Integrating Restoration and Scheduling Decisions for Disrupted Interdependent Infrastructure Systems

Burak Cavdaroglu* Erik Hammel† John E. Mitchell‡ Thomas C. Sharkey§
William A. Wallace¶

Abstract

We consider the problem faced by managers of critical civil interdependent infrastructure systems of restoring essential public services after a non-routine event causes disruptions to these services. In order to restore the services, we must determine the set of components (or tasks) that will be temporarily installed or repaired, assign these tasks to work groups, and then determine the schedule of each work group to complete the tasks assigned to it. These restoration planning and scheduling decisions are often undertaken in an independent, sequential manner. We provide mathematical models and optimization algorithms that integrate the restoration and planning decisions and specifically account for the interdependencies between the infrastructure systems. The objective function of this problem provides a measure of how well the services are being restored over the horizon of the restoration plan, rather than just focusing on the

*Department of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, (cavdab@rpi.edu). The work of this author was supported by the National Science Foundation under grant number CMS-0301661 and by the US Department of Homeland Security under Award Number: 2008-ST-061-ND 0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the US Department of Homeland Security.

†Department of Mathematical Sciences, Rensselaer Polytechnic Institute, (hammyz@gmail.com).

‡Department of Mathematical Sciences, Rensselaer Polytechnic Institute, (mitchj@rpi.edu). The work of this author was supported by the National Science Foundation under grant number CMS-0301661.

§Department of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, (sharkt@rpi.edu).

¶Department of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, (wallaw@rpi.edu). The work of this author was supported by the National Science Foundation under grant number CMS-0301661 and by the US Department of Homeland Security under Award Number: 2008-ST-061-ND 0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the US Department of Homeland Security.

performance of the systems after all restoration efforts are complete. We test our methods on realistic data representing infrastructure systems in New York City. Our computational results demonstrate that we can provide integrated restoration and scheduling plans of high quality with limited computational resources. We also discuss the benefits of integrating the restoration and scheduling decisions.

1 Introduction

A disaster is a non-routine event that has the potential for catastrophic impacts on physical, natural and social systems. After a disaster occurs, one of the most important ways of limiting the impact of the disaster on society is the timely development of a plan to restore the services disrupted by the disaster. The managers of Critical Civil Infrastructure (CCI) systems are often faced with the most demanding choices in this so-called *restoration planning* since their systems deliver essential public services such as power, telecommunications, water, and transportation. The public not only hopes for a timely restoration of these services but needs them in order for society to recover from the disaster. Therefore, it is quite important to construct and develop decision technologies for these managers to support restoration and division of labor activities in response to non-routine events that disrupt the services of the CCI systems.

These CCI systems are more vulnerable to disasters and the services provided by them more difficult to restore due to the increasing amount of interdependencies among them; disruptions in one system can spread to others causing cascading and potentially catastrophic consequences (see Mendonca and Wallace [8] and Wallace et al. [14]). As an example, a central office (where calls are received and routed) of the telecommunications infrastructure requires a certain amount of power to function in its own infrastructure. A disruption of power to the central office will cause it to fail in the telecommunications infrastructure thereby causing disruptions in this infrastructure as well. Therefore, it is extremely important for the restoration plans of all the systems to consider the impact of their service on the restoration of services provided by the other systems. For example, a goal of the overall recovery effort of the systems could be to provide communications among police, fire departments, and hospitals. The telecommunications restoration plan will obviously work to restore this capability but if power is not restored to central offices in this infrastructure the goal of the overall recovery effort cannot be accomplished. This means that decentralized decision-making in planning the restoration efforts of the CCI systems may not allow for the timely restoration of essential public services.

In recent years, there has been much work about the concept, importance, challenges, and the complexities of the interdependencies of the CCI systems (see, e.g., O'Rourke [9]). Rinaldi et al. [11] discuss that managers of the CCI systems have become more inclined to consider the interdependencies among the CCI systems and conduct their restoration plan to restore all the essential public services provided by these interdependent infrastructure systems. In this paper, we will provide the managers of the CCI systems tools that can be used to develop plans to restore services provided by the interdependent infrastructure systems that specifically account for the interdependencies of the systems. Our approach is unique in that it measures the performance of the interdependent infrastructure systems over the horizon of the problem in order to understand how well the systems come back up online during the recovery efforts. This is opposed to focusing on the performance of the interdependent infrastructure systems only after all recovery efforts are completed.

There has been work in recent years that provides some level of decision support to the interdependent infrastructure systems managers in forming plans to restore essential public services. Lee et al. [5] develop an interdependent layered network (ILN) representation of the services delivered by the interdependent infrastructure systems. This ILN model can be used to measure the service outages from a disruption and is also used to determine the set of components (e.g., the arcs in the ILN) that need to be installed or repaired to restore all services. However, this work does not answer the critical questions of *how* and *when* to repair the arcs. Gong et al. [3] focus on answering these two questions given a set of arcs that need to be installed/repaired. They provide a Benders Decomposition approach to determine efficient solutions to the multi-objective scheduling problem whose objectives include the total cost incurred by the scheduling/assignment decisions, the total tardiness of the tasks, and the makespan of the schedule. However, this work does not answer the important question of *what* arcs need to be installed to restore the services.

The previous work on plans to restore services provided by interdependent infrastructure systems focuses on a single level of the decision-making, i.e., either the selection of components to install or repair that will restore the services (see Lee et al. [5]) or the assignment/scheduling decisions of installing/repairing the structures (see Gong et al. [3]). There are significant drawbacks to making these decisions sequentially, especially due to the conflicting nature of the objectives of the two problems. In particular, the metric used to evaluate the installation decisions is often the cost of the decisions, while the metric to evaluate the second set of decisions is often the time to complete the recovery efforts. This means that the recovery efforts may end up being inexpensive yet require a long time to complete, i.e., the essential public services will not be restored as efficiently as possible. This paper will provide an integrated model that incorporates the installation selection decisions (sometimes referred to as the *restoration planning* decisions), the assignment of the selected tasks to work groups, and the scheduling of the work groups to perform their set of assigned tasks. Furthermore, our formulation offers a unique performance measure of the scheduling decisions that measures how well the set of interdependent infrastructure systems come back online rather than focus on how late a particular task is completed (e.g., the classic tardiness measure of a task) or how long it takes us to complete all tasks (i.e., the classic makespan measure of a schedule).

The objective of this paper is to develop approaches to provide full restoration plans to managers that can be used to restore essential public services to disrupted interdependent infrastructure systems. We develop a mixed-integer program (MIP) that integrates the installation decisions and the assignment/scheduling decisions. This is a significant advantage of our approach over previous work on interdependent infrastructure systems (Lee et al. [5] and Gong et al. [3]). Our formulation of the MIP fully utilizes the problem characteristics as described by the managers of the CCI systems (see, e.g., Lee [4]). The objective function of this MIP measures how well the services provided by the interdependent infrastructure systems are restored over time by exploiting a network-flow based formulation of the services provided by a set of interdependent infrastructures systems at appropriate time intervals. However, it is possible that the managers of these systems do not have access to advanced commercial software packages in order to solve this MIP to formulate their restoration plans. There are free MIP solvers available that are almost as good as the commercial solvers, although these may have other drawbacks. For example, MIPs can be submitted online to the NEOS server [2] provided the data is not confidential, and the open source suite of COIN-OR solvers [7] is generally of high quality, although a new user requires some time to learn the software. Therefore, we provide a heuristic solution method that provides quality solutions to this problem without the aid of commercial software packages. This heuristic solution takes full advantage of the underlying structure of the proposed model: it first selects a set of arcs to install that help to restore services and then schedules these arcs using 'classic' dispatching rules (see Pinedo [10]) for scheduling problems.

The remainder of this paper is organized as follows. Section 2 discusses the mathematical details of the ILN model of Lee et al. [5] and the MIP formulation of the integrated restoration and scheduling problem. Section 3 discusses the heuristic solution method that we propose for the problem. The formulation and solution method is then tested on realistic data that represents a large portion of Manhattan. This data was developed from the input from the managers of various CCI systems, including the power and telecommunication systems. The results show that our approaches can be used to generate an integrated restoration plan of high quality using only a modest amount of computing time on a laptop. This is especially important for the managers of the systems since they may not have extensive computational resources available to them. The results of the computational testing are discussed in Section 4. We conclude the paper in Section 5 with some future research directions.

2 The Integrated Restoration and Scheduling Model

In this section, we discuss a mathematical formulation that can be used to develop an integrated restoration and scheduling plan to restore services to disrupted interdependent infrastructure systems. In this problem, we are interested in determining the set of components that will be installed (i.e., the restoration decisions), the assignment of selected components (the tasks) to available work groups, and the order that each work group will complete the tasks assigned to them. The objective of this problem will be concerned with the functionality of the set of systems throughout the horizon of the problem. In each time period, the set of systems will function in a manner that minimizes its total operating costs and unmet demand costs. Therefore, our objective will implicitly seek to minimize the services that are disrupted or 'down' for long periods of time after the non-routine event. The objective function is a weighted combination of three different types of costs: (i) the cost of operating the infrastructures at each time period, (ii) the cost of unmet demand at each time period, and (iii) the cost of restoring the infrastructures. The explicit inclusion of the cost of unmet demand is a novel aspect of this research and captures the desire of society for the expedient restoration of services. It will thus be necessary to apply an appropriate model that can measure the functionality of the set of systems in each time period, and in particular capture costs (i) and (ii). We will apply the ILN model of Lee et al. [5] in order to measure this functionality. This model is refined through inclusion of both the scheduling aspects and the temporal aspects.

The ILN model is a network-flow based representation of the services provided by these infrastructure systems at a given point in time. In particular, the 'flow' in a layer of the network corresponds to the services provided by that infrastructure. For example, the power system has a single-commodity flow of electricity, with demand points, supply points (eg, generators or the entry points of the network into the area of interest), and transshipment nodes (eg, substations), along with power lines represented by arcs. The telecommunications infrastructure is formulated as a multi-commodity network flow model, with different commodities for each origin-destination pair [1]. This model can capture various types of interdependencies between the infrastructure systems (see Lee et al. [6]). However, for the purposes of this paper, we focus on the most common of these interdependencies and the one that appears in our case study of the power and telecommunications infrastructure of lower Manhattan: the so-called *input* interdependency. This interdependency occurs when a node in one infrastructure requires the services of another infrastructure in order to operate. A central office in the telecommunications infrastructure has an input interdependency with the power infrastructure: if its full required amount of power is not delivered to it, the central office does not operate.

The ILN model of Lee et al. [5] will be applied in each time period to determine the total operating costs and unmet demand costs over the current state of the set of interdependent infrastructure systems. Each infrastructure will be represented as one of the networks in the ILN model. The network representation of the infrastructure has typical characteristics of network flow problems (see Ahuja et al. [1]): each node has a certain amount of supply/demand of each type of commodity in the network and we seek to send flow (e.g., services in the infrastructure) from supply nodes to demand nodes over the arcs in the network. The demand nodes in the network are split into two types: 'standard' demand nodes and 'interdependent' demand nodes. The interdependent demand nodes are those on which components of another infrastructure have an input interdependency. This means that a demand node in a power network that feeds power to a central office in the telecommunications infrastructure is an interdependent demand node.

It is appropriate to penalize shortcomings in the services to a standard demand node on a per-unit basis. As discussed in Lee et al. [5], we need to treat disruptions in services to interdependent demand nodes differently. We penalize shortcomings in the services to an interdependent demand node with a fixed-charge cost regardless of the magnitude of the shortcoming. In other words, we are penalized the same amount if we meet one percent or ninety-nine percent of the demand of an interdependent node. The reasoning, as discussed by Lee et al. [5], is that the component of the *other* infrastructure with the input interdependency will not be operational in its own infrastructure. The ILN model introduces binary variables for these interdependent demand nodes that represent whether the demand of that node is fully satisfied. These variables are then used to determine the appropriate penalty costs *and* whether the component of the other infrastructure is operational in its own infrastructure.

Our model contains four types of constraints: (a) network flow constraints in each infrastructure, with the explicit inclusion of slack variables representing unmet demand, (b) constraints to represent the logical input dependencies between different infrastructures, enforced using binary variables, (c) constraints to represent the scheduling decisions in the restoration process, and (d) a constraint to link the scheduling decisions to the availability of arcs. The first two types of constraints are handled as in the ILN model, with copies constructed for each time period. We now discuss the formulation of the scheduling part of the model.

Our integrated restoration and scheduling problem will select a set of arcs to install or repair in the infrastructure systems. It will then schedule the selected arcs (or tasks) on a series of work groups. This involves assigning each selected task to a work group and then determining the sequence in which a work group will perform the tasks assigned to it. In a scheduling context, we are essentially solving an *unrelated parallel machine scheduling* problem. These types of scheduling problems are typically NP-hard (see Pinedo [10]) and notoriously difficult to solve to optimality. Typical approaches to formulate parallel machine scheduling problems where the sequence of the tasks is important (for example, when we are minimizing total completion time or tardiness) is to incorporate a binary variable that represents the decision of performing a certain task as the i-th task that a particular machine completes. However, we will formulate the scheduling decisions somewhat differently using the so-called time-indexed formulation for these problems (see Sousa and Wolsey [13] and Savelsbergh et al. [12]). The formulation that we offer is motivated by our discussions with the managers of the interdependent infrastructure systems. From these discussions, we learned that the completion times of the tasks are typically expressed as integral number of days and that the planning horizon of the restoration activities is relatively small (between 30-60 days). Further, the number of potential tasks is quite large compared to the number of days in the problem. These facts motivate a formulation of the scheduling decisions that focus on determining which task a work group completes on a particular day rather than the order of the tasks it will complete.

We will enforce the practical restriction that a work group must work on a particular task on consecutive days until it is completed (i.e., we are not allowing preemptions).

We will now formally discuss the input parameters of our integrated restoration and scheduling problem. We are given a set of infrastructures $\mathcal{M}$ and a set of time periods which we denote as $T = \{1, \ldots, \tau\}$. For each infrastructure $m \in \mathcal{M}$, we have a set of nodes, V^m, a set of permanent (or unaffected by the disruption) arcs E^m, and a set of temporary arcs that can be installed, $\bar{E}^m$. For the sake of brevity, our mathematical formulation of the integrated restoration and scheduling problem only includes single-commodity flows. However, it can be easily modified to handle infrastructures having multi-commodity flows in both directions by adding a '*commodity*' index to the necessary variables and parameters in the model. The amount of supply at a node $i \in V^m$ is denoted by b_i^m. We will let $V^{m,+}$ be the set of supply nodes in infrastructure m (i.e., $b_i^m > 0$), $V^{m,-}$ be the set of demand nodes in infrastructure m (i.e., $b_i^m < 0$), and $V^{m,=}$ be the set of transhipment nodes in infrastructure m (i.e., $b_i^m = 0$). We will let h_{it}^m denote the unit penalty cost of unmet demand at node $i \in V^{m,-}$ in time period t and w_i^m denote the transhipment capacity of node $i \in V^{m,=}$. We denote c_e^m for $e \in E^m \cup \bar{E}^m$ as the cost of a unit of flow on arc e in infrastructure m and u_e^m be the upper bound of flow on arc e in infrastructure m. Given node $i \in V^m$, we denote $\delta^{m,+}(i)$ as the set of arcs entering i, $\delta^{m,-}(i)$ as the set of arcs leaving i, and $\delta^m(i)$ as the union of these sets.

For infrastructure m, we will let B^m denote the set of interdependent demand nodes in the infrastructure, i.e., a node in another infrastructure has an input interdependency with a node in B^m. Further, we will let A^m be the set of nodes in infrastructure m that have input interdependencies with another infrastructure. We denote the interdependencies between infrastructure m and n as $F(m,n)$, i.e., $(i,j) \in F(m,n)$ implies that $i \in B^m$ and $j \in A^n$ so that node j in infrastructure n is dependent on demand node i in infrastructure m. We also refer to node i as the parent node of node j.

We let q_e^m for $e \in \bar{E}^m$ be the fixed cost of installing temporary arc e in infrastructure m. We denote the set of work groups by K. The cost of assigning task e in infrastructure m to work group $k \in K$ is denoted by $d_{e,k}^m$ and the time required by work group k to complete this task is denoted by $p_{e,k}^m$.

The objective function contains elements that have direct monetary costs (operating costs and restoration costs) and elements that correspond to the loss of a service (the unmet demands). In order to place these elements into a single objective, we choose weights for the unmet demands. In particular, we let $h_{i,t}^m$ denote the weighting given to the slack at node i of infrastructure m at time t. The choice of these weights is discussed in Section 4.1.

We are now in a position to discuss the decision variables in our integrated restoration and scheduling problem. In each time period, we will need to determine the amount of flow on the arcs in each of the infrastructure systems to evaluate the level of services throughout the interdependent infrastructure systems. We denote the flow on arc $e \in E^m \cup \bar{E}^m$ in time period t as $x_{e,t}^m$. We will denote the amount of unmet demand (sometimes referred to as *slack*) of demand node $i \in V^{m,+}$ in time period t as $s_{i,t}^m$. For each pair $(i,j) \in F(m,n)$, we define a binary variable $y_{m,i}^{n,j,t}$ that is equal to 1 if the slack at the parent node i in infrastructure m is zero and thus the child node j in infrastructure n is operational. For each arc $e \in \bar{E}^m$, we define the binary variable z_e^m to represent the decision to install the temporary arc e in infrastructure m and the binary variable $a_{e,k}^m$ to represent the decision of assigning the task of installing the temporary arc to work group $k \in K$. In modeling the scheduling decisions of each work group, we define the binary variable $\alpha_{e,k,t}^m$ that is equal to 1 if work group k completes task $e \in \bar{E}^m$ during time period t. We also define the binary variable $\beta_{e,t}^m$ that is equal to 1 if we have completed task $e \in \bar{E}^m$ by time period t.

The mixed-integer programming formulation of our integrated restoration and scheduling problem is given by:

$$\text{minimize} \quad \sum_{t \in T} \left[\sum_{m \in M} \sum_{e \in E^m \cup \bar{E}^m} c_e^m x_{e,t}^m \right] \tag{1}$$

$$+ \sum_{t \in T} \left[\sum_{m \in M} \sum_{i \in V^{m,-} \setminus B^m} h_{i,t}^m s_{i,t}^m \right] \tag{2}$$

$$+ \sum_{t \in T} \left[\sum_{m \in M} \sum_{i \in B^m} \sum_{n \in M, n \neq m} \sum_{(i,j) \in F(m,n)} h_{i,t}^m (-b_i^m)(1 - y_{m,i}^{n,j,t}) \right] \tag{3}$$

$$+ \sum_{m \in M} \sum_{e \in \bar{E}^m} q_e^m z_e^m + \sum_{m \in M} \sum_{e \in \bar{E}^m} \sum_{k \in K} d_{e,k}^m a_{e,k}^m \tag{4}$$

subject to

$$\sum_{e \in \delta^-(i)} x_{e,t}^m \leq b_i^m \quad \forall t \in T, \forall i \in V^{m,+}, \forall m \in M \tag{5}$$

$$s_{i,t}^m + \sum_{e \in \delta^+(i)} x_{e,t}^m = -b_i^m \quad \forall t \in T, \forall i \in V^{m,-}, \forall m \in M \tag{6}$$

$$\sum_{e \in \delta^+(i)} x_{e,t}^m - \sum_{e \in \delta^-(i)} x_{e,t}^m = 0 \quad \forall t \in T, \forall i \in V^{m,=}, \forall m \in M \tag{7}$$

$$\sum_{e \in \delta^+(i)} x^m_{e,t} \leq w^m_i \quad \forall t \in T, \, \forall i \in V^{m,=}, \, \forall m \in M \tag{8}$$

$$x^m_{e,t} \leq u^m_e \quad \forall t \in T, \, \forall m \in M, \, \forall e \in E^m \tag{9}$$

$$s^m_{i,t} \leq (1 - y^{n,j,t}_{m,i})(-b^m_i) \quad \forall t \in T, \, \forall (i,j) \in F(m,n) \tag{10}$$

$$\sum_{e \in \delta^-(j)} x^n_{e,t} \leq b^n_j y^{n,j,t}_{m,i} \quad \forall t \in T, \, \forall (i,j) \in F(m,n) \, with \, j \in V^{n,+} \tag{11}$$

$$\sum_{e \in \delta^+(j)} x^n_{e,t} \leq -b^n_j y^{n,j,t}_{m,i} \quad \forall t \in T, \, \forall (i,j) \in F(m,n) \, with \, j \in V^{n,-} \tag{12}$$

$$\sum_{e \in \delta^+(j)} x^n_{e,t} \leq w^n_j y^{n,j,t}_{m,i} \quad \forall t \in T, \, \forall (i,j) \in F(m,n) \, with \, j \in V^{n,=} \tag{13}$$

$$\sum_{k \in K} a^m_{e,k} = z^m_e \quad \forall m \in M, \, \forall e \in \bar{E}^m \tag{14}$$

$$\sum_{t \in T} \alpha^m_{e,k,t} = a^m_{e,k} \quad \forall k \in K, \, \forall m \in M, \, \forall e \in \bar{E}^m \tag{15}$$

$$\beta^m_{e,t} - \beta^m_{e,t-1} = \sum_{k \in K} \alpha^m_{e,k,t} \quad \forall t \in T, \, m \in M, \, \forall e \in \bar{E}^m \tag{16}$$

$$\sum_{e \in \bar{E}^m} \sum_{s=t}^{\min\{T, t+p^m_{e,k}-1\}} \alpha^m_{e,k,s} \leq 1 \quad \forall t \in T, \, \forall k \in K \tag{17}$$

$$x^m_{e,t} \leq u^m_e \beta^m_{e,t} \quad \forall t \in T, \, \forall m \in M, \, \forall e \in \bar{E}^m \tag{18}$$

$$x^m_{e,t} \geq 0 \quad \forall t \in T, \, \forall m \in M, \, \forall e \in E^m \tag{19}$$

$$s^m_{i,t} \geq 0 \quad \forall t \in T, \, \forall m \in M, \, \forall i \in V^m \tag{20}$$

$$y^{n,j,t}_{m,i} \in \{0,1\} \quad \forall t \in T, \, \forall m,n \in M, \, \forall (i,j) \in F(m,n) \tag{21}$$

$$z^m_e \in \{0,1\} \quad \forall m \in M, \, \forall e \in \bar{E}^m \tag{22}$$

$$a^m_{e,k} \in \{0,1\} \quad \forall m \in M, \, \forall e \in \bar{E}^m, \, \forall k \in K \tag{23}$$

$$\alpha^m_{e,k,t} \in \{0,1\} \quad \forall t \in T, \, \forall m \in M, \, \forall e \in \bar{E}^m, \, \forall k \in K \tag{24}$$

$$\beta^m_{e,k,t} \in \{0,1\} \quad \forall t \in T, \, \forall m \in M, \, \forall e \in \bar{E}^m, \, \forall k \in K \tag{25}$$

The objective function of this model minimizes the sum of (i) the flow costs and (ii) the unsatisfied demand costs over the whole planning horizon of the restoration activities, and (iii) the sum of the arc installations and assignment costs. The term (1) represents the total flow costs incurred over the planning horizon while (2)-(3) represents the cost of the unsatisfied demands. We note that (2) represents the cost of the unsatisfied demands of nodes in a particular infrastructure that do not impact the other interdependent infrastructures. The term (3) represents the costs of unsatisfied demands at a nodes in an infrastructure that are 'parent' nodes to nodes of other infrastructures. In this case, we are penalized the total amount of the demand for not meeting the requested demand of the parent node in its infrastructure system since we cannot operate the child node. The term (4) represents the arc installation costs and the assignment costs of tasks to work groups.

Constraints (5)-(9) are network-flow constraints for each infrastructure system that ensure we do not use more than the available supply of nodes $i \in V^{m,+}$, measure the amount of unmet demand for nodes $i \in V^{m,-}$, and ensure flow-balance for transhipment nodes $i \in V^{m,=}$. Capacity constraints are also imposed on the arcs in E^m.

Constraints (10)-(13) represent the interdependencies between the infrastructures. Constraint (10) ensures that child node $j \in V^n$ is operational given the interdependence $(i, j) \in F(m, n)$ only if the unmet demand of node $i \in V^{m,+}$ is zero. Constraints (11)-(13) guarantee that we only operate children nodes in an infrastructure if the unmet demands of their parents are zero (i.e., $y^{n,j,t}_{m,i} = 1$).

Constraints (14)-(17) are associated with the assignment and scheduling decisions. Constraint (14) guarantees that we assign each arc that we choose to install to a work group. Constraint (15) ensures that if we complete an arc, then we must have selected it (thus guaranteeing that we pay the installation costs associated with the arc). Constraint (16) ensures that: (i) an arc only becomes available after it is completed by some work group and (ii) if an arc is available in time period $t - 1$, it is available in time period t. We note that these constraints ensure that once an arc is completed, it is available for the remainder of the horizon. Constraint (17) ensures that, at most, one arc is being processed on work group k in time period t. This is because if work group k completes arc e in period s, then we must have been working on it in periods $s - p^m_{e,k} + 1$ through period s.

Finally, constraint (18) links together the scheduling parts of the model and the network flow parts of the model. It ensures that we only send flow over the arc after it is finished being installed. The overall logic is that if an arc has not been completed by time t then its β variable is zero, which forces the flow on that arc to be zero (18). In turn, that may result in unmet demand in the parent infrastructure, with some s variable strictly positive (6), which can then force a y variable to be zero (10). If a y variable is zero then flow variables in the child network may be forced to be zero (11)-(13), leading to unmet demand in the child infrastructure (6).

This formulation of the integrated restoration and scheduling problem is a large-scale mixed-integer program with underlying scheduling decisions associated with it. These types of problems are known to be computationally difficult. Therefore, we turn our attention now to developing a customized heuristic for this class of problems that integrates network design and scheduling heuristics together.

3 An Integrated Design and Scheduling Heuristic Solution Method

The integrated restoration and scheduling problem has components representing network flow decisions (i.e., the ILN model), network design decisions (i.e., which arcs we will install into the network), and scheduling decisions (i.e., allocating the work groups to complete the selected arcs). Therefore, it is desirable for heuristics that are applied to this problem to utilize the structure of these underlying decisions. We will propose a heuristic solution method that operates in three distinct phases, each of which correspond to one of these underlying sets of decisions. These phases will be referred to as the 'network flow,' 'network design,' and 'scheduling' phases. The network flow phase focuses on determining the operations of the network. The network design phase focuses on using the current operations to determine which arc(s) is 'best' to install into the current network. The scheduling phase then focuses on scheduling these selected arcs on the work groups.

The idea behind the network flow phase of the heuristic is to route flow in each of the networks in the ILN model. There is obviously much flexibility in the type of method used to route flow (e.g., we can solve the ILN model to optimality). We will describe a method that we have used that has synergy with the network design phase. This method determines the flows in each network independently of one another but 'links' them by assigning priority levels to the demand nodes within the network. The routing procedure will then ensure that we can meet demand at high priority nodes effectively. The priority levels can be set to reflect the interdependencies with other infrastructures. For example, we can assign the priority levels of the interdependent demand nodes to be higher than standard demand nodes within a network and even assign the priority levels based on the perceived importance of the interdependent demand node within the other infrastructure. The routing procedure proceeds as follows: it determines the minimum cost flow in order to meet the demand of the node with the highest priority. It then fixes the flow throughout the network and then determines the minimum cost flow to meet the demand of the node with the second highest priority. It then updates and fixes the flow in the network and solves a minimum cost flow problem to meet the demand of the third highest priority node. We note that it is possible that not all the demand of a node can be met - this corresponds to a disruption in the service. It is also possible to modify this procedure by placing different penalty weights for unmet demand based on the priority levels of the nodes and solving one (large) minimum cost network flow problem.

The network design phase then focuses on selecting the next arc, or set of arcs, to install into the network. It is obvious that we want to select an arc (or arcs) that improve the total of the operational costs plus the unmet demand costs in the ILN model. The network design phase of the heuristic will then estimate the improvement in the performance (measured via the costs) of the ILN model of installing an arc (or arcs) into the network under consideration. The design phase then selects the 'best' arc measured by the ratio of the improvement in the performance to the processing time of the arc. Note, therefore, the design phase may actually be called upon during the scheduling phase of the heuristic when work groups become available. The selection of an arc is thus based on the classic weighted shortest processing time (WSPT) first dispatching rule for parallel machine scheduling problems (see, e.g., Pinedo [10]). This phase, again, has flexibility in terms of how the improvement in the performance is measured for each arc. It may not be possible to determine the *exact* improvement by installing an arc into one of the networks because this requires solving the ILN model. The proposed method to estimate the improvement has synergy with the network flow phase of the heuristic: once we have the flows routed in the network under consideration, we select the highest priority demand node that has a disruption in service (i.e., unmet demand). The estimate of the improvement for installing an arc into the ILN model will then be measured via the amount of additional flow that can be delivered to this demand node.

The description of the network design phase is general in the sense that it can be applied to both small-scale and large-scale events that cause the disruptions. However, in the case of large-scale events, it may be that entire portions of the infrastructure are destroyed by the event. This is the situation in the case study discussed in Lee et al. [5], which represents the failure of components in and around the Brooklyn-Battery Tunnel in lower Manhattan. In these situations, the network representing an infrastructure can be viewed as essentially two distinct components: the 'real' infrastructure which contains the components that were unaffected by the extreme event and the 'temporary' infrastructure which contains the components that can be installed/repaired in the network. It is often the case in these situations that the installation of a *single* arc in the temporary infrastructure cannot help improve the performance of the set of interdependent infrastructure systems - we must install a path of arcs in the temporary infrastructure that connects some supply node back into the operational infrastructure. Therefore, the network design phase, as currently described, will rarely identify an arc that improves the performance of the ILN model.

In the case of large-scale events, however, we can modify the network design phase to select a 'path' of arcs in the temporary infrastructure. Each path of arcs in the temporary infrastructure has a capacity which can be measured as the minimum capacity of an arc on the path. The path also has an associated processing time: the total of the processing times of the uninstalled arcs on it. Therefore, we can aggregate paths of arcs in the temporary infrastructure into a single arc and then apply the same rule as the one for small-scale events (i.e., determine the estimate for improvement for each aggregated arc). We have chosen the following procedure to aggregate the paths of arcs in the temporary infrastructure: for each node in the real infrastructure, we determine the shortest path, with respect to processing times, from a supply node through the temporary infrastructure to that node. We then only consider arcs that represent the shortest path from some supply node through the temporary infrastructure to some node in the real infrastructure. This greatly reduces the number of paths that need to be considered and is also practically motivated since the arc from the temporary infrastructure into the real infrastructure is usually the bottleneck of the paths in our case study. There is an issue of determining the appropriate processing times of a path of arcs since we can expect that they will not be processed by the same work group. For each arc, we have chosen to assign its processing time as its average processing time across the different work groups.

The scheduling phase of the heuristic is concerned with the assignment of arcs to work groups. It will actually call upon the network design phase to determine the next arc or set of arcs to be processed. The scheduling phase will operate differently depending on if the network design phase determines a single arc (i.e., the small-scale event situation) or a set of arcs (the large-scale event situation). For the small-scale event situation, the scheduling phase simply calls the network design phase whenever a work group becomes available. We note that the network design phase will assume that the arcs being processed by other work groups are available in the network in order to ensure that we move on to other demand nodes if we are processing arcs to restore services to a particular node. For the large-scale event situation, the scheduling phase will keep a queue of arcs that need to be processed by the work groups. When a work group becomes available, it will process the next arc in the queue if it is non-empty. Otherwise, the network design phase will be called to determine the next path of arcs in the temporary infrastructure to process. The selected arcs will then be placed in the queue. There is flexibility in placing the selected arcs in the queue:

for example, we could determine the placement of arcs into the queue that minimizes the makespan of the selected arcs. We could also place the arcs into the queue according to their longest average processing time.

It turns out that each of these different phases are actually embedded into one another: the scheduling phase calls the network design phase, which can call the network flow phase. Since there is flexibility in each of the phases, we will summarize the heuristic solution method that is applied in Section 4.2.2:

Specialized Heuristic Solution Method

Step 0. *Initialization*: Determine the flow in the infrastructure under consideration by successively routing flow to demand nodes based on their priority levels. This determines the amount of unmet demand at each node. Refer to this as the 'current operations of the network.' Initialize the queue of arcs to be empty.

Step 1. *Scheduling Phase*: We perform the scheduling phase until either the end of the horizon or all possible demand is met. When a work group becomes available:

- If there is an arc in the queue, assign that arc to the available work group.
- Otherwise, we call the *network design phase* for 'large-scale' events to determine the next set of arcs to be processed. We call this phase assuming that all arcs currently being processed are available in the network by focusing on the 'current operations of the network.' This phase is also called on the demand node with unmet demand and the highest priority level. Once the network design phase returns a path of arcs:
  - Update the current operations of the network by routing flow over the select path to the appropriate demand node.
  - Place the uninstalled arcs in the path into the queue according to their largest average processing time.

4 Experimental Results

4.1 Data Set

We will test the mathematical formulation of the integrated restoration and scheduling problem and the heuristic for it on a data set that is a realistic representation of the power and telecommunication systems of a large portion of Manhattan. Lee et al. [5] developed a realistic representation of these systems through discussions with and data obtained from respective infrastructure managers in Manhattan. The empirical results in this section were generated by applying our mathematical formulation of the integrated restoration and scheduling problem and our heuristic to this realistic data set. The model has been implemented in the OPL Development Studio 6.3 and we used CPLEX 12.1 to solve the mixed-integer programming formulation of the problem.

In our empirical study, we will focus on a situation, similar to Lee et al. [5], where the set of damaged/affected arcs occurs solely in the power infrastructure. Note that this does not mean that the disruption only affects the power infrastructure because the interdependencies between the infrastructures will cause disruptions in the telecommunications infrastructure. This does mean that we only need to make the restoration decisions for the installation of temporary arcs in the power infrastructure; however, we still apply the network flow decisions to both infrastructures to determine the performance of our restoration efforts.

The size of the power and telecommunications infrastructures in the data set can be represented by the number of nodes and arcs of each infrastructure network. The power infrastructure is represented by 3316 arcs (E^p) and 1810 nodes (V^p) in the data set. As was done in Lee et al. [5], we assume that all possible components that can be installed to restore the power services can be represented as *power arcs*, i.e., we cannot install new power nodes. There are a total of 695 temporary arcs in the power infrastructure. Many of these arcs will not be chosen in any restoration plan, so we perform some pre-processing in order to generate the set of temporary arcs in our mathematical model. In particular, we determine the selected arcs in the restoration plan of the ILN model used by Lee et al. [5]. In total, 49 arcs are selected from the 695 temporary arcs. We will include these 49 arcs in $\bar{E}^p$, the set of temporary arcs in our model. We then remove these 49 arcs from the set of 695 temporary arcs and reapply the ILN model. In this situation, 92 temporary arcs are selected. We randomly selected a subset of 52 of these 92 temporary arcs and place them in $\bar{E}^p$. Therefore, the set of temporary arcs in our mathematical model includes 101 of the 'best' temporary arcs of the 695 original temporary arcs. There are 15 supply nodes ($V^{p,+}$), 134 demand nodes ($V^{p,-}$), and 1661 transhipment nodes ($V^{p,=}$) in the power system. Of the 134 demand nodes, 17 of these nodes serve as feeds for a distinct central office of the telecommunications infrastructure. In other words, the set B^p contains these 17 nodes.

The telecommunications infrastructure is represented by 1097 arcs (E^t) and 547 nodes (V^t) in the data set. There are 17 nodes of the 547 nodes that correspond to the central offices in the system. The proper functionality of central offices is directly dependent upon the power infrastructure because of their one-to-one connection to 17 power demand nodes mentioned above. According to the definition of input interdependency provided by Lee et al. [5], any unsatisfied demand (slack) in a power node connected to a central office would result in complete failure of the central office. This situation is also modeled in our mixed-integer programming formulation through constraints (10)-(13). Every call is routed to its local central office and then to the local central office for its recipient, and then on to its recipient. We will make the simplifying assumption that the set of supply and demand nodes of the telecommunications infrastructure system are just the 17 central offices. In other words, all phone calls that originate in a certain (geographical) location will be viewed as the supply of one of the central offices and all phone calls that terminate in the same geographical location will be viewed as the demand of the same central office. Unlike the power infrastructure, the calls in the telecommunications system are modeled as multi-commodity flows with a certain number of calls between each origin and destination pair.

We will consider two classes for the penalty costs of unmet demands in the integrated restoration planning and scheduling problem. In both these classes, the per-unit penalty cost of a unit of demand in a time period will be independent of the demand node and infrastructure, i.e., we have $h_{it}^m = h_t$ for $i \in V^{m,-}$ and $m \in M$. We will consider a class of penalty costs where the penalty cost of unmet demands is constant over the horizon, i.e., $h_{it}^m = h$ for $i \in V^{m,-}$, $m \in M$, and $t \in T$. This class of constant penalty costs can be used in situations where each time period is of equal importance. The second class of penalty costs are ones that are discounted by the horizon of the problem, i.e., $h_{it}^m = h(\tau - t)/\tau$ for $i \in V^{m,-}$, $m \in M$, and $t \in T$ where τ is the number of time periods in the problem. This means that the the incurred penalty is discounted by the period in which it takes place. This class of penalty costs can be used in situations in which it is critical to get large portions of services back up and running in a reasonable time frame.

Workers and equipment are bundled into work groups and have sufficient skills to accomplish any of the tasks in the set. In this empirical study, we have assumed *3 work groups* in total which is motivated by the discussions with the managers of the infrastructure systems. It is also assumed that each task only requires one work crew. The main differences between the work groups are that they will require different times to complete each task. We have assumed that the work groups are currently available to the managers, so that there are no assignment costs in the problem. We assume that the planning horizon of the restoration plan is 30 days, which the managers indicated is an acceptable time to perform the restoration activities.

We performed all of our experiments using a laptop with a Intel Core 2 Duo processor operating at 2.26 GHz and 4 GB of RAM. These experiments therefore use computing resources similar to those that would be available to the managers of the interdependent infrastructure systems during the time frame in which they need to formulate the restoration plans. This is quite important in determining the applicability of our model and methods to providing a decision support tool for these managers.

4.2 Experiments

4.2.1 The Value of Integration

We first seek to identify the potential value of integrating the restoration and scheduling decisions. In other words, we wish to determine the amount of additional costs that would be incurred by first determining the restoration decisions and then determining the scheduling decisions sequentially. The previous research of Lee et al. [5] focused on determining the restoration decisions of the power infrastructure in order to restore the interdependent infrastructure systems of power and telecommunications in lower Manhattan. We are, therefore, concerned with comparing the optimal schedule of the restoration decisions of Lee et al. [5] with the optimal solution to our integrated restoration and scheduling problem. The mixed-integer programming formulation for this set of interdependent infrastructure systems contains 1,267,950 variables (1,255,830 continuous variables and 12,120 binary variables) and 300,202 constraints, which is significantly larger than numbers for ILN model used in Lee et al. [5]. The reason for this increase is because our integrated restoration and scheduling problem includes network flow decision variables for every time period in the problem as opposed to a single time period in the ILN model. Note that, due to the large number of variables, CPLEX 12.1 had issues with memory management and did not solve the integrated restoration and scheduling problem to optimality for any problems considered.

We determine the restoration and scheduling plan associated with the sequential approach by first determining the selection of temporary arcs to be installed by solving the ILN model used in Lee et al. [5] and then optimizing the scheduling decisions using our formulation and objective function from Section 2 with the appropriate penalty costs (either constant or discounted). The ILN model finds a total of 45 temporary arcs to be installed. In order to determine the value of integration, we would ideally compare the optimal schedule for these arcs and the optimal integrated restoration and scheduling plan. However, CPLEX 12.1 terminates prior to determining the optimal integrated restoration and scheduling plan due to the amount of memory required in solving the problem. In particular, a solution with an optimality gap of 1.17% was found after 55 seconds for the model with constant penalty costs. The additional costs incurred by the sequential decision-making approach to forming the full restoration and scheduling plan was 7.76% higher than the best-known restoration and scheduling problem identified by CPLEX 12.1. Therefore, the value of the integration is at least 7.76% for the problem with constant penalty costs.

For the model with discounted penalty costs, a solution with an optimality gap of 1.31% was found within 47 seconds. The additional costs incurred by the sequential decision-making approach to forming the full restoration and scheduling plan was 5.63% higher than the best-known restoration and scheduling problem identified by CPLEX 12.1. This means that the value of the integration is at least 5.63% for the problem with discounted penalty costs. Therefore, there are significant cost benefits for determining the restoration planning and scheduling decisions in an integrated manner over the traditional manner of sequential decision-making to these decisions.

4.2.2 Full Integrated Restoration and Scheduling Plans for the Power and Telecommunications Systems

We now seek to provide an integrated restoration and scheduling plan that specifically incorporates the input interdependencies between the power and the telecommunications systems. This section discusses the results of applying our specialized heuristic solution method to this integrated restoration and scheduling problem. We do so by benchmarking our heuristic solution method with respect to the (near-)optimal solution determined via solving the mixed-integer programming formulation of the problem with CPLEX 12.1. Recall that CPLEX 12.1 did not solve the problem to optimality for the problem with constant penalty costs and discounted penalty costs. Therefore, we have compared the *lower bound* returned by CPLEX 12.1 after it terminated with the solution returned by the heuristic in benchmarking it. Therefore, the optimality gaps reported for the solutions returned by the heuristic are actually *upper bounds* on the actual optimality gap of these solutions.

Table 1 reports the upper bounds on the optimality gaps for the solutions returned by the specialized heuristic solution method (SHSM) and a modified heuristic solution method (M-SHSM) where, during the scheduling phase, the selected arcs are put in the queue based on their optimal placement in it. In examining the objective functions of the solutions returned by the heuristics, we note that we calculated the performance of the ILN model in each time period over the available network (i.e., which arcs have been completed) rather than using the routing phase of the heuristic to calculate this performance. The reason that this is done is because the ILN model is used as a measure of the physical operations of the network, so that we are concerned with measuring the physical operations of the available network according to the solution returned by the heuristic. In other words, it is often not necessary for the power/telecommuncations infrastructures to make operational decisions over the network - the flow on these systems is determined through their physical properties. Therefore, the ILN model can be used to measure the performance of the solution methods since it is not necessary, for the full restoration and scheduling plan, to determine the actual operational decisions in the network. The results of Table 1 indicate that the heuristic solution methods run effectively in terms of both solution time and quality in order to determine these full restoration and scheduling plans for interdependent infrastructure systems.

	Constant Penalty Costs		Discounted Penalty Costs	
	Error Bound	Time (s)	Error Bound	Time (s)
SHSM	13.3%	30	10.1%	30
M-SHSM	2.7%	68	2.5%	61

Table 1: The performance of the heuristic solution methods for the 101 arc case study.

5 Conclusion

We have developed a mathematical formulation that integrates the restoration planning and scheduling decisions in order to restore essential services provided by interdependent infrastructure systems. Our model has several unique features over previous research on interdependent infrastructure systems: (i) the model fully integrates the restoration planning and scheduling decisions, (ii) the objective function provides a measure of how well the services are being restored throughout the recovery effort, rather than just at the end of it, and (iii) the formulation of the scheduling decisions is motivated by discussions with the managers of the interdependent infrastructure systems about the characteristics of the problem. This model may be limited in its application due to the fact that the managers of these systems may not have access to advanced commercial software packages to solve mixed-integer programming problems. Therefore, we have developed a heuristic solution method based on ideas from network flow problems and scheduling problems. This heuristic solution is quite flexible in terms of its application and fully utilizes the structure of the different components of the problem. We tested our mathematical model and heuristic solution method on a set of realistic data representing the power and telecommunications infrastructures of Manhattan. Our computational results demonstrate that there is significant value in integrating the restoration planning and scheduling decisions as opposed to making them in a decentralized, sequential manner. Further, we demonstrate that our heuristic solution method is capable of providing integrated restoration plans and scheduling decisions of high quality with computational resources similar to those that would be available to the managers of the infrastructure systems in the time frame in which they need to make these decisions.

The work in this paper will set the foundation for several important areas of future research. Solutions of high quality for the integrated restoration planning and scheduling problem can be obtained using commercial software with modest computational resources and also with our customized heuristic for the problem. In the future, it will be interesting to consider other specializations of our general heuristic solution method for these types of problems. Further, the solutions to this problem could be quite useful in evaluating *mitigation* and *pre-positioning* strategies for resource allocation before the disastrous event. For disastrous events that can be forecasted (e.g., hurricanes), we can pre-position resources (for example, generators) in areas where services may be disrupted by the disastrous event in order to mitigate the impact of the event on the services provided by the interdependent infrastructure systems. In order to measure the success of the pre-positioning decisions, we will need to evaluate them by solving the integrated restoration planning and scheduling problem under various scenarios. Therefore, the development of customized approaches for this problem will be important in order to determine the optimal pre-positioning strategies.

References

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network flows: Theory, algorithms, and applications.* Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

[2] E. D. Dolan, R. Fourer, J. J. Moré, and T. S. Munson. Optimization on the NEOS server. *SIAM News*, 35(6):4, 8–9, July/August 2002.

[3] J. Gong, E.E. Lee, J.E. Mitchell, and W.A. Wallace. Logic-based multi-objective optimization for restoration planning. In W. Chaovalitwongse, K.C. Furman, and P.M. Pardalos, editors, *Optimization and Logistics Challenges in the Enterprise*, chapter 11. Springer, 2009.

[4] E.E. Lee. *Assessing vulnerability and managing disruptions to interdependent infrastructure systems: A network flows approach.* PhD thesis, Rensselaer Polytechnic Institute, Troy, NY, 2006.

[5] E.E. Lee, J.E. Mitchell, and W.A. Wallace. 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, 2007.

[6] E.E. Lee, J.E. Mitchell, and W.A. Wallace. Network flow approaches for analyzing and managing disruptions to interdependent infrastructure systems. In John G. Voeller, editor, *Wiley Handbook of Science and Technology for Homeland Security*, volume 2, pages 1419–1428. John Wiley, 2009.

[7] R. Lougee-Heimer. The Common Optimization Interface for Operations Research. *IBM Journal of Research and Development*, 47(1):57–66, 2003.

[8] D. Mendonca and W.A. Wallace. Impacts of the 2001 World Trade Center attack on New York City critical infrastructures. *Journal of Infrastructure Systems*, 12(4):260–270, 2006.

[9] T.D. O'Rourke. Critical infrastructure, interdependencies, and resilience. *The Bridge: National Academy of Engineering*, 37(1):22–29, 2007.

[10] M.L. Pinedo. *Scheduling: Theory, Algorithms, and Systems.* Springer, New York, New York, 2008.

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

[12] M.W.P. Savelsbergh, R.N. Uma, and J. Wein. An experimental study of LP-based approximation algorithms for scheduling problems. *INFORMS Journal on Computing*, 17:123–136, 2005.

[13] J.P. Sousa and L.A. Wolsey. A time indexed formulation of non-preemptive single machine scheduling problems. *Mathematical Programming*, 54(3):353–367, 1992.

[14] W.A. Wallace, D. Mendonca, E. Lee, J.E. Mitchell, and J. Chow. Managing disruptions to critical interdependent infrastructures in the context of the 2001 World Trade Center attack. *Beyond September 11th: An Account of Post-Disaster Research*, pages 165–198, 2003.