Vehicle Routing for the Last Mile of Power System Restoration

Pascal Van Hentenryck
Brown University,
Providence, RI, USA
pvh@cs.brown.edu

Carleton Coffrin
Brown University,
Providence, RI, USA
cjc@cs.brown.edu

Russell Bent
Los Alamos National Laboratory,
Los Alamos NM, USA
rbent@lanl.gov

This paper considers last-mile disaster recovery for power restoration, i.e., how to schedule and route a fleet of repair crews to restore the power network as fast as possible after a disaster. To tackle the computational difficulties raised by this joint repair/restoration problem, the paper proposes a four-stage approach based on the idea of *constraint injection*, which decouples the power-restoration and vehicle-routing optimizations, while still capturing the restoration aspect in the routing component. The approach is shown to produce high-quality solutions and to scale to large disasters based on the United States infrastructure.

Keywords - Last Mile Power Restoration, Vehicle Routing.

1 Background & Motivation

Every year, seasonal hurricanes threaten coastal areas. The severity of hurricane damage varies from year to year, but hurricanes often cause power outages that have considerable impacts on both quality of life (e.g., crippled medical services) and economic welfare. Therefore considerable human and monetary resources are always spent to prepare for, and recover from, threatening disasters. At this time, policy makers work together with power system engineers to make the critical decisions relating to how money and resources are allocated for preparation and recovery of the power system. Unfortunately, due to the complex nature of electrical power networks, these preparation and recovery plans are limited by the expertise and intuition of power engineers. Moreover, current preparation methods often do not use valuable disaster-specific information.

This research considers, for the first time, last-mile disaster recovery for power restoration, i.e., how to schedule and route a fleet of repair crews to restore the power network as fast as possible after a disaster. Our prior research indicated the feasibility of last-mile disaster recovery for the water supply, showing significant benefits over the practice in the field [24]. That work was deployed as part as Los Alamos National Laboratory's operational tools to provide recommendations to the U.S. Department of Homeland Security. However, last-mile power restoration adds another level of complexity, since it introduces a combinatorial optimization aspect to traditional power restoration processes. A direct approach, which jointly optimizes the vehicle schedule and the power restoration process cannot meet the real-time constraints imposed in disaster recovery.

To meet these computational challenges, this paper proposes the idea of *constraint injection*, which decouples the power-restoration and vehicle-routing optimizations, while still capturing the restoration aspect in the routing component. The constraint-injection approach first solves two power restoration subproblems to generate precedence constraints between repairs which are then injected in the vehicle-routing subproblem to produce high-quality joint repair/restoration schedules. The resulting multi-stage approach produces substantial improvements over the practice in field on real-life benchmarks of significant sizes, demonstrating both solution quality and scalability. The rest of the paper is organized as follows. Section 2 positions the problem with respect to prior work in power restoration and Section 3 formalizes the problem. Section 4 presents the multi-stage approach based on constraint injection, while Section 5 reports the experimental results.

2 Prior Work

Power engineers have been studying power system restoration (PSR) for at least 30 years (see [3] for a comprehensive collection of work). The goal of PSR research is to find fast and reliable ways to restore a power system to its normal operational state after a blackout event. PSR research has considered not only *steady-state* behavior, in which the flow of electricity is modeled by physical laws, but also *dynamic* behavior which considers transient states occurring during the process of modifying the power system state (e.g., when energizing components). Indeed, these short, but extreme, states may cause unexpected failures which must also be considered carefully [4]. Moreover, power systems are comprised of many different components (e.g., generators, transformers, and capacitors) which have some flexibility in their operational parameters but may be constrained arbitrarily. For example, generators often have a set of discrete generation levels and transformers have a continuous but narrow range of tap ratios. Restoration algorithms often take these into account.

The PSR research community has recognized that global optimization is often impractical for such complex nonlinear systems and adopted two main solutions strategies. The first strategy is to use domain-expert knowledge (i.e., power engineer intuition) to guide an incomplete search of the solution space. These incomplete search methods include *knowledge-based and expert systems* [20, 15, 5, 6] and *local search* [17, 18]. The second strategy is to approximate the power system with a linear model and to try solving the approximate problem optimally [25, 13, 12]. Some work hybridized both strategies by designing expert systems that solves a series of approximate problems optimally [19, 14]. Observe however that most PSR work assumes that all network components are operational and "only" need to be reactivated (e.g., [4, 5]). The PSR focus is thus to determine the best order of activation and the best reconfiguration of the system components.

This paper focuses on joint repair and restoration problem, i.e., how to dispatch crews to repair the power-system components in order to restore the power system as quickly as possible. There are strong links between traditional PSR research and our disaster-recovery research. In particular, finding a good order of restoration is central in the repair-dispatching problem. However, the joint repair/recovery problem introduces a combinatorial optimization aspect to restoration that fundamentally changes the nature of the underlying optimization problem. The salient difficulty is to combine two highly complex subproblems, vehicle routing and power restoration, whose objectives may conflict. In particular, the routing aspect optimized in isolation may produce a poor restoration schedule, while an optimized power restoration may produce a poor routing and delay the restoration. To the best of our knowledge, joint repair and restoration is the first PSR application that considers repair and reactivation decisions simultaneously.

It is also important to mention that the research described in this paper is part of a larger effort on disaster preparedness and recovery whose goal is to mitigate the impact of disasters on multiple infrastructures. Disaster response typically consists of a planning phase which takes place before the disaster hits and a recovery phase which is initiated after the disaster has occurred. The planning phase often involves a two-stage stochastic or robust optimization problem with explicit scenarios which are generated by sophisticated weather and fragility simulations (e.g., [10, 11, 24]). The recovery phase is generally a deterministic optimization problem which assumes, in a first approximation, that the damages to the various infrastructures are known. For the power infrastructure, the planning phase determines where to stockpile power components under various disaster scenarios [11]. This paper focuses in the recovery phase, i.e., how to repair and restore the power infrastructure as fast as possible, given the stockpiling decisions.

3 Problem Formalization

This section formalizes the Power Restoration Vehicle Routing Problem (PRVRP).

The Routing Component The PRVRP is defined in terms of a graph $G = \langle S, E\rangle$ where S represents sites of interest and E are the travel times between sites. The sites are of four types: (1) the depots H^+ at which repair vehicles depart; (2) the depots H^- at which repair vehicles must return; (3) the depots W^- where stockpiled resources are located; and (4) the locations W^+ where electrical components (e.g., lines, buses, and generators) must be repaired. Due to infrastructure damages, the travel times on the edges are typically not Euclidian, but do form a metric space. For simplicity, this paper assumes that the graph is complete and $t_{i,j}$ denotes the distance between sites i and j.

The restoration has at its disposal a set V of vehicles. Each vehicle $v \in V$ is characterized by its departure depot h_v^+, its returning depot h_v^-, and its capacity c_v. Vehicle v starts from h_v^+, performs a number of repairs, and return to h_v^-. It cannot carry more resources than its capacity.

The restoration must complete a set J of restoration jobs. Each job j is characterized by a pickup location p_j^+, a repair location p_j^-, a volume d_j, a service time s_j, and a network item n_j. Performing a job consists of picking up repair supplies at p_j^+ which uses d_j units of the vehicle's capacity, traveling to site p_j^-, and repairing network item n_j at p_j^- for a duration s_j. After completion of job j, network item n_j is working and can be activated.

A solution to the PRVRP associates a route $\langle h_v^+, w_1, \ldots, w_k, h_v^-\rangle$ with each vehicle $v \in V$ such that all locations are visited exactly once. A solution can then be viewed as assigning to each location $l \in H^+ \cup W^+ \cup W^-$, the vehicle $vehicle(l)$ visiting l, the load $load_l$ of the vehicle when visiting l, the next destination of the vehicle (i.e., the successor σ_l of l in the route of l), and the earliest arrival time EAT_l of the vehicle at location l. The loads at the sites can be defined recursively as follows:

$$\begin{array}{lcll} load_l & = & 0 & \text{if } l \in H^+ \\ load_{\sigma_l} & = & load_l + d_l & \text{if } l \in W^+ \\ load_{\sigma_l} & = & load_l - d_l & \text{if } l \in W^-. \end{array}$$

Pickup locations increase the load, while delivery locations decrease the load. The earliest arrival times can be defined recursively as

$$\begin{array}{lcll} EAT_l & = & 0 & \text{if } l \in H^+ \\ EAT_{\sigma_l} & = & EAT_l + t_{l,\sigma_l} & \text{if } l \in W^+ \\ EAT_{\sigma_l} & = & EAT_l + t_{l,\sigma_l} + s_l & \text{if } l \in W^-. \end{array}$$

The earliest arrival time of a location is the earliest arrival time of its predecessor plus the travel time and the service time for repair locations. The earliest departure time EDT_l

from a location is simply the earliest arrival time to which the service time is added for delivery locations. A solution must satisfy the following constraints:

$$\begin{array}{lcll} vehicle(p_j^+) & = & vehicle(p_j^-) & \forall j \in J \\ EAT_{p_j^+} & < & EAT_{p_j^-} & \forall j \in J \\ load_l & \leq & c_{vehicle(l)} & \forall l \in W^+ \cup W^-. \end{array}$$

The first constraint specifies that the same vehicle performs the pairs of pickups and deliveries, the second constraint ensures that a delivery takes place after its pickup, while the third constraint makes sure that the capacities of the vehicles are never exceeded.

The Power Network $\mathcal{PN} = \langle N, L \rangle$ is defined in terms of a set N of nodes and a set L of lines. The nodes $N = N^b \cup N^g \cup N^l$ are of three types: the buses N^b, the generators N^g, and the loads N^l. Each bus b is characterized by its set N_b^g of generators, its set N_b^l of loads, its set LO_b of exiting lines, and its set LI_b of entering lines. The maximum capacity or load of a node in $N^g \cup N^l$ is denoted by $\hat{P}_i^v$. Each line l is characterized by its susceptance B_l and its transmission capacity $\hat{P}_l^l$. Its from-bus is denoted by L_l^- and it to-bus by L_l^+. The network item n_j of job j is an item from $N \cup L$. The set $\{n_j \mid j \in J\}$ denotes the damaged items D.

The PRVRP Objective is to minimize the total watts/hours of blackout, i.e., $\int unservedLoad(t)\, dt$. Each repair job provides an opportunity to reduce the blackout area (e.g., by bringing a generator up) and the repairs occur at discrete times $T_1 \leq T_2 \leq \ldots \leq T_{|J|}$. Hence the objective can be rewritten into the minimization of $\sum_{i=2}^{|J|} unservedLoad(T_{i-1}) \times (T_i - T_{i-1})$. It remains to describe the meaning of “unserved load” in this formula. At each discrete time T_i, exactly i network elements have been repaired and can be activated, but it may not be beneficial to reactivate all of them. Hence, since we are interested in a best-case power flow analysis, we assume that, after each repair, the optimal set of elements is activated to serve as much of the load as possible. Generation and load can also be dispatched and shed appropriately.

Under these assumptions, computing the unserved load becomes an optimization problem in itself. Model 1 depicts a MIP model for minimizing the unserved load assuming a linearized DC model of power flow. The inputs of the model are the power network (with the notations presented earlier), the set D of damaged nodes, the set R of already repaired nodes, and the value *MaxFlow* denoting the maximum power when all items are repaired. Variable y_i capture the main decision in the model, i.e., whether to reactivate repaired item i. Auxiliary variable z_i determines if item i is operational. The remaining decision variables determine the power flow on the lines, loads, and generators, as well as the phase angles for the buses.

Model 1 A MIP Model for Minimizing Unserved Load.

Inputs:

$\mathcal{PN} = \langle N, L \rangle$	the power network
D	the set of damaged items
R	the set of repaired items
$MaxFlow$	the maximum flow (MW)

Variables:

$y_i \in \{0,1\}$	- item i is activated
$z_i \in \{0,1\}$	- item i is operational
$P_i^l \in (-\hat{P}_i^l, \hat{P}_i^l)$	- power flow on line i (MW)
$P_i^v \in (0, \hat{P}_i^v)$	- power flow on node i (MW)
$\theta_i \in (-\frac{\pi}{6}, \frac{\pi}{6})$	- phase angle on bus i (rad)

Minimize

$$MaxFlow - \sum_{b \in N^b} \sum_{i \in N_b^l} P_i^v \quad (1)$$

Subject to:

$$y_i = 1 \quad \forall i \in (N \cup L) \setminus D \quad (2)$$
$$y_i = 0 \quad \forall i \in D \setminus R \quad (3)$$
$$z_i = y_i \quad \forall i \in N^b \quad (4)$$
$$z_i = y_i \wedge y_j \quad \forall j \in N^b, \forall i \in N_j^g \cup N_j^l \quad (5)$$
$$z_i = y_i \wedge y_{L_i^+} \wedge y_{L_i^-} \quad \forall i \in L \quad (6)$$
$$\sum_{j \in N_i^l} P_j^v = \sum_{j \in N_i^g} P_j^v + \sum_{j \in LI_i} P_j^l - \sum_{j \in LO_i} P_j^l \quad \forall i \in N^b \quad (7)$$
$$0 \leq P_i^v \leq \hat{P}_i^v * z_i \quad \forall j \in N^b, \forall i \in N_j^g \cup N_j^l \quad (8)$$
$$-\hat{P}_i^l * z_i \leq P_i^l \leq \hat{P}_i^l * z_i \quad \forall i \in L \quad (9)$$
$$P_i^l \geq B_i * (\theta_{L_i^+} - \theta_{L_i^-}) + M * (\neg z_i) \quad \forall i \in L \quad (10)$$
$$P_i^l \leq B_i * (\theta_{L_i^+} - \theta_{L_i^-}) - M * (\neg z_i) \quad \forall i \in L \quad (11)$$

The model objective minimizes the unserved load. Constraints (2)–(6) determine which items can be activated and which are operational. Constraints (2) specify that undamaged items are activated and constraints (3) specify that damaged items cannot be activated if they have not been repaired yet. Constraints (4-6) describe which items are operational. An item is operational only if all buses it is connected to are operational. Constraints (4) consider the buses, constraints (5) the loads and generators which are only connected to one bus, and constraints (6) the lines which are connected to two buses. Constraints (7) express Kirchhoff’s law of energy conservation, while constraints (8–11) imposes restrictions on power flow, consumption, and production. Constraints (8) impose lower and upper bounds on the power consumption and production for loads and generators and ensure that a non-operational load or generator cannot consume or produce power. Constraints (9) impose similar bounds on the lines. Finally, constraints (10–11) define the flow on the lines in terms of their susceptances and the phase angles. These constraints are ignored when the line is non-operational through a big M transformation. In practice, M can be set to $B_i * \frac{\pi}{3}$ and the logical connectives can be transformed into linear constraints over 0/1 variables.

Computational Considerations The PRVRP is extremely challenging from a computational standpoint, since it composes two subproblems which are challenging in their own right. On the one hand, pickup and delivery vehicle-routing problems have been studied for a long time in operations research. For reasonable sizes, they are rarely solved to optimality. In particular, when the objective is to minimize the average delivery time (which is closely related to the PRVRP objective), Campbell et al. [9] have shown that MIP approaches have serious scalability issues. The combination of constraint programming and large-neighborhood search has been shown to be very effective in practice and has the advantage of being flexible in accommodating side constraints. On the other hand, computing the unserved load generalizes optimal transmission switching which has also been shown to be challenging for MIP solvers [13]. In addition to line switching, the PRVRP also considers the activation of load and generators. Therefore, it is highly unlikely that a direct approach, combining MIP models for both the routing and power flow subproblems, would scale to the size of even small restorations. Our experimental results with such an approach were in fact very discouraging, which is not surprising given the above considerations. The rest of this paper presents an approach that aims at decoupling both subproblems as much as possible, while still producing high-quality routing schedules.

MULTI-STAGE-PRVRP($Network\ \mathcal{PN}, PRVRP$ G)
$\mathcal{S} \leftarrow MinimumRestorationSetProblem(\mathrm{G}, \mathcal{PN})$
$\mathcal{O} \leftarrow RestorationOrderProblem(\mathcal{PN}, \mathcal{S})$
$\mathcal{R} \leftarrow PrecedenceRoutingProblem(\mathrm{G}, \mathcal{O})$
return $PrecedenceRelaxation(\mathcal{PN}, \mathcal{R})$

Figure 1: The Multi-Stage PRVRP Algorithm.

Model 2 The Minimum Restoration Set Model.

Inputs:

$\mathcal{PN} = \langle N, L \rangle$	the power network
D	the set of damaged items
$MaxFlow$	the maximum flow (MW)

Variables:

$y_i \in \{0,1\}$	- item i is activated
$z_i \in \{0,1\}$	- item i is operational
$P_i^l \in (-\hat{P}_i^l, \hat{P}_i^l)$	- power flow on line i (MW)
$P_i^v \in (0, \hat{P}_i^v)$	- power flow on node i (MW)
$\theta_i \in (-\frac{\pi}{6}, \frac{\pi}{6})$	- phase angle on bus i (rad)

Minimize

$$\sum_{i \in N \cup L} y_i \quad (1)$$

Subject to:

$$\sum_{b \in N^b} \sum_{i \in N_b^l} P_i^v = MaxFlow \quad (2)$$

$$y_i = 1 \quad \forall i \in N \setminus D \quad (3)$$

Constraints (4–11) from Model 1

4 Constraint Injection

As mentioned, a direct integration of the routing and power-flow models, where the power-flow model is called upon to evaluate the quality of (partial) routing solutions, cannot meet the real-time constraints imposed by disaster recovery. For this reason, we explore a multi-stage approach exploiting the idea of *constraint injection*. Constraint injection enables us to decouple the routing and power-flow models, while capturing the restoration aspects in the routing component. It exploits two properties to perform this decoupling. First, once all the power has been restored, the subsequent repairs do not affect the objective and the focus can be on the routing aspects only. Second, and most importantly, a good restoration schedule can be characterized by a partial ordering on the repairs. *As a result, the key insight behind constraint injection is to impose, on the routing subproblem, precedence constraints on the repair crew visits that capture good restoration schedules.*

The injected constraints are obtained through two joint optimization/simulation problems. First, the Minimum Restoration Set Problem computes the smallest set of items needed to restore the grid to full capacity. Then, the Restoration Order Problem determines the optimal (partial) order for restoring the selected subset in order to minimize the total blackout hours. The resulting partial order provides the precedence constraints injected in the pickup and delivery vehicle-routing optimization. Once the routing solution is obtained, injected precedence constraints between vehicles are relaxed, since they may force vehicles to wait unnecessarily. The final algorithm is a multi-stage joint optimization/simulation algorithm depicted in Figure 1. We will now review each of the steps in detail.

The Minimum Restoration Set Problem (MRSP) determines the smallest set of items to restore for ensuring full network capacity. Model 2 depicts the mathematical model using a linear DC model. The optimization is closely related to the model for the unserved load presented in Model 1, but has three significant changes. First, the objective (1) now minimizes the number of repairs. Second, constraint (2) ensures that the network will operate at full capacity. The remaining model constraints are identical to (4–11) in Model 1. However, constraints (3) from Model 1 is excluded since we allow all items to be repaired.

The Restoration Ordering Problem Once a minimal set of items to repair is obtained, the Restoration Ordering Problem (ROP) determines the best order in which to repair the items. The ROP ignores the routing aspects and the duration to move from one location to another, which would couple the routing and power flow aspects. Instead, it views the restoration as a sequence of discrete steps and chooses which item to restore at each step. This subproblem is similar to the network re-energizing problem studied in PSR research but only considers the steady-state behavior, because this is the appropriate level for the PRVRP. Model 3 depicts the ROP model for the linearized DC model. The ROP contains essentially $|R|$ flow models similar to those from Model 1, where R denotes the set of selected items to repair. These flows are linked through the decision variables o_{rk} which specify whether item r is repaired at step k. Constraint (3) makes sure that at most one item is repaired at each step, constraint (4) ensures that an item remains repaired in future time steps, and constraint (5) makes sure that an item is activated only if it has been repaired. Constraint (2) computes the flow at each step and the objective (1) minimizes the sum of the differences between the maximum flow and the flow at each step. Constrains (6-14) are explained in Model 1.

For instances with more than 30 steps, this model can be too difficult to solve for state-of-the-art MIP solvers. Instead, we use a technique called Large Neighborhood Search (LNS) to find near-optimal solution quickly (e.g., [22, 7]). The key idea underlying LNS is to fix parts of a solution in a structured but randomized way and to reoptimize over the remaining decision variables. This process is iterated until the solution has not been improved for a number of iterations. In the case of the ROP, LNS relaxes a particular subsequence, fixing the remaining part of the ordering, and reoptimes the relaxed sequence. The reoptimization can use any optimization technology. Note also that LNS provides an innovative way of integrating optimization and simulation and hence generalizes naturally to non-linear black-box models of power flow.

Vehicle Routing with Precedence Constraints The ROP produces an ordering of the repairs which is used to inject precedence constraints on the jobs. This gives rise to a vehicle routing problem that will implement a high-quality restoration plan while optimizing the dispatching itself. Note that the ROP is not used to impose a – total – ordering; instead it really injects a partial order between the jobs. Indeed, several repairs are often necessary to restore parts of the unserved demand: Imposing a total order between these repairs reduces the flexibility of the routing, thus possibly degrading solution quality. As a result, the ROP solution partitions the set of repairs into a sequence of groups and the precedence constraints are imposed between the groups. The resulting Pickup and Delivery Vehicle Routing Problem with Precedence Constraints (PDVRPPC) consists in assigning a sequence of jobs to each vehicle, satisfying the vehicle capacity and pickup and delivery constraints specified earlier, as well as the precedence constraints injected by the ROP. A precedence constraint $i \rightarrow j$ between job i and j is satisfied if $EDT_i \leq EDT_j$. The objective consists in minimizing the average repair time, i.e., $\sum_{j \in J} EDT_j$. The PDVRPPC is solved using LNS and constraint programming. LNS and constraint programming are very effective for complex vehicle routing problems (e.g., [7, 8]). In contrast, traditional MIP systems have difficulty with the objective function of the PDVRPPC (e.g., [9]).

Model 3 The Restoration Ordering Problem.

Inputs:

$\mathcal{PN} = \langle N, L \rangle$	the power network
D	the set of damaged items
R	the set of items to repair
$MaxFlow$	the maximum flow (MW)

Variables:

$flow_k$	- the flow in step k (MW)
$o_{ik} \in \{0,1\}$	- item i is repaired in step k
$y_{ik} \in \{0,1\}$	- item i is activated in step k
$z_{ik} \in \{0,1\}$	- item i is operational in step k
$P^l_{ik} \in (-\hat{P}^l_i, \hat{P}^l_i)$	- power flow on line i in step k (MW)
$P^v_{ik} \in (0, \hat{P}^v_i)$	- power flow on node i in step k (MW)
$\theta_{ik} \in (-\frac{\pi}{6}, \frac{\pi}{6})$	- phase angle on bus i in step k (rad)

Minimize

$$\sum_{k=1}^{|R|} (MaxFlow - flow_k) \quad (1)$$

Subject to: $(1 \leq k \leq |R|)$

$$flow_k = \sum_{b \in N^b} \sum_{i \in N^l_b} P^v_{ik} \quad (2)$$

$$\sum_{r \in R} o_{rk} = k \quad (3)$$

$$o_{rk-1} \leq o_{rk} \quad \forall r \in R \quad (4)$$

$$y_{ik} \leq o_{ik} \quad \forall i \in D \quad (5)$$

$$y_{ik} = 1 \quad \forall i \in (N \cup L) \setminus D \quad (6)$$

$$y_{ik} = 0 \quad \forall i \in D \setminus R \quad (7)$$

$$z_{ik} = y_{ik} \quad \forall i \in N^b \quad (8)$$

$$z_{ik} = y_{ik} \wedge y_{jk} \quad \forall j \in N^b, \forall i \in N^g_j \cup N^l_j \quad (9)$$

$$z_{ik} = y_{ik} \wedge y_{L^+_i k} \wedge y_{L^-_i k} \quad \forall i \in L \quad (10)$$

$$\sum_{j \in N^l_i} P^v_{jk} = \sum_{j \in N^g_i} P^v_{jk} + \sum_{j \in LI_i} P^l_{jk} - \sum_{j \in LO_i} P^l_{jk} \quad \forall i \in N^b$$

$$0 \leq P^v_{ik} \leq \hat{P}^v_i * z_{ik} \quad \forall j \in N^b, \forall i \in N^g_j \cup N^l_j \quad (11)$$

$$-\hat{P}^l_i * z_{ik} \leq P^l_i \leq \hat{P}^l_i * z_{ik} \quad \forall i \in L \quad (12)$$

$$P^l_{ik} \geq B_i * (\theta_{L^+_i k} - \theta_{L^-_i k}) + M * (\neg z_{ik}) \quad \forall i \in L \quad (13)$$

$$P^l_{ik} \leq B_i * (\theta_{L^+_i k} - \theta_{L^-_i k}) - M * (\neg z_{ik}) \quad \forall i \in L \quad (14)$$

The Precedence Relaxation Problem The last step of the algorithm is a post-processing optimization which relaxes some of the injecting constraints. Indeed, it may happen that vehicles end up waiting at some repair locations due to the precedence constraints. In such circumstances, it is almost always beneficial to relax the precedence constraints and let the vehicle perform its repairs earlier. This step assumes that the routes are fixed and only studies which injected constraints can be relaxed to reduce the overall size of the blackouts. The model can be specified in terms of a linearized DC model very much like the ROP (e.g., by introducing variables specifying the time at which a repair is performed). Once again, the MIP model does not scale to the large-scale disasters and our implementation also uses LNS.

Simulation-Based Optimization This section has specified models for the MRSP and the ROP in terms of the linearized DC model. However, constraint injection naturally applies to a more general setting in which the power-flow model is encapsulated in a black-box simulator. This is important, since the electrical power industry has developed several complementary tools for modeling the behavior of power systems (e.g. T2000, PSLF, Powerworld, PSS). In this setting, the MRSP and the ROP are solved using LNS over the simulator. The functionality of the simulator should be similar to the linearized DC model in Model 1. In other words, given a power network $\mathcal{PN}$, a set D of damaged items, and a set R of repaired items, the simulator returns the minimal unserved load in the network.

5 Experimental Results

Benchmarks The disaster scenarios are based on the US infrastructure and were generated at Los Alamos National Laboratory using state-of-the-art hurricane simulation tools similar to those used by the National Hurricane Center [2]. The benchmarks are based on four different geographic locations. For a given location, the benchmarks share the same power and transportation infrastructure but differ in the damage scenarios. Each scenario is characterized by its damage to the power system and transportation infrastructures and is generated by the disaster simulation tools (e.g., weather and fragility simulations). This produces a total of 51 different benchmarks. Each geographic location has a power network containing about 300 items and there are about 13 repair crews available for restoration. The number of damaged items ranges from 0 to 121. For simplicity, we group the benchmarks in three categories small ($|J| < 20$), medium ($|J| < 50$), and large ($|J| \geq 50$). In total, there are 28 small, 14 medium, and 9 large benchmarks. The large benchmarks are considerably more difficult than prior work in related areas. For example, the standard IEEE-118 benchmark has not been solved optimally in the context of optimal line switching [13] or network interdiction [21] (our MIP results are consistent for this difficulty level).

The Baseline & Relaxed Algorithm To validate our results, we compare our PRVRP algorithm to a baseline algorithm modeling the practice in field which proceeds roughly as follows: (1) Power engineers use their knowledge of the network to decide which key items to repair; (2) Crews are dispatched to make the necessary repairs; (3) Crews prefer to fix all broken items near the area they are dispatched. This process can be captured as an instance of our constraint-injection algorithm which the following choices: (1) the MRSP and ROP are solved with a greedy heuristic that incrementally chooses to repair the item bringing the largest increase in power flow; (2) The routing problem is identical to the PDVRPPC, except that the objective function seeks to minimize the total travel distance, not the sum of earliest delivery times. This captures the fact that each vehicle crew works independently to perform their repairs as fast as possible. Additionally, we calculate a relaxation of the PRVRP that assumes an infinite number of repair crews. In this relaxation, every restoration only requires the time for the pickup, delivery, and repair. This relaxation provides an upper bound on the distance between our solution and the optimal solution.

The optimization algorithms were implemented in the COMET system [1, 16, 23] and the experiments were run on Intel Xeon CPU 2.80GHz machines running 64-bit Linux Debian. The experiments use the standard linearized DC power flow equations for the power simulator either as a black-box simulator or directly inside the MIP models presented earlier. Both a MIP and LNS based formulations of the PRVRP problem are considered. The LNS approach is necessary for scaling to large instances and it also demonstrates the feasibility of a black-box simulation approach. Due to fast-response requirements in disaster recovery, each subproblem is solved with a fixed time limit, so that a solution can be found in less than one hour. The time limits are as follows: 2 minutes for MSRP, 20 minutes for ROP, and 30 minutes for PDVRPPC. All of the algorithms require an LNS component to solve the routing aspect of the problem and hence the solution may vary between runs. As a result, we report the mean value of 5 runs of each algorithm. Note also that, on 4 medium and 4 large benchmarks, the MIP solver cannot find a feasible solution to the ROP within the time limit.

Quality of the Results Figures 2 and 3 present the final restoration results for one run of the algorithms on a medium and large instance respectively.[1] The MIP model is omitted from Figure 2 because a feasible solution to the MSRP was not found within the time limit. The results show that the constraint-injection algorithm produces dramatic improvements compared to the practice in the field. Moreover, the results are often close to the infinite-vehicle relaxation, indicating that our algorithm finds near-optimal solutions. Space constraints prevent us from presenting similar results for all the benchmarks. Instead, we present an aggregation of these results for each benchmark size. The reported values are summed across all instances within a benchmark category and then scaled relative to the baseline algorithm. Table 1 presents the results for the PRVRP objective. The first three rows include the benchmarks that can be solved by all algorithms, while the last three rows include the benchmarks that could not be solved by the MIP-based constraint injection. The constraint-injection approaches consistently reduces the blackout area by 50%

[1] The average speed limit of the damaged road network (50mph) is used to convert the distance traveled by each repair vehicle into units of time.

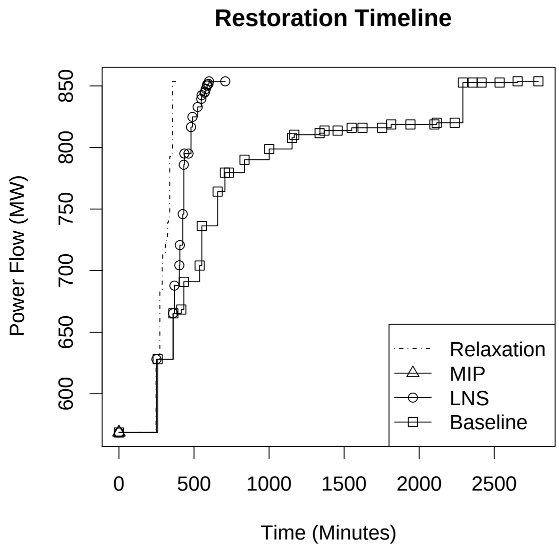

Figure 2: PRVRP Results Comparison (41 damaged items)

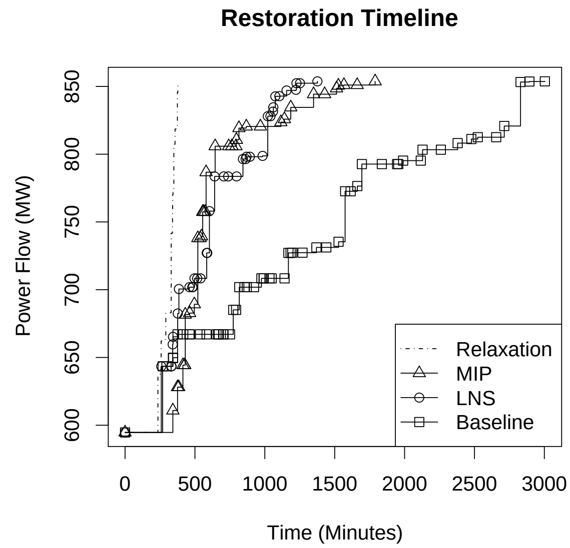

Figure 3: PRVRP Results Comparison (67 damaged items)

Restoration Objective						
Size (Count)		Baseline	MIP	LNS	Relaxation
Small	28	100%	46.1%	46.2%	34.6%
Medium	10	100%	31.3%	30.6%	21.4%
Large	5	100%	40.9%	46.8%	21.0%
Small	0	–	–	–	–
Medium	4	100%	–	47.3%	33.0%
Large	4	100%	–	69.7%	24.3%

Table 1: PRVRP Routing Quality

Size (Count)		Baseline	MIP	LNS
Average Restoration Set Size				
Small	28	7.64	6.79	7.04
Medium	10	25.3	23.2	23.9
Large	5	49	44.8	45.4
Restoration Order Quality				
Small	28	100%	59.3%	58.1%
Medium	10	100%	38.5%	38.7%
Large	5	100%	41.6%	52.3%

Table 2: PRVRP Subproblem Quality

or more. Finally, the quality of the LNS-based constraint injection reduces to a 30% improvement on the largest instances (i.e., the 4 that are unsolvable by the MIP-based constraint injection). This is primarily caused by one specific benchmark whose structure we are investigating. Finally, because these instances are often two to three times larger than the medium-sized instances, additional time is required for the PDVRPPC stage of the algorithm. Our future work will investigate how to boost the performance of the PDVRPPC algorithm for very large instances.

Table 2 presents the quality results for the MRSP and ROP subproblems. They indicate the LNS-based and MIP-based algorithms produce 10% improvements for the MRSP and between 40% and 60% improvements on the ROP. Moreover, the results indicate that using an LNS algorithm over a black-box simulator does not degrade the quality of the MRSP and ROP solutions significantly.

6 Conclusion

This paper studied the Power Restoration Vehicle Routing Problem, a novel problem in power system restoration whose goal is to decide how coordinate repair crews effectively in order to recover from blackouts as fast as possible after a disaster. PRVRPs are complex as they combine vehicle routing and power restoration scheduling problems. The paper proposed a multi-stage optimization algorithm based on the idea of constraint injection that meets the aggressive runtime constraints necessary for disaster recovery. The algorithms were validated on real-life benchmarks using the infrastructure of the United States and state-of-the-art hurricane simulation tools. Experimental results show that the constraint-injection algorithms can reduce the blackouts by 50% or more over the practice in the field. Moreover, the results show that the constraint-injection algorithm using large neighborhood search over a blackbox simulator provide competitive quality and scales better than using a MIP solver on the subproblems.

References

[1] Comet 2.1 user manual, dynadec inc. http://dynadec.com/, 2009.

[2] Fema hazus overview. http://www.fema.gov/plan/prevent/hazus/, 2010.

[3] M. Adibi. *Power System Restoration(Methodologies & Implementation Strategies)*. 2000.

[4] M.M. Adibi and L.H. Fink. Power system restoration planning. *Power Systems, IEEE Transactions on*, 9(1):22 –28, feb. 1994.

[5] M.M. Adibi, L.R.J. Kafka, and D.P. Milanicz. Expert system requirements for power system restoration. *Power Systems, IEEE Transactions on*, 9(3):1592 –1600, aug. 1994.

[6] J.J. Ancona. A framework for power system restoration following a major power failure. *Power Systems, IEEE Transactions on*, 10(3):1480 –1485, aug. 1995.

[7] R. Bent and P. Van Hentenryck. A Two-Stage Hybrid Local Search for the Vehicle Routing Problem with Time Windows. *Transportation Science*, 8(4):515–530, 2004.

[8] R. Bent and P. Van Hentenryck. A Two-Stage Hybrid Algorithm for Pickup and Delivery Vehicle Routing Problems with Time Windows. *Computers and Operations Research (Special Issue on Applications in Combinatorial Optimization)*, pages 875–893, 2006.

[9] Ann Melissa Campbell, Dieter Vandenbussche, and William Hermann. Routing for relief efforts. *Transportation Science*, 42(2):127–145, 2008.

[10] Russell Bent Carleton Coffrin, Pascal Van Hentenryck. Strategic planning for power system restoration. *Proceedings of the International Conference on Vulnerability and Risk Analysis and Management (ICVRAM 2011), Hyattsville, MD*, april 2011.

[11] Russell Bent Carleton Coffrin, Pascal Van Hentenryck. Strategic stockpiling of power system supplies for disaster recovery. *Proceedings of the 2011 IEEE Power & Energy Society General Meetings (PES 2011), Detroit, MI*, july 2011.

[12] A. Delgadillo, J.M. Arroyo, and N. Alguacil. Analysis of electric grid interdiction with line switching. *Power Systems, IEEE Transactions on*, 25(2):633 –641, may. 2010.

[13] E.B. Fisher, R.P. O'Neill, and M.C. Ferris. Optimal transmission switching. *Power Systems, IEEE Transactions on*, 23(3):1346 –1355, aug. 2008.

[14] J.A. Huang, L. Audette, and S. Harrison. A systematic method for power system restoration planning. *Power Systems, IEEE Transactions on*, 10(2):869 –875, may. 1995.

[15] J.A. Huang, F.D. Galiana, and G.T. Vuong. Power system restoration incorporating interactive graphics and optimization. pages 216 –222, may. 1991.

[16] L. Michel, A. See, and P. Van Hentenryck. Transparent Parallelization of Constraint Programming. *INFORMS Journal on Computing*, 21(3):363–382, 2009.

[17] A.L. Morelato and A.J. Monticelli. Heuristic search approach to distribution system restoration. *Power Delivery, IEEE Transactions on*, 4(4):2235 –2241, oct. 1989.

[18] H. Mori and Y. Ogita. A parallel tabu search based approach to optimal network reconfigurations for service restoration in distribution systems. volume 2, pages 814 – 819 vol.2, 2002.

[19] T. Nagata, H. Sasaki, and R. Yokoyama. Power system restoration by joint usage of expert system and mathematical programming approach. *Power Systems, IEEE Transactions on*, 10(3):1473 –1479, aug. 1995.

[20] T. Sakaguchi and K. Matsumoto. Development of a knowledge based system for power system restoration. *Power Apparatus and Systems, IEEE Transactions on*, PAS-102(2):320 –329, feb. 1983.

[21] J. Salmeron, K. Wood, and R. Baldick. Analysis of electric grid security under terrorist threat. *Power Systems, IEEE Transactions on*, 19(2):905 – 912, May 2004.

[22] P. Shaw. Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. In *Proceedings of Fourth International Conference on the Principles and Practice of Constraint Programming (CP'98)*, pages 417–431. Springer Verlag, October 1998.

[23] P. Van Hentenryck. *Constraint-Based Local Search*. The MIT Press, Cambridge, Mass., 2005.

[24] P. Van Hentenryck, R. Bent, and C. Coffrin. Strategic planning for disaster recovery with stochastic last mile distribution. In Andrea Lodi, Michela Milano, and Paolo Toth, editors, *CPAIOR*, volume 6140 of *Lecture Notes in Computer Science*, pages 318–333. Springer, 2010.

[25] M.H. Yolcu, Z. Zabar, L. Birenbaum, and S.A. Granek. Adaptation of the simplex algorithm to modeling of cold load pickup of a large secondary network distribution system. *Power Apparatus and Systems, IEEE Transactions on*, PAS-102(7):2064 –2068, jul. 1983.