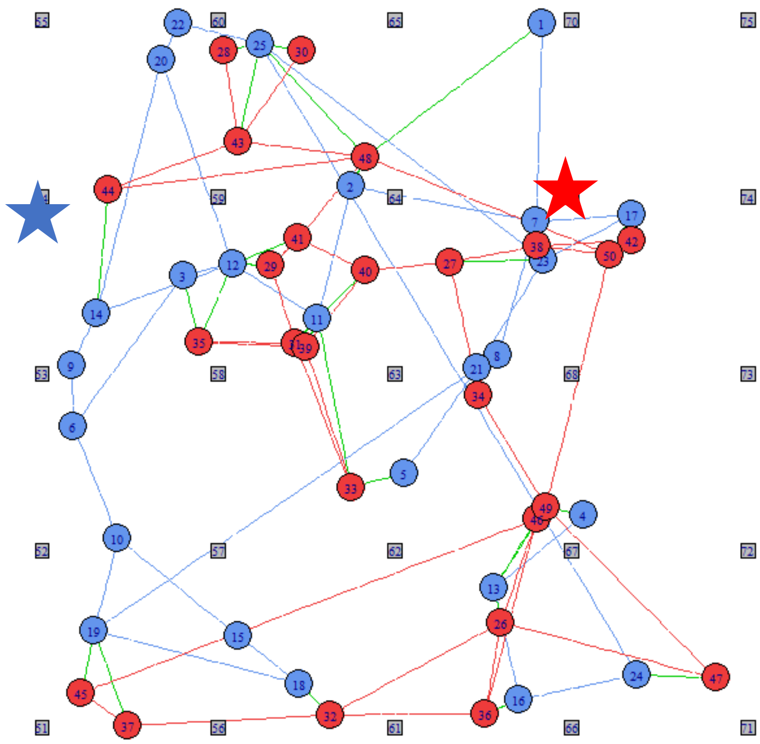


*Appendix C.2.3 3 Facilities*

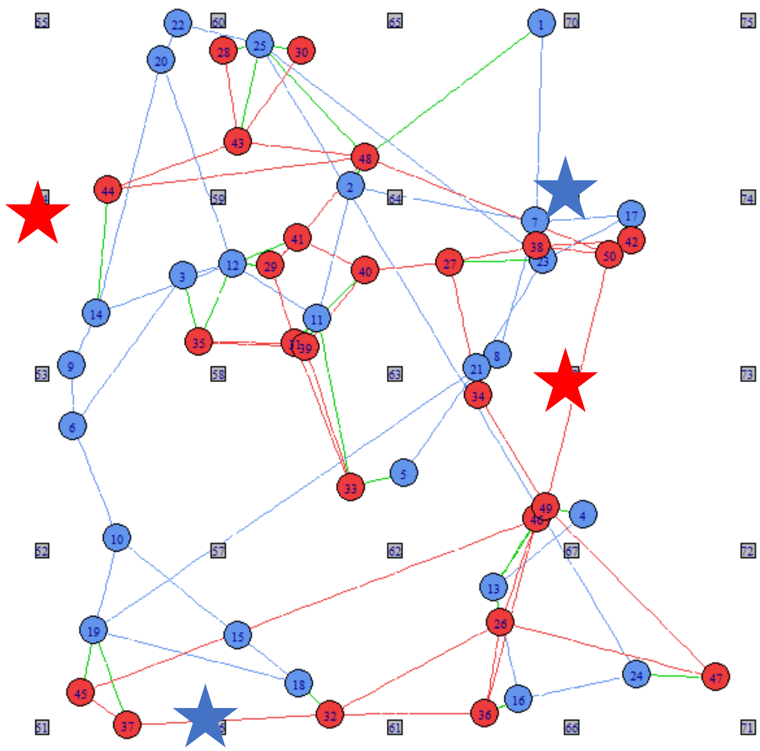


## Appendix C.3 Random Disruption Scenario

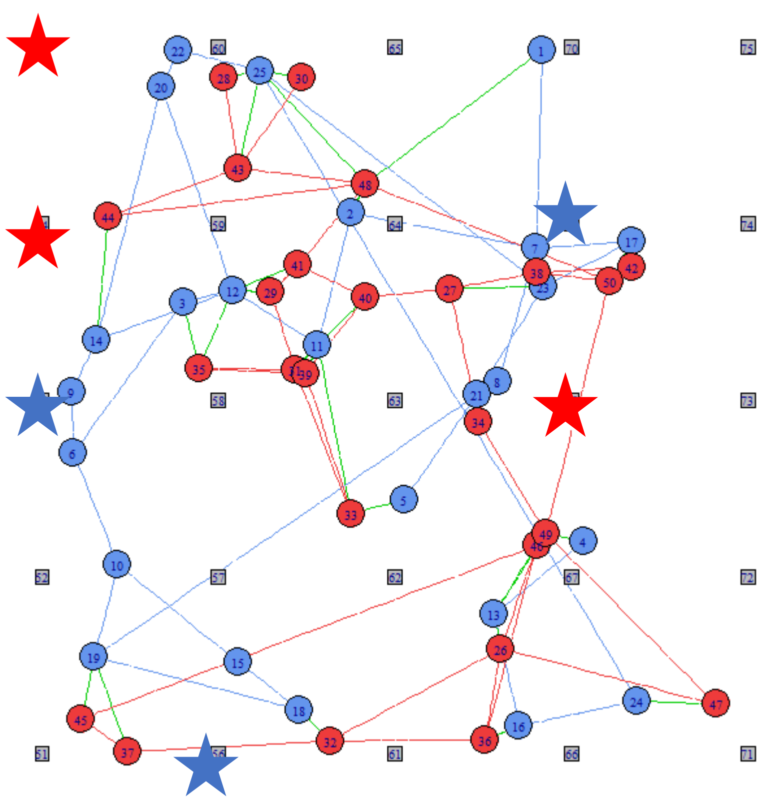
*Appendix C.3.1 1 Facility*



*Appendix C.3.2 2 Facilities*

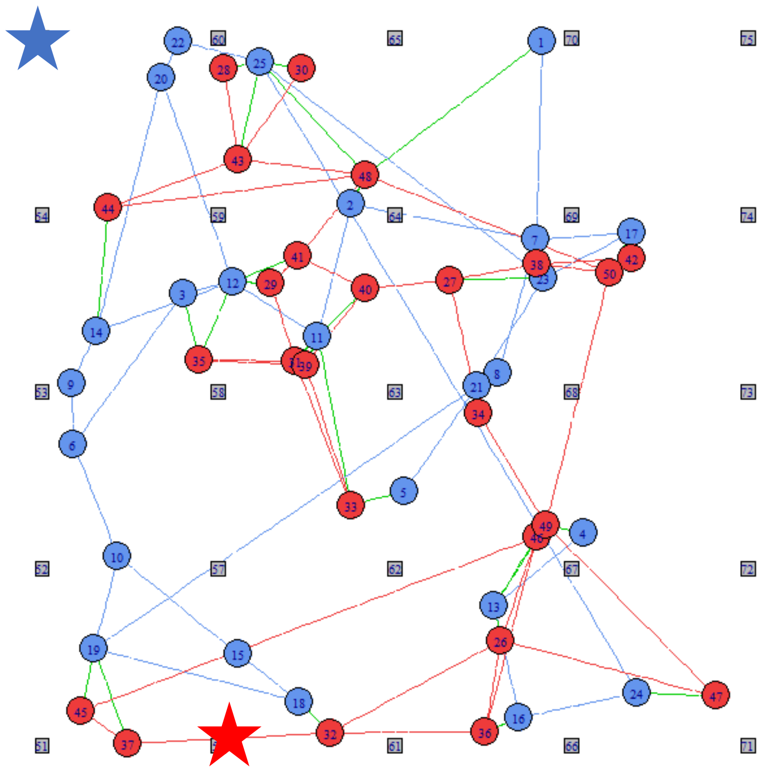



*Appendix C.3.3 3 Facilities*

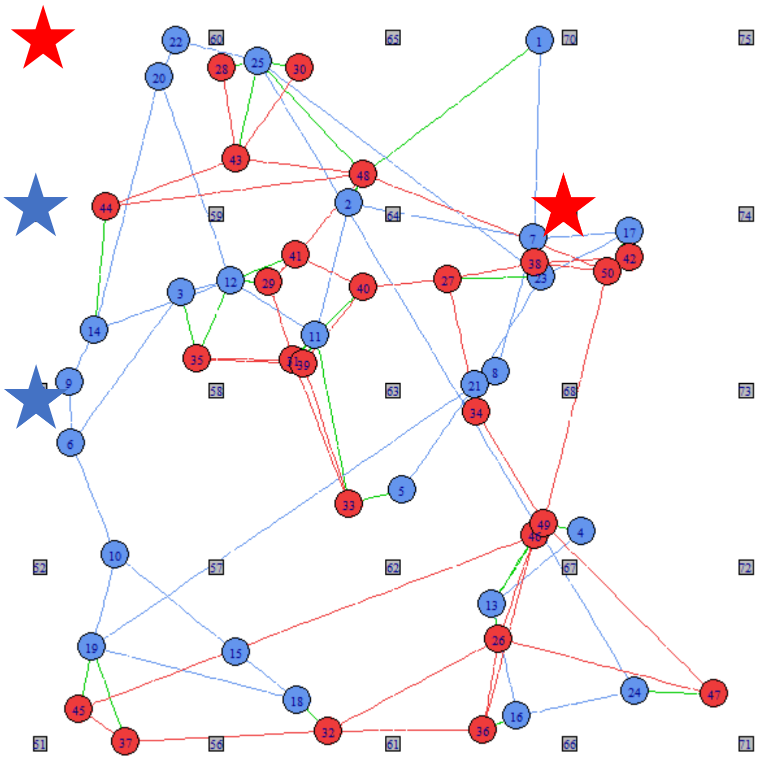


## Appendix C.4 Spatial Disruption Scenario

*Appendix C.4.1 1 Facility*



*Appendix C.4.2 2 Facilities*



*Appendix C.4.3 3 Facilities*