Joint Assessment and Restoration of Power Systems

Nabeel Gillani

May 1, 2012

Brown University Department of Computer Science

Advisors: Professor Pascal Van Hentenryck, Carleton Coffrin (Ph.D. Candidate)

Reader: Professor Claire Mathieu

Contents

Acknowledgements

As is often the case, there are too many people to thank for the immense personal growth that I have experienced throughout the course of this project, but I want to give it my best shot. First, thank you to God, my mother Rozina, father Nadirshah, and brother Riaz for encouraging me and helping me build the confidence that is so important when doing challenging work. They've been endless sources of support and wisdom. Thank you to my awesome advisors, Professor Pascal Van Hentenryck and Carleton Coffrin, for letting me join their team and cultivating me as a young Jedi in the world of optimization. They are not only my colleagues and teachers, but phenomenal mentors and friends. Thanks to my friends Anita, Alyna, Jonah, Josh, Amit, Tara, and Stephen for reminding me of the importance of passion and humility when learning new concepts, and my childhood friends Dean, Naushad, Tahsin, and Cyrus for showing me what tenacity is. Thanks to my friend Rakesh for editing this work and making it look like I know how grammar works. Finally, thanks to the students and faculty of the Brown University Department of Computer Science and the Brown University Division of Applied Mathematics, as well as countless other professors and students from different departments, who have helped me realize how beautifully integrated math, computer science, and positively impacting the world really are.

Abstract

This project studies how to jointly assess the extent of damage inflicted upon the electrical grid by a hurricane and produce a vehicle routing plan to guide restoration. It builds upon previous work that assumed the extent of the damages done to the electrical grid was known precisely prior to restoration. It lifts this assumption by offering three different approaches for joint assessment and restoration: (1) An online stochastic combinatorial optimization (OSCO) approach using repair crews; (2) A stochastic two-stage approach that uses dedicated exploration crews to determine the true extent of the disaster's impact on the electrical network and then executes an offline routing with repair crews; (3) A hybrid online stochastic combinatorial optimization and exploration (EXPL-OSCO) approach that employs elements of the first two methods. Experimental results generated using information from weather and fragility simulations based on U.S. infrastructure indicate that EXPL-OSCO produces high quality routing solutions that most effectively reduce the size of the blackout in the disaster's aftermath.

"Not everything that counts can be counted, and not everything that can be counted counts."
- Albert Einstein

1 Introduction

The proliferation of networked digital devices throughout the world is making it easier to provide support to individuals and families in the face of natural or man-made disasters. However, there is still a lack of robust tools to inform the logistical planning for, and response to, these devastating situations. Recent disasters such as Hurricane Katrina and the earthquakes in Haiti and Japan have demonstrated the fragility of infrastructure systems and the pressing need for decision support for policymakers looking to minimize human suffering.

Disaster response is a part of the broader field of Humanitarian Logistics, which focuses on using techniques in operations research to manage processes that often center on efficient human and capital resource allocation under tight time constraints. One of the major reasons for disaster response challenges has been the lack of monetary and human resource support for humanitarian logistics planning. With numerous stakeholders and a multi-faceted bottom line, problems involving humanitarian logistics have failed to generate the same track record of positive iteration and improvement when compared to analogously challenging operations research problems in the private sector. This economic problem is coupled with the inherent complexity of disasters emanating from broad geographic affliction, difficulty to anticipate the evolution of human impact, and uncertain interactivity of different factors that may exacerbate their toll on societies [13].

Global disaster trends further attest to this anticipative difficulty. While the average number of mortalities emanating from disasters fell from approximately 140,000 pear year in the 1980s to just over 50,000 in 2003, the death of over 230,000 people as a result of the 2004 Indian Ocean Tsunami illustrates the inherent uncertainty of a given disaster's impact [2]. Realizing shortcomings in response, international leaders have started to call for greater efforts in risk minimization to empower communities to efficiently respond to disasters when they occur instead of simply waiting for copious, and slow-moving, international planning and resources in disaster aftermath [2]. Indeed at their core, disasters are unpredictable, nondeterministically evolving, and costly from both a monetary and human suffering standpoint. This makes designing methods for anticipating and responding effectively to them all the more challenging–but vital for a more parallelized, locally-empowering way of ensuring the robustness and progress of societies everywhere.

1.1 Hurricane Response as an Opportunity

While disasters appear to be virtually impossible to plan for and respond to in practice, thinking strategically about the different parts of a disaster both prior to, during, and after impact can help minimize human and capital losses. One particularly important realization of disaster response planning is repairing the damaged electrical network after a hurricane strikes. An operational electrical network is critical to the administration of a number of other vital services during disaster response. As [9] discusses, there is an interdependence between different networks that is often exposed as a result of cascading failures following disasters, e.g., failures in the information or communication networks that result from dependence on an electrical grid that has been damaged. This often prevents a number of important, potentially life-saving activities such as delivering medical services to the injured or enabling communication between relief crews. Given how fundamental the electrical grid is, it is only natural to investigate how we can quickly and efficiently fix it in the face of disasters (hurricanes) that seasonally plague both the Caribbean and coastal U.S.

The inherent stochasticity underlying the extent of a hurricane's damage on the electrical network calls for a collaborative effort by policymakers and power systems engineers to determine how to repair the grid in a timely manner. Restoration plans, however, are often motivated primarily by the intuition and expertise of these engineers instead of disaster-specific information such as damage profiles produced by the National Hurricane Center.

With an idea of how important it is to fix electrical damages after a hurricane hits, this work reconsiders the last-mile disaster recovery for power restoration. This entails producing a routing schedule for repair crews to restore the electrical network as quickly as possible (i.e., minimizing the size of the power blackout) after a disaster strikes. Since power network failure response produces a large combinatorial space of decisions required by policymakers and repair crews in regards to both power restoration and vehicle routing, earlier models that looked to jointly optimize these subproblems failed to yield high-quality solutions that could be used in practice. Instead, this work builds on the decomposition approach, first proposed in [11], that was shown to significantly reduce the size of a blackout overtime in field applications. This decomposition approach has been used in Los Alamos National Labratory (LANL) tools and activated to advise the U.S. federal government, each time a hurricane of category 3 or above threatens to hit the U.S.

The main contribution of this work is to lift one of the simplifications found in [11]: that the precise damages inflicted upon the electrical network by a hurricane are known a priori, i.e., before the power restoration and routing problems are solved. Instead, it considers a joint infrastructure assessment and repair problem (JIARP) that aims to both identify the extent of the damage and produce high-quality solutions to restore the electrical network. As inputs, the JIARP receives a set of possible damages scenarios (i.e., hurricanes) produced by weather and fragility simulations at LANL. Motivated by field practices, the JIARP considers two cases: the situation where repair crews are also responsible for damage assessment, and the situation where policymakers have dedicated exploration crews that use faster transportation (e.g., helicopters) to survey damages, but cannot themselves make repairs.

To address the JIARP, we present three algorithms: (1) an online stochastic combinatorial optimization (OSCO) algorithm which uses only repair crews to both assess and fix damages, making routing decisions dynamically as uncertainty about the damage set is revealed through site visits; (2) a 2-stage approach that first uses exploration crews to assess damages before using the offline optimization approach to determine vehicle routes for restoration; and (3) a hybrid OSCO approach that deploys both crews simultaneously, conducting both exploration and restoration at the same time. As discussed in detail in the Experimental Results section, hybrid OSCO consistently outperforms both OSCO and two-stage approaches, particularly for different damage amounts, since it only increases the blackout size by reasonably small percentages over the clairvoyant solution.

The remainder of this introduction discusses some of the theory and connections to other fields underlying the different approaches for joint assessment and repair of the electrical network. The rest of this work details the power restoration problem and optimization algorithms as presented in [11]; specifies the joint assessment and repair problem with a detailed description of the proposed algorithms (i.e., OSCO, two-stage, and hybrid OSCO); illustrates, through experimental results, the effectiveness of each in reducing the size of the blackout in comparison to the clairvoyant routing where all damages are known a priori; and concludes the discussion with analysis and other approaches to be considered in the future.

1.2 Theory and Approaches for Response Under Uncertainty

This work relates from both a conceptual and modeling standpoint to several topics in optimization and machine learning. At their core, the algorithms proposed for the JIARP all involve systematically revealing the uncertainty of hurricane damages in order to fully restore the electrical grid. This idea–uncovering some unknown information through iterative inquiry and decision making–is a core tenet of OSCO and simple decision tree classification in machine learning.

1.2.1 Online Stochastic Combinatorial Optimization

Online Stochastic Combinatorial Optimization is an adaptive optimization approach that makes decisions online, influenced by some uncertainty underlying this decision making. OSCO combines methods from online algorithms, stochastic programming, and combinatorial optimization to produce a dynamic decision-making strategy under some exogeneous uncertainty [6]. Here, the term “exogeneous” simply means that the uncertainty in the problem stems from a source beyond the algorithm’s past decisions. This is contrasted with Markov Decision Processes (MDPs), which have endogeneous uncertainty, i.e., uncertainty at a given state that emanates from decisions made at previous states.

OSCO’s hybridization of different methods in computer science and statistics yields an approach that is robust. Much previous work in the space of online algorithms has ignored input distributions that could significantly improve the online decision making for many problems [6]. Stochastic programming itself considers stochasticity in input “data” instead of uncertainty in decision variables, such as which customer to visit next for vehicle routing problems.[1] In the context of the JIARP, it is the latter type of uncertainty that is most prevalent.

For disaster response in the JIARP, the objective is to route vehicles to restore the electrical grid under uncertain damage information, which can be extended to a broader uncertainty about the true path of a given hurricane. The OSCO and hybrid OSCO approaches essentially entail routing vehicles according to a distinguished plan (selected via a consensus decision over the offline routing plans generated for possible scenarios). When a vehicle reaches a damage site, it produces a new (offline) routing plan for all scenarios based on the assessment of damages at that location and the locations that have been visited so far. The solution is a composite plan built on demand as uncertainty about damages is revealed. The details of these methods are described in the sections below. Given the need for dynamic, flexible decision making in dealing with damage uncertainty, OSCO is a good modeling choice for problems arising from the joint assessment and repair of power systems and deserves further exploration.

1.2.2 Decision Trees and Entropy

While OSCO serves as an appropriate formulation for one class of solutions for the JIARP, the two-stage approach utilizes the concept of decision trees from machine learning to guide the damage discovery process. Decision trees are used in statistics and machine learning to classify input data according to certain characteristics. Each node in the tree can represent some characteristic (i.e., the random variable) that will guide classification–e.g., size of screen, weight, cost, etc. if building a classifier for different consumer technologies. Each branch represents a value (or range of values) for a particular characteristic [10].

[1] Although there is also a probability distribution over the “data”, i.e., the hazard scenarios that comprise the pool of possible damages. The key here is that as the OSCO algorithm executes, it gains more “clarity” of the uncertainty, i.e. the real damages, which manifests itself as the potentially damaged sites the routing algorithm prescribes for repair crews to visit.

For the purposes of the damage discovery stage in the JIARP, building a decision tree serves as a reasonable heuristic for initializing the dispatch of exploration crews to different damage sites. The objective of exploration is to visit damage locations in order to eventually classify the uncertain scenario as one that belongs to the predetermined set of possible scenarios. Here, the nodes represent different sites that may or may not be damaged in reality, and the branches represent which disaster scenarios contain (or do not contain) damages at that site. Hence, the result is a binary decision tree. The leaf nodes each correspond to a single damage scenario, where the depth of the leaf node represents the number of sites that may need to be visited in order to completely isolate the empirical damage set as the real damage scenario. The process for constructing the decision tree is described in more detail in the sections below.

The concept of (Shannon's) entropy [12] naturally arises when studying this classification problem in the context of the two-stage approach for the JIARP. For the case of source coding in information theory (a common application area for this work), Shannon defines entropy H for some random variable X belonging to $\{x_i, ..., x_n\}$ as

$$H(X) = -\sum_{i=1}^{n} p(x_i) log_2 p(x_i)$$

Here, x_i refer to the possible states of X, and $p(x_i)$ the probability of achieving this state. For the problem of source coding, Shannon proves that the expected number of bits needed to encode a particular message cannot be less than the entropy as defined above.[2] This expected code length increases if the code is designed for source q (which, like p, is a probability distribution over the states that X can assume) when the true source is p.

While there is no formal or rigorous discussion of entropy as it relates to the JIARP, it is interesting to think of the exploration problem in this context. Just like the expected code length is bounded below by the entropy, so too is the expected "exploration length" (i.e., the number of sites visited) by some analogous measure of entropy before the empirical damage set is appropriate classified. Moreover, just as designing an encoding scheme for an incorrect source exacerbates the penalty for encoding, an exploration scheme that operates under the false assumption that scenario s instead of the true scenario t describes the real set of damages leads to extra site visits that increase the overall size of the blackout. This knowledge, coupled with the decision tree theory provided above, suggests that damage scenarios with low "entropy" (i.e., the most distinguishable, particularly early on in the exploration, from others) will produce explorations that take the shortest amount of time. It also justifies the importance of devising an exploration scheme that helps us classify as quickly as possible the empirical damages resulting from a hurricane's actual path.

[2] Entropy, here, can be thought of as the "number of questions" that must be asked to uniquely identify the original message for the which the code has been produced. It makes intuitive sense, then, that the expected code length cannot be lower than this.

2 The Power Restoration Problem

The Power Restoration Problem is composed of two subproblems: power flow and vehicle routing. This section formalizes the Power Restoration Vehicle Routing Problem (PRVRP) and generalizes the problem described in [11].

2.1 The Vehicle Routing Problem

The PRVRP is defined in terms of a graph $G = (S, E)$, where S refers to sites of interest and E is the travel time between two given sites. There are 4 different types of sites: H^+ represents where the departure locations of the vehicles; H^- is where they must return; W^+ are the depots where resource are stockpiled; and W^- are the locations where electrical components (e.g. lines, buses, and generators) must be repaired. Each location $l \in W^+$ has a repair supply c_l.[3] Due to infrastructure, the travel times on the edges are typically not Euclidean, but do form a metric space. For simplicity, this paper assumes that the graph is complete and $t_{i,j}$ is the travel time between $i, j \in S$.

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 returns 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 repair location p_j^-, a volume d_j, a service time s_j, a repair supply c_j, and a network item n_j. Performing a job consists of picking up the repair supply c_j at some location p_j^+ taking d_j units of the vehicle's capacity, traveling to site p_j^- and repairing network item n_j at p_j^-, which takes some time s_j to complete. After completion of job j, network item n_j is working and can be activated.

A solution to the PRVRP associates a route $(h_v^+, w_1, ..., w_k, h_v^-)$ with each vehicle v such that all repair locations are visited exactly once and all pickup locations at most once (note that there can be many "pickup locations" at a physical depot where resources have been stockpiled). A solution, then, assigns to each visited location $l \in H^+ \cup W^+ \cup W^- \cup H^-$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:

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

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

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

[3] As a relaxation, this work assumes that there is homogeneity in the damaged components, i.e., that any damaged item can be fixed with any item picked up at W^+. The power netowork, however, is still defined in terms of unique electrical components (e.g., lines, buses, loads, and generators). As is discussed in the conclusion, the homogeneous component routing model is a reasonable approximation given the nature of damages following a hurricane.

The earliest arrival time of a location σ_l is equal to the earliest arrival time of its predecessor, l plus the travel time (t_l, σ_l) and the service time (s_l) for delivery 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:

$$\begin{array}{lll} load_l & > & 0 & \forall l \in W^- \\ load_l & \leq & c_{vehicle(l)} & \forall l \in W^+ \cup W^-. \end{array}$$

The first constraint specifies that the vehicle must have repair supplies before performing a pickup, and the second constraint ensures that the capacities of the vehicles are never exceeded.

2.2 The Power Network

The Power Network is represented as $PN = (N, L)$, where N is the set of nodes and L the set of lines comprising the underlying graph. The nodes $N = N^b \cup N^g \cup N^l$ are of three types: the buses N^b, generators N^g, and loads N^l. Each bus b is characterized by its set of generators, denoted N^g_b of generators, its set of N^l_b loads, and its exiting (LO_b) and entering (LI_b) lines. The maximum capacity or load of a node in $N^g \cup N^l$ is denoted by $\hat{P}^v_i$. Each line j is characterized by its susceptance B_j, as well as its transmission capacity $\hat{P}^l_j$. The line j's "from" bus is denoted as $L^-_j$and its "to" bus as L^+_j. The network item n_k of job k is an item belonging to the set $N \cup L$. The set $\{n_k \mid k \in J\}$ represents the damage set of a particular disaster scenario, D.

This work uses the widely accepted linear approximation to the DC power flow model[7],[8],[5]to determine the steady-state power flow. A classic linearized DC model assumes fixed values for both generation and load. It then aims to determine a phase angle θ_i for each bus. The flow along every line $j \in L$ is then given by $B_j(\theta_{L^+_j} - \theta_{L^-_j})$ and Kirchoff's Current Law ensures that flow is conserved throughout the network. We can solve this model through a system of linear equations. In the context of a disaster, however, the appropriate amount of generation and load is unknown, and so, must be determined. Hence, we employ linear programming to solve the linearized DC model with the added decision variables of load and generation values.

2.3 The Routing Objective

The objective of the optimization 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 $T_1 \leq T_2 \leq ... \leq T_{|J|}$. The objective, then, can be rewritting as minimizing the discretized quantity

$$\sum_{i=2}^{|J|} unservedLoad(T_{i-1}) \times (T_i - T_{i-1})$$

Here, *unservedLoad* can be understood as follows: At each discrete time T_i, exactly i network items have been repaired and can be activated, but it may not be beneficial to reactivate all of them.

Fig. 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
$\mathit{MaxFlow}$	the maximum flow (MW)

Variables:

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

Minimize

$$\mathit{MaxFlow} - \sum_{b \in N^b} \sum_{i \in N^l_b} P^v_i \tag{1}$$

Subject to:

$$y_i = 1 \quad \forall i \in (N \cup L) \setminus D \tag{2}$$
$$y_i = 0 \quad \forall i \in D \setminus R \tag{3}$$
$$z_i = y_i \quad \forall i \in N^b \tag{4}$$
$$z_i = y_i \wedge y_j \quad \forall j \in N^b, \forall i \in N^g_j \cup N^l_j \tag{5}$$
$$z_i = y_i \wedge y_{L^+_i} \wedge y_{L^-_i} \quad \forall i \in L \tag{6}$$
$$\sum_{j \in N^l_i} P^v_j = \sum_{j \in N^g_i} P^v_j + \sum_{j \in LI_i} P^l_j - \sum_{j \in LO_i} P^l_j \quad \forall i \in N^b \tag{7}$$
$$0 \leq P^v_i \leq \hat{P}^v_i z_i \quad \forall j \in N^b, \forall i \in N^g_j \cup N^l_j \tag{8}$$
$$-\hat{P}^l_i z_i \leq P^l_i \leq \hat{P}^l_i z_i \quad \forall i \in L \tag{9}$$
$$P^l_i \geq B_i(\theta_{L^+_i} - \theta_{L^-_i}) + M(\neg z_i) \quad \forall i \in L \tag{10}$$
$$P^l_i \leq B_i(\theta_{L^+_i} - \theta_{L^-_i}) - M(\neg z_i) \quad \forall i \in L \tag{11}$$

Figure 1 depicts a MIP model for minimizing the unserved load assuming a linearized DC power flow model. The inputs of the model are the power network (with notations as defined in the previous section), the set of D damaged items, the set of R repaired nodes at any given time, and the value $MaxFlow$ denoting the maximum power when all items are repaired. The activation variables y_i capture the $0/1$ values of the main decision in the model, i.e., whether or not to reactivate item i. The variables z_i denotes if network item i is or is not operational. The remaining decision variables give us the power flow on the lines, loads, and generators, as well as the phase angles for the buses. The objective function minimizes the quantity $unservedLoad(t)$. Constraints $(2)-(6)$ determine which items can be activated and which are operational. Contraints (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 to which it is connected are also 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 Kirchoff's law of energy conservation, while constraints $(8)-(11)$ impose 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, as well as the phase angles of corresponding buses. 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.

2.4 Optimization under Computational Complexity

The PRVRP is extremely computationally challenging because it composes two subproblems (power restoration and vehicle routing), which are challenging in their own right. On 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. [3] have shown that MIP models have serious scalability issues.

The combination of constraint programming and large neighborhood search (LNS) have shown to be very effective in practice, and have the advantage of being flexible in accomodating side constraints and a variety of objective functions. On the other hand, computing the unserved load generalizes optimal transmission switching, which has also proven to be challenging for MIP models [5]. In addition to line switching, the PRVRP also considers the activation of loads and generators. Therefore, it is highly unlikely that taking a direct approach, i.e., combining two challenging MIP problems, will produce a scalable or efficient solution for even small restoration problems. Reference [11] presents an approach for decoupling both subproblems, while still producing high-quality routing schedules.

Decomposing the problem appropriately allows for the possibility of meeting the real-time constraints imposed by disaster recovery. The complexity of this task makes direct integration of both the routing and power flow models an unfavorable solution. For this reason, reference [11] explores a multi-stage approach utilizing the idea of *constraint injection*. Constraint injection decouples the routing and power flow models, while capturing restoration aspects in the routing component. It takes advantage of two properties to obtain the decoupling. First, once all of the power has been restored, the subsequent repairs do not affect the objective and we are able to focus on the routing exclusively. Second, and most importantly, a good restoration schedule can be characterized by a partial ordering on the repairs. *As a result, a key insight behind constraint*

Algorithm 1 The Multi-Stage PRVRP Algorithm

$\texttt{Multi}-\texttt{Stage}-\texttt{PRVRP}(Network\,PN,\; PRVRP\,G)$
1 $S \leftarrow MinimumRestorationSetProblem(G,\, PN)$
2 $O \leftarrow RestorationOrderingProblem(PN,\, S)$
3 **return** $PrecedenceRoutingProblem(G,\, O)$

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 (MRSP) computes the smallest set of items needed to restore the grid to full capacity. Then, the Restoration Ordering Problem (ROP) determines the order for restoring the selected subset to minimize total blackout hours. This order provides the precedence constraints injected in the pickup and delivery vehicle-routing optimization. The final algorithm is a multi-stage optimization depicted in Algorithm 1.

The ROP produces an ordering for the repairs, which is then used to inject precedence constraints on the delivery jobs. This gives rise to a vehicle routing problem that will implement a high-quality restoration plan while simultaneously optimizing the vehicle dispatching. Note that the ROP injects a partial order over the jobs. Indeed, several repairs are often necessary to restore parts of the unserved demand. 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. Imposing a global ordering between all of the repair crews reduces the flexibility of the routing, thereby degrading solution quality. Hence, the ordering constraints are imposed only per-vehicle. The resulting pickup and delivery vehicle routing problem with precedence constraints consists of assigning a sequence of jobs to each vehicle and 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 $EDT_i \leq EDT_j$ or if jobs i and j are scheduled on different vehicles, i.e., $vehicle(i) \neq vehicle(j)$. The objective is defined as minimizing $\sum_{j \in J} EDT_j$.[4] In this work and in [11], the routing problem is solved using constraint programming and LNS. The LNS relaxation procedure is implemented adaptively based on the ratio of completed searches to searches that reach some predefined failure threshold. The relaxation neighborhoods include both the pickup and dropoff task locations, as well as the vehicle assignments for pickups and dropoffs.

[4] This objective approximates the true power restoration objective and is tight when all restoration actions restore similar amounts of power. When combined with constraint injection, this approximation works well in practice.

Algorithm 2 The OSCO Algorithm for Joint Assessment and Repair

OSCO($set\{Scenario\}\ H$)
$H^* \leftarrow H$
$\sigma^* \leftarrow choosePlan(0, \epsilon, H^*)$
while $|H^*| > 1$
do $r \leftarrow earliestRepairVisit(\sigma^*)$
  $H^* \leftarrow updateScenarios(H^*, r, \xi(r))$
  $\sigma^* \leftarrow choosePlan(time(r, \sigma^*), \sigma^*, H^*)$

3 Joint Assessment and Repair

The problem description and corresponding algorithmic solution described in the previous problem assumes that the extent of the damage inflicted upon the network by the disaster is known precisely a priori. This is not always the case in practice. In general, for hurricanes, there is a set of hazard scenarios that describe the possible network damages. Each scenario h is obtained by tracing the path of a hurricane using a fragility simulator and specifies a set D_h of damaged sites, the distance matrix t_h between any two sites since transportation infrastructure may also be damaged, and the probability of that particular scenario occurring, p_h.

This section details the main contributions of this work–i.e., algorithms that aid in the joint damage assessment and restoration of the power infrastructure. The algorithms use the damages corresponding to the different h in order to guide the assessment and restoration. For each damage site r over all h, it is unclear a priori whether or not the site is damaged. This damage uncertainty, represented by the random variable $\xi(r)$, is revealed when a crew visits r.

To implement joint assessment and repair, this work considers two realistic situations: the case where repair crews are the only available resources and the case where exploration crews are also available. The repair crews use trucks as transportation, and the exploration crews use faster vehicles (e.g., helicopters), but cannot perform repairs.

For simplicity, one assumption of this work is that the exact infrastructural damage is captured by some $h \in H$, that is, some hazard in our global pool of possible hazards. This assumption can be lifted by running the selected algorithm (described below) multiple times if it finds that none of the scenarios in the pool match reality. Indeed, it suffices to sample a new set of scenarios conditional on the revealed uncertainty, and then to rerun the algorithm with this new scenario pool.

3.1 The OSCO Algorithm

One interesting method for addressing the issue of joint infrastructure repair and assessment is to use online stochastic combinatorial optimization (OSCO). The key idea behind OSCO algorithms is to make decisions online, that is, to re-generate near-optimal routing plans as we uncover $\xi(r)$ for each r. These decisions are taken by relaxing the anticipatory constraints (i.e., when the future is revealed), and using optimization algorithms on the resulting offline problems. In particular, relaxing the anticipatory constraints in this problem means that the algorithm assumes that all damages in each scenario are revealed immediately. The basic structure of the OSCO algorithm is provided in Algorithm 2. The algorithm is executed until a single hazard scenario h^* is left, at which point it suffices to follow the selected repair plan.

Algorithm 3 Plan Selection in the OSCO Algorithm

ChoosePlan($Time\, t,\ Plan\, \sigma,\ set\{Scenario\}\, H$)
1 **for** $h \in H$
2 **do** $\sigma_h \leftarrow solveRouting(t,\, \sigma,\, h)$
3 $N \leftarrow \cup_{h \in H}\, nextRepairVisit(\sigma_h)$
4 **return** $argmax_{h \in H}\, p_h \cdot score(\sigma_h,\, N)$

The algorithm starts with the set of all scenarios (line 1). Every iteration of the loop corresponds to a repair crew arriving at a site and assessing its damage. For each such event, the algorithm has the set $H^* \subseteq H$ of possible scenarios and a distinguished plan σ^* that is being executed. The distinguished plan is reconsidered each time a crew reaches a potentially damaged site r, since the uncertainty $\xi(r)$ is now revealed. Line 4 thus only considers the first repair site in the distinguished plan σ^*. It then updates the current set of scenarios given the damaged state of r (line 5) and recomputes the new distinguished plan (line 6). Updating the set of scenarios consists of removing the remaining scenarios that do not share r's *true state*, i.e.,

$$H^* = \begin{cases} \{h \in H^* | \, r \in D_h\} & \text{if } \xi(r) \\ \{h \in H^* | \, r \notin D_h\} & \text{if } \neg\xi(r) \end{cases}$$

Note that multiple vehicles may reach different sites r_1, r_2 at the exact same instant. In this case, the scenario pool is updated according to $\xi(r_i)$ for each $i \in firstVisitedLocations$.

The key step of the OSCO algorithm is to select a distinguished plan based on the first-visited damaged locations of each scenario. To do so, it uses a generalization of the consensus plan proposed in [4]. This consensus plan selection is depicted in Algorithm 3.

The algorithm first determines a set of routing plans for all active hazards H^* respecting the routing decisions taken in the past (up to time t), which are represented in σ (lines 1-2). In particular, if a crew was dispatched to visit a site but has not yet arrived by the time a new set of plans is generated, the new plan will preserve that decision. This step uses the offline optimization discussed earlier, but avoids recomputing the the MRSP and ROP solutions, which are abstracted in the injected constraints. Once all routing plans for H^* are computed, the algorithm builds the set N of all repair sites that are visited next over the total number of newly produced plans. These sites represent interesting locations to visit next and are used to choose which plan σ_h is most desireable overall.

The consensus algorithm [4] ranks the plans by first assigning a score to every potential site N. In particular, for a plan σ, a site s receives a score of 1 if s is visited next by a vehicle in scenario h and 0 otherwise. We refine this scoring mechanism in this paper and assign a score that depends not only on whether a site s is visited next in a plan, but also on the time s is visited, i.e.,

$$score(\sigma,\, s) = TimeHorizon - earliestRepairTime(\sigma,\, s)$$

where $TimeHorizon$ is a large number (e.g., the repair schedule horizon), and $earliestRepairTime(\sigma,\, s)$ specifies the repair time of s in σ or $TimeHorizon$ if s is not repaired in σ. The score of a plan σ is then defined as in the consensus algorithm from [4]:

$$score(\sigma,\, N) = \sum_{s \in N} score(\sigma,\, s)$$

This metric favors the plan which is most similar to other plans, i.e., it visits many of the same repair sites appearing in other plans and tends to visit them at times that are most similar to these plans. This generalization of the consensus algorithm is important, since many damaged sites are close to one another and simply crediting the next visit is too coarse in practice.

3.2 The Two-Stage Algorithm

Let us now consider a case where we have at our disposal exploration crews (e.g., helicopters) and propose the two-stage approach for the joint assessment and repair of power infrastructure. The two-stage approach consists of the following steps:

1. use the exploration crews to identify the real damage scenario
2. solve the offline problem with the damages for this scenario as an input to arrive at a final routing schedule

Since there are multiple exploration crews, the first step is a generalization of the isolation problem used to approximate optimal decision trees and adaptive TSPs under stochastic demands [1].[5] We approach the generalization to multiple vehicles using a greedy algorithm inspired by decision tree algorithms in machine learning (for example,[10]).

Consider a set of disaster scenarios H and a single crew located at site i that is ready to be dispatched to another site. To maximally reduce the uncertainty about the real damaged set, the crew should be sent to the site s that maximizes

$$min(|H| - |H_s^+|, |H| - |H_s^-|)$$

where $H_s^+ = \{h \in H \,|\, s \in D\}$ and $H_s^- = \{h \in H \,|\, s \notin D\}$. However, this expression does not take into consideration the travel time required to visit site s. We therefore normalize this expression to

$$min(|H| - |H_s^+|, |H| - |H_s^-|)/d_{i,s,H}$$

Where $d_{i,s,H}$ is the expected distance from the crew location to the site s over all scenarios H, i.e.,

$$d_{i,s,H} = \sum_{h \in H} p_h \cdot t_h[i, s]$$

Note that we take the expected distance because the travel time matrix may vary according to different disaster scenarios. The 2-stage algorithm generalizes this approach to multiple crews by selecting, for each exploration vehicle v, the next "best" site s_v. It assumes that the number of exploration crews is reasonably small (i.e., not more than 10 - and in this work, approximately 4), which is usually the case in practice. The algorithm dynamically dispatches crews as they become available one at a time, i.e., after they arrive at a particular site and report on its damage state. Assume that at some point in the exploration, there are H scenarios remaining, k crews have been dispatched for assessment, and an additional crew i can now be dispatched to the next potentially-damaged site. Since the damage state of the sites that the k crews are currently visiting is still uncertain (i.e., they could be either broken or not broken), there are 2^k possible outcomes of these explorations. Each such outcome $o \in O$ can be associated with the set of scenarios $H_o \subseteq H$ that contain that particular configuration of damages. The two-stage algorithm then dispatches the available exploration crew to the site that reduces the uncertainty the most across all scenarios after normalization, i.e.,

$$argmax_{s \in S} \sum_{o \in O} min(|H_o| - |H_{o,s}^+|, |H_o| - |H_{o,s}^-|)/d_{i,s,H_o}$$

[5] The isolation problem is solved by a $O(log^2(|S|)log(|H|))$ approximation.

where $H^+_{o,s}$ represents the set of scenarios in H that reflect the configuration prescribed in o with s actually damaged, and $H^-_{o,s}$ is comprised of those scenarios respecting o without s damaged.

3.3 The Hybrid-OSCO Algorithm

The final algorithm, which we refer to as EXPL-OSCO, combines elements of both OSCO and two-stage joint assessment and repair. It proposes proceeding simultaneously with OSCO vehicle routing and the helicopter exploration component of the two-stage approach. As repair crews are routed according to the OSCO algorithm, they reveal (just like in the strictly-OSCO approach) $\xi(r)$ for each r visited first in the current distinguished plan. However, since now there is a possibility that exploration crews have revealed some of the uncertainty regarding the reality damage scenario, even prior to the first instance where repair crews reach a damage node, the EXPL-OSCO algorithm reconsiders its distinguished plan even at pickup locations. The hybrid algorithm updates its scenario pool with the information $\xi(r)$ but also with the information revealed from the exploration.

In general, this approach quickly reduces the number of scenarios to consider (i.e., $|H^*|$) and produces a better consensus plan. It is also possible for the exploration algorithm to use information about revealed damages produced by the OSCO algorithm to reduce the scenarios to consider when dispatching helicopter crews.[6]

[6] In practice, communication in this direction would not yield tremendous gains as the time it takes the OSCO algorithm to reach a delivery site tends to be greater than the entire time required for exploration.

4 Experimental Results

The experimental results capture the behavior of the OSCO, EXPL-OSCO, and two-stage approaches to solving the JIARP. They also show how different factors affect solution quality and algorithm behavior. The results are split into 2 main parts: "Small Damage Size (SDS)" refers to experiments for scenario pools of sizes 20 and 40 for which the minimum restoration set problem (MRSP) has been computed. "Large Damage Size (LDS)" refers to experiments for 20-scenario and 40-scenario pools as well, but to approximate a larger problem (i.e., one with more damaged items), the MRSP has been omitted and the restoration is performed on the entire set of damage items. Note that the Restoration Ordering Problem is still invoked to obtain the partial ordering of constraints to administer per vehicle.

4.1 The Disaster Benchmarks

The algorithms were evaluated on disaster scenarios based on the US power and transportation infrastructures. The disaster scenarios were generated at Los Alamos National Laboratory using state-of-the-art hurricane simulation tools similar to those used by the National Hurricane Center. The benchmarks are for a specific geographic location in the United States with a power transmission network comprised of about 168 items. There were 16 repair crews available for restoration, as well as 4 helicopters crews for exploration. The helicopter crews travel twice as fast as the repair crews.

One benchmark was produced by uniformly sampling 20 scenarios from a pool of 100 disaster scenarios and identifying one of these scenarios as reflective of the true damages. Another benchmark was produced by uniformly sampling 40 scenarios from the 100-scenario pool. The table below illustrates the minimum, maximum, and average number of service tasks for each experimental setup:

	$\|H\| = 20$	$\|H\| = 40$
$MRSP$	min: 14, max: 30, avg: 20.5 (8)	min: 14, max: 29, avg: 21 (20)
$no - MRSP$	min: 38, max: 58, avg: 47.1 (8)	min: 38, max: 58, avg: 47.8 (24)

Here, the numbers in parentheses within each cell signify the number of scenarios in the pool with a number of service tasks greater than or equal to the average over all scenarios.

4.2 Experimental Setting

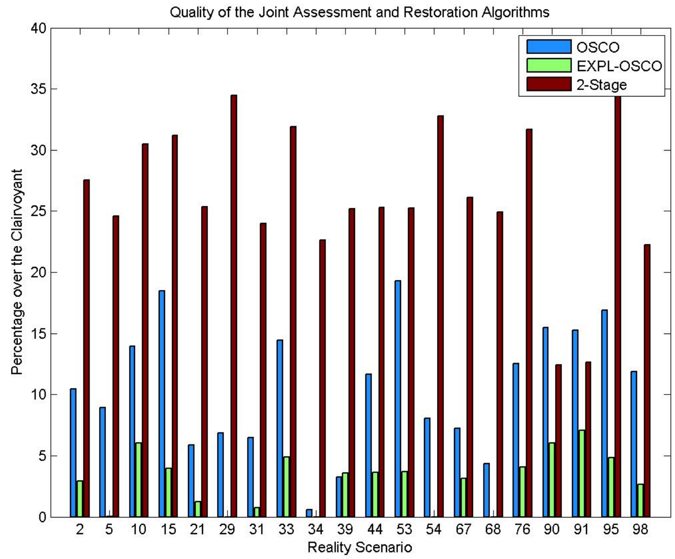
The optimization algorithms were run on Intel Xeon CPU 2.80GHz machines running 64-bit Linux Debian. Due to the fast-response requirement in disaster recovery, the offline routing algorithm for each scenario is terminated after 10 minutes. The routing step is a multi-start, randomized large-neighborhood search algorithm and hence our experimental results report the average over 5 runs.

4.3 Results for Small Damage Size

4.3.1 20-scenario simulations

Figure 2 reports the solution quality for each algorithm executed over 20 scenarios, computing the MRSP. It reports the percentage difference between the clairvoyant (i.e., offline problem where the exact damage set is known) algorithms and the objective values produced by the three JIARP algorithms. In essence, it specifies how much larger the blackout is if the extent of the damages is unknown, compared to the case where there is no damage uncertainty. The

Fig. 2: Solution quality for the JIARP algorithms for 20-scenario, MRSP case



x-axis identifies the scenario selected as the "real damage set" for a particular simulation, and the y-axis depicts the cost (in percentage) of performing joint assessment and restoration.
The results reveal a number of interesting findings. First, the EXPL-OSCO algorithm almost always dominates the other algorithms. On average, EXPL-OSCO produces a solution that is only 3% worse than the clairvoyant, while the OSCO and two-stage algorithms produce solutions that are about 10.5% and 25% worse, respectively. Second, the two-stage algorithm is significantly worse than the other two algorithms in general (with the exception of scenarios 90 and 91, which have damages that are quite different from the other scenarios and hence easier to isolate during exploration), increasing the routing objective by up to 35% in the worst case (e.g., benchmark 29). The EXPL-OSCO algorithm, in contrast, is always under 20%. Benchmark 39 is an outlier where the two OSCO algorithms are very close; it requires very few visits and is sensitive to noise.

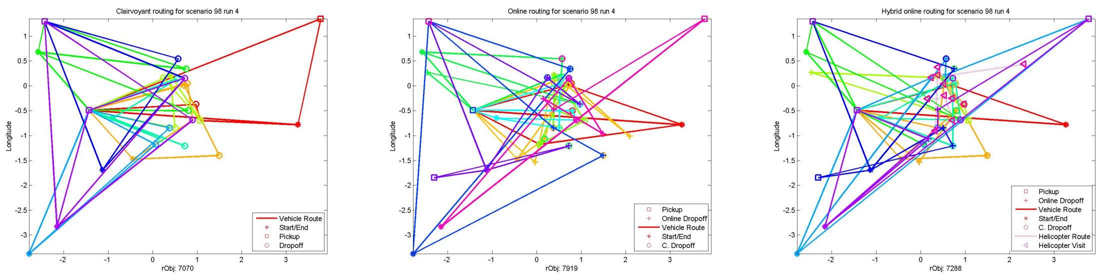
The table in the section 7.1 of the Appendix illustrates the behavior of the OSCO and EXPL-OSCO algorithms. The table reports, over 5 runs and with rounded values, The number IV of site visits[7] before the reality scenario is isolated, the time IT at which reality is isolated, the number AV of additional visits[8] performed by the algorithms compared to the clairvoyant, and the completion time FT of the entire restoration.

The results are particularly illuminating. On average, OSCO and EXPL-OSCO completely reveal the true damage set at approximately 40% and 19% of their respective restorations. This highlights the benefit of the hybrid OSCO and exploration approach. OSCO visits, on average, 10 additional sites compared to the clairvoyant, while EXPL-OSCO adds just under 3 visits. Interestingly, EXPL-OSCO takes only about 1 fewer visit to discover the real damage scenario via exploration crews than OSCO does using repair crews. The visited sites, however, are rather

[7] Here, we only report the number of potential service site visits. Pickup locations visited are omitted from this value to focus on site visits that could reveal some of the damage uncertainty.

[8] This value only includes potential service sites (again omitting pickup locations). It can also be larger than the value IV because it counts vehicles that *eventually* reach a damage site not belonging to reality, not simply those that were reached at the exact time that the damage uncertainty was fully revealed.

Fig. 3: Example vehicle routes for 20-scenario, MRSP simulations



different and the exploration crews decrease the amount of time required to isolate reality by more than a factor of 2. It is also worth observing the difference between AV and IV in OSCO, which arises from the fact that repair crews are committed to a particular repair.

The IT column for EXPL-OSCO also corresponds to the reality isolation time for the two-stage algorithm. Although this step takes only 19% of the restoration of EXPL-OSCO on average, it significantly delays the restoration at the time when the blackout is the largest. This delay, coupled with the fact that the OSCO algorithms start by picking up supplies that they can use to complete many repairs, largely contributes to the two-stage algorithm's relatively poor performance. Based on a scenario pool size and average number of required repairs per scenario both totalling 20, it seems unlikely that the two-stage algorithm could be competitive, even with a more sophisticated exploration scheme. Benchmarks 90 and 91 appear to support this theory. With damages that are very different from other scenarios in the pool, they require very little exploration to isolate the reality scenario, but the two-stage algorithm for these still does not fare well compared to EXPL-OSCO. EXPL-OSCO appears to exploit the strengths of both the OSCO and two-stage approaches: it identifies the real damage scenario very quickly and forces fewer repair crews to make extraneous visits. These combine to only slightly prolong the blackout when it is at its peak.

Figure 3 illustrates an example vehicle and exploration crew routes for the clairvoyant, OSCO, and EXPL-OSCO (labeled as "Hybrid online") algorithms, plotted according to longitude and lattitude. Routes for crews starting at the same location are represented with the same color. On the legend, "C. Dropoff" refers to a service site visited in the clairvoyant routing, "Online Dropoff" refers to a service site visited in either OSCO or EXPL-OSCO, and "Start/End" refers to the start and end locations of a particular vehicle.

4.3.2 40-scenario simulations

Figure 4 illustrates the percentage penalty for each JIARP algorithm, computed over a set of 40 possible damage scenarios.

The trends depicted here are reflective for those found in the 20-scenario simulations. Here, EXPL-OSCO is again the best strategy, increasing the blackout, on average, by only 2%. OSCO increases the blackout by about 9%, and two-stage by 23%. These values are approximately equivalent to the values of 3%, 10.5%, and 25% characterizing the 20-scenario simulations.

There are some differences, however, in algorithm behavior for individual scenarios. For example, scenarios 10 and 38 demonstrate that the maximum percentage delay introduced by

Fig. 4: Solution quality for the JIARP algorithms for 40-scenario, MRSP case

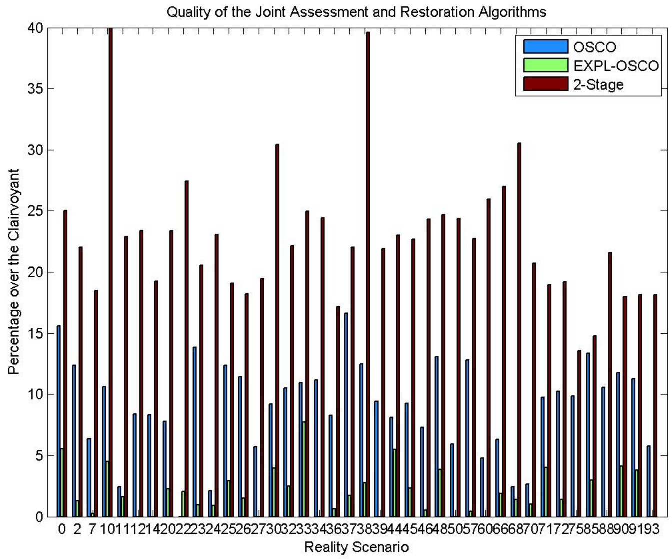
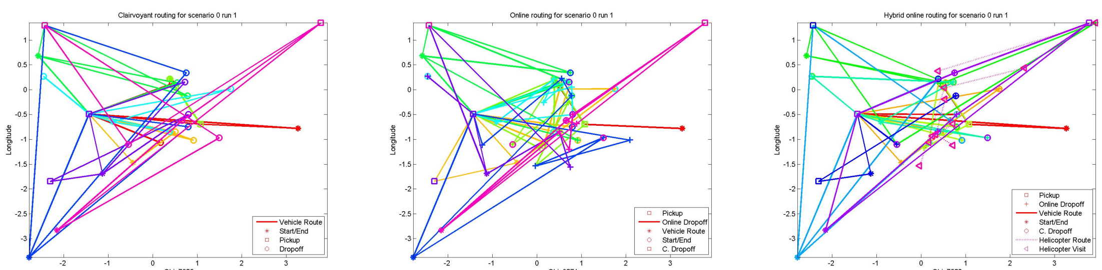


Fig. 5: Example vehicle routes for 20-scenario, MRSP simulations

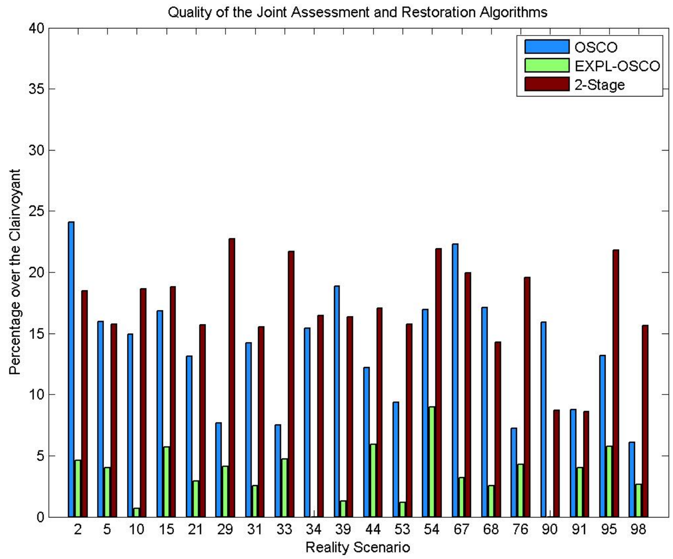


two-stage is now over 40% instead of 35% as in the 20-scenario case. Moreover, the two-stage algorithm for scenarios 90 and 91 no longer dominates OSCO. This is because the isolation step for those scenarios now takes longer (by approximately 40% - 78 units of time instead of 55). These scenarios are outliers, however, as the exploration stage takes approximately 86.5 units of time on average, or 12% less than the 20-scenario case.

Looking at the table in section 7.2 of the Appendix provides an indication of why the results are so similar. The average time of discovery (IT), number of additional visits (AV), and time of last service (FT) for the 40-scenario OSCO and EXPL-OSCO algorithms are all virtually identical to their counterparts in the 20-scenario simulations. Only the number of initial visits (IV) is slightly higher in the 40-scenario experiments for both OSCO and EXPL-OSCO. This most likely stems from the greater number of reoptimizations that the 40-scenario simulations undergo before uncovering reality (a further discussion on this is included in the "Computational Considerations" section below).

Figure 5 shows sample vehicle and helicopter routes for the 40-scenario, MRSP simulations.

Fig. 6: Solution quality for the JIARP algorithms for 20-scenario, no-MRSP case



4.4 Results for Large Damage Size

Scaling the number of damage scenarios without scaling the amount of damage *within* each scenario appears to have a minimal effect on the perceived quality and behavior of the JIARP algorithms. The results below illustrate the ramifications of more service sites for both 20- and 40-scenario simulations on the JIARP algorithms' impact on the size of the blackout.

4.4.1 20-scenario simulations

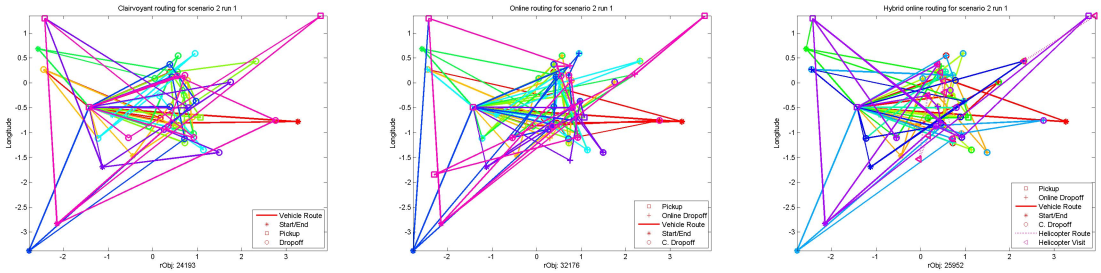
Figure 6 exhibits the solution quality of the JIARP algorithms for the 20-scenario case.

The results show that, on average, EXPL-OSCO increases the blackout by about 3.5%; OSCO by nearly 14%; and two-stage by approximately 17%. The values for OSCO and two-stage differ from the 20-scenario, MRSP simulations, which yielded a 10.5% and 25%, respectively, increase in the size of the blackout. The main reason for the smaller percentage discrepancy between the two-stage and clairvoyant routings is the fact that the results of the exploration, which operated over the entire damage set even in the initial MRSP simulations, are exactly the same, whereas the average final dropoff time is now nearly doubled because of the larger damage sets.

The table in section 7.3 of the Appendix helps explain the behavior of both OSCO and EXPL-OSCO in this context. Looking at EXPL-OSCO, the reality scenario is isolated at less than 10% of restoration, with a discovery time (IT) and number of vehicle visits (IV) before discovery that are identical to their counterparts in the 20-scenario MRSP simulations. This, again, is because the exploration occurs over the exact same damage set as the initial set of 20-scenario simulations. On the other hand, the statistic AV is over 50% smaller, and can be explained by the fact that the larger damage set calls for more pickups per vehicle early on in the routing schedule.

IT for OSCO is nearly 40% greater, however, in comparison to the earlier 20-scenario, MRSP simulations. This means that on average, OSCO algorithms reveal the true damage set at approximately 31% of the restoration time. This value is less than the 40% mentioned for the initial

Fig. 7: Example vehicle routes for 20-scenario, no-MRSP simulations



20-scenario results, but is three times larger than the 10% for EXPL-OSCO. Moreover, even though the number of additional visits (AV) for OSCO averages to approximately two fewer than before, OSCO in this context spends over one more vehicle visit to a damaged location (IV) before reality is uncovered. This, coupled with the later time of reality discovery, suggests that the larger damage set prolongs the amount of time before the uncertainty is revealed. This prolonged uncertainty manifests itself as a prolonged blackout.

Figure 7 shows example vehicle and helicopter routes for the 20-scenario, no-MRSP simulations. Note the denseness of these plots compared to those shown in the previous section. This is because of the larger number of pickups and dropoffs tasked to each vehicle.

These results suggest a key trend: more damages per scenario do not appear to have a significant impact on the behavior or solution quality of EXPL-OSCO, but narrow the gap between the quality of both the OSCO and two-stage approaches.

4.4.2 40-scenario simulations

Figure 8 shows the solution quality for the JIARP algorithms operating on the 40-scenario simulations without the MRSP invoked.

These results show a trend unseen in previous simulations: the average increase in the size of the blackout when using OSCO is 18.6%, followed by 15.1% for two-stage and 4.6% for EXPL-OSCO. These results contrast sharply with the 9%, 23%, and 2% for OSCO, two-stage, and EXPL-OSCO, respectively, in the 40-scenario simulations that invoked the MRSP. The larger routing discrepancies from OSCO can be explained by larger individual differences per scenario. For example, scenario 37 has a discrepancy of nearly 35%, which is approximately twice the percentage difference of 18% found in the MRSP simulations. EXPL-OSCO still dominates both algorithms.

Analyzing the behavior of these algorithms via the chart in section 7.4 of the Appendix helps explain some of the differences. For one, the isolation time (IT) for OSCO is approximately 20% later than in the 20-scenario, no MRSP case, illustrating one of the contributing factors to OSCO falling below two-stage in terms of average solution quality. Compared to the 40-scenario MRSP simulations, IT in this context is 60% greater. The number of sites visited that are actually damaged before reality is revealed (IV) is nearly 13, compared to 8 for the 40-scenario, MRSP instances, while the additional visits (AV) are approximately equivalent. These values suggest that a large number of potential hazard scenarios compounded with a large damage set

Fig. 8: Solution quality for the JIARP algorithms for 40-scenario, no-MRSP case

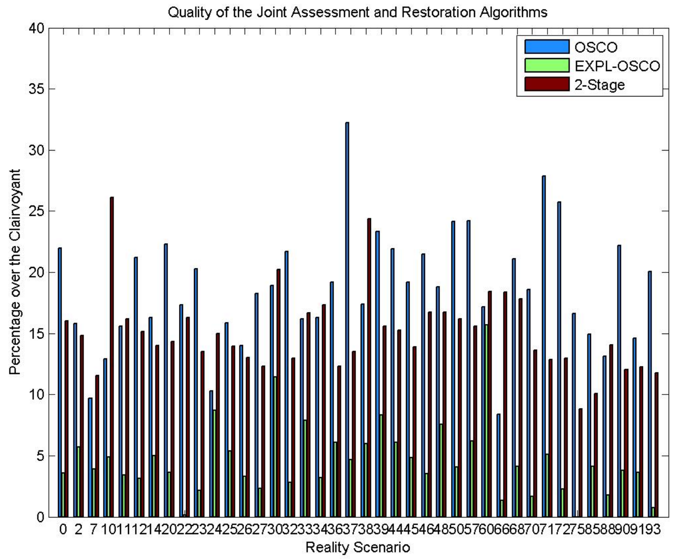
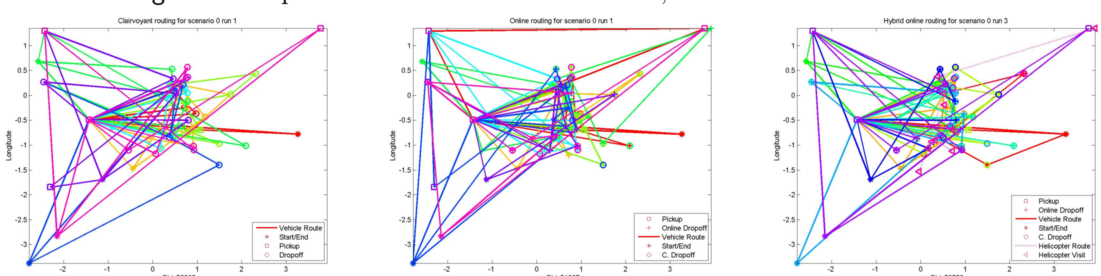


Fig. 9: Example vehicle routes for 40-scenario, no-MRSP simulations



do not necessarily lead OSCO to guide vehicles to more "incorrect" damage locations (i.e., those that are not damaged in reality), but do prolong the amount of time that vehicles spend before learning what the real damage set precisely is. This prolonged uncertainty manifests itself as a prolonged blackout.

For EXPL-OSCO (and hence, for two-stage), the helicopter exploration is identical to the 40-scenario MRSP simulations, and so the statistics IT and IV are also the same. The isolation time (IT) is approximately 12% earlier compared to both 20-scenario simulations. This, coupled with the larger problem instance, explains the lower percentage increase in blackout size when invoking independent exploration and repair stages. EXPL-OSCO in this context, however, has a later finish time (FT) and fewer additional visits (AV) when compared to the 40-scenario MRSP case.

Figure 9 illustrates example vehicle and helicopter routes for the 40-scenario, no-MRSP simulations.

Results for the 40-scenario, no-MRSP simulations confirm a trend that surfaces in the 20-scenario, no-MRSP instances: as damage uncertainty grows in terms of both the number of damages per scenario and total number of possible scenarios, EXPL-OSCO remains the most dominant algorithm, but two-stage overtakes OSCO in terms of its ability to reduce the size of the blackout.

4.5 Computational Considerations

It is important to note the computational complexity that the JIARP adds over the PRVRP, where the extent of the damages are known a priori. For all four simulation scenarios described above, the two-stage algorithm requires the least computational overhead compared to the clairvoyant. The problem consists of simply building the decision tree and greedily routing helicopters to uncover the damage set. The offline clairvoyant routing is then executed to arrive at a feasible solution.

OSCO tends to be the most expensive for each of the simulations, largely because it generally takes the longest to isolate the real damage scenario. Each time it arrives at a potential service location, it runs the offline optimization for each scenario that has not yet been pruned. For both sets of 20-scenario simulations, Each site visit eliminates 3 scenarios on average. This leads to nearly 80 total offline optimizations. For the 40-scenario MRSP simulations, each visit prunes 6 scenarios on average, yielding nearly 155 total offline optimizations. The 40-scenario no-MRSP simulations are more computationally demanding, pruning on average 5 scenarios per service site visit, which results in nearly 180 total offline optimizations.

EXPL-OSCO is over twice as fast as OSCO for 20-scenario MRSP, pruning on average 8 scenarios per site visit and running fewer than 35 total offline optimizations. It is slightly more demaining for the 20-scenario no-MRSP case, pruning 6 scenarios on average per site visit and requiring nearly 40 offline optimizations. In the 40-scenario MRSP context, each site visit prunes nearly 16 scenarios, leading to under 75 total offline optimizations. Slightly more demanding is the 40-scenario no-MRSP context, which eliminates approximately 13 scenarios per visit and requires just over 80 offline optimizations.

EXPL-OSCO computationally dominates OSCO because it also reoptimizes the distinguished plan at each pickup location after some of the damage uncertainty is revealed by the exploration crews. In all, EXPL-OSCO requires fewer than half of the computational resources that OSCO does to produce high-quality results.

It is easy to parallelize the offline optimizations computed before selecting a distinguished plan under both OSCO and EXPL-OSCO because they are independent. Therefore, a small cluster of Linux machines is largely sufficient to meet the real-time constraints imposed by disaster recovery applications.

5 Conclusion

This work discussed the joint assessment and restoration of power systems following hurricane-inflicted damages. It lifted the assumption from previous work [CITE] that the extent of the damage was known prior to determining vehicle routes to service damaged sites. The Joint Infrastructure and Repair Problem (JIARP) receives, as input, a number of potential damage scenarios obtained by weather and fragility simulations. It is then tasked with simultaneously determining the true damage set following a hurricane and repairing damages to eliminate the blackout. Approaches for solving the JIARP are motivated by two cases found in field practices: the case where dedicated exploration crews (e.g., helicopters that cannot perform repairs) are available and the case where only repair crews are available and responsible for both assessment and repair.

Three algorithms are presented as approaches for solving the JIARP: (1) an online stochastic combinatorial optimization (OSCO) algorithm, which uses repair crews to make visitation and repair decisions dynamically as the uncertainty about network damages is repealed upon successive reoptimizations of the routing plan; (2) a two-stage approach that uses exploration crews to survey potential damages to decide on the real damage set and then runs the offline optimization for the scenario determined to depict reality; and (3) a hybrid OSCO approach that performs both exploration and restoration simultaneously, taking advantage of both exploration and repair crews.

Experimental results on hurricane disasters for the U.S. power infrastructure show an interesting trend. For the case of small damage uncertainty (i.e., approximately 20 damages per scenario) when simulating in both 20- and 40-scenario pools, OSCO and EXPL-OSCO yield high-quality solutions and increase the blackout by only 10-11% and 3-4%, respectively. Two-stage, on the other hand, increases the blackout by nearly 25% in both of these cases. As the damage uncertainty is increased from approximately 20 damaged components to over 40 per scenario on average, EXPL-OSCO maintains its dominance (never exceeding a 5% increase in the size of the blackout), but OSCO and two-stage converge in performance. For the case of a large damage set and simulations involving 20 potential damage scenarios, OSCO increases the blackout by nearly 14% and two-stage by 17%. However, as we increase the number of potential scenarios to 40 and hold constant the number of damaged components per scenario, OSCO increases the blackout by nearly 19% and two-stage by only 15%.

These results indicate that EXPL-OSCO is consistently the best choice out of the three algorithms presented for solving the JIARP. Moreover, for reasonably well-behaved stochasticity (i.e., damage amounts that are not too large), OSCO yields high-quality solutions to the JIARP and outperforms the two-stage approach. However, as we increase the damage uncertainty, relying solely on repair crews to handle both assessment and repair of damages in the power network prolongs the blackout. Conducting joint assessment and restoration by using repair and exploration crews in parallel, even with larger damage stochasticity, produces a blackout that is only marginally bigger than one resulting from a clairvoyant routing.

Future work in this area will include exploring how the OSCO, EXPL-OSCO, and two-stage joint assessment and restoration algorithms perform for a wider range of disasters (i.e., beyond hurricanes), as well as in a multi-infrastructure setting (i.e, beyond the power grid). Our current research has already expanded upon the single-commodity power restoration problem presented in this work. Prototype results for a multi-commodity restoration scheme, executed for the 20-scenario, MRSP case, reveal that the percentage increase in the size of the blackout due to damage uncertainty is exactly comparable to the results presented here. This is largely because most electrical components damaged following a hurricane are lines, and thus, the multi-commodity setting and single-commodity relaxation are not so different for hurricane relief efforts.

Swift and efficient disaster response is fundamental to reducing the impact of uncontrollable events on civilizations throughout the world. This work is only a small piece of a larger effort to better understand the ramifications of disasters and how policymakers and relief agencies can used informed decision-making to minimize human suffering. It is our hope that further research bridging the mathematical and humanitarian spheres will help ensure the progress and vitality of societies in the coming years.

6 References

References

[1] V. Nagarajan A. gupta and R. Ravi. *Approximation Algorithms for Optimal Decision Trees and Adaptive TSP Problems*, pages 690–701. Automata, Languages and Programming. Springer, 2010.

[2] D. Alexander. Globalization of disaster: Trends, problems and dilemmas. *Journal of International Affairs*, 59(2):1–22, Spring/Summer 2006.

[3] Dieter Vandenbussche Ann Melissa Campbell and William Hermann. Routing for relief efforts. *Transportation Science*, 42(2).

[4] R. Bent and P. Van Hentenryck. *The Value of Consensus in Online Stochastic Scheduling*. Proceedings of the 14th International Conference on Automated Planning and Scheduling (ICAPS 2004). AAAI Press, 2004.

[5] R.P. O'Neill E.B. Fisher and M.C. Ferris. Optimal transmission switching. *Power Systems, IEEE Transactions on*, 23(3).

[6] P. Van Hentenryck and R. Bent. *Onsline Stochastic Combinatorial Optimization*. The MIT Press, Cambridge, Mass., 2006.

[7] M. Sarma J. Glover and T. Overbye. Power systems analysis and design. *CL-Engineering*, 2007.

[8] K. Wood J. Salmeron and R. Baldick. Worst-case interdiction analysis of large-scale electric power grids. *Power Systems, IEEE Transactions on*, 24(1).

[9] M. Kaaniche J.C. Laprie, K. Kanoun. Modelling interdependencies between the electricity and information infrastructures. *26th International Conference on Computer Safety, Reliability and Security, SAFECOMP-2007, Nurenberg : Allemagne*, 2007.

[10] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[11] C. Coffrin P. Van Hentenryck and R. Bent. Vehicle routing for the last mile of power system restoration. *Proceedings of the 17th Power Systems Computation Conference (PSCC' 11), Stockholm, Sweden*, August 2011.

[12] C.E. Shannon. A mathematical theory of communication. *The Bell Systems Technical Journal*, 27:623–656, July,October 1948.

[13] L.N. Van Wassenhove. Humanitarian aid logistics: supply chain management in high gear. *Journal of the Operational Research Society*, 57:475–489, 2006.

7 Appendix

7.1 Algorithm Behavior - 20 scenarios, MRSP

H	OSCO				EXPL-OSCO			
	IV	*IT*	*AV*	*FT*	*IV*	*IT*	*AV*	*FT*
2	5	215	11	572	6	102	4	542
5	7	222	12	481	5	86	4	454
10	7	191	13	482	6	102	4	432
15	5	206	10	642	6	109	2	521
21	6	226	7	448	5	86	2	390
29	5	206	8	515	7	128	2	482
31	4	213	9	525	5	83	3	459
33	9	241	11	614	7	120	4	531
34	5	218	8	516	5	86	2	518
39	3	160	4	585	6	102	4	577
44	6	235	8	654	6	102	3	579
53	6	243	9	579	5	86	1	457
54	6	235	6	518	6	118	1	475
67	6	219	8	730	6	118	4	701
68	3	221	3	426	5	83	0	386
76	8	284	8	697	7	128	2	630
90	11	269	12	700	3	55	2	678
91	7	253	10	724	3	55	3	721
95	9	227	13	548	7	120	4	455
98	9	260	11	536	5	86	2	517
Av	6.4	227.2	9.9	574.6	5.6	97.8	2.7	525.3

7.2 Algorithm Behavior - 40 scenarios, MRSP

H	OSCO				EXPL-OSCO			
	IV	*IT*	*AV*	*FT*	*IV*	*IT*	*AV*	*FT*
0	12	260	13	564	7	92	3	541
2	7	211	11	611	7	83	4	567
7	6	236	8	466	4	68	3	471
10	8	207	13	473	11	143	4	440
11	4	236	3	493	7	83	2	467
12	11	264	12	615	8	92	3	520
14	8	241	11	619	5	78	2	571
20	5	202	7	418	6	78	3	351
22	0	207	0	425	7	92	2	423
23	7	240	12	699	5	78	3	537
24	7	209	11	597	7	92	4	530
25	5	236	7	703	7	83	1	689
26	7	224	10	561	4	68	5	503
27	11	252	11	576	6	78	2	531
30	6	215	10	537	7	103	2	463
32	10	251	11	516	5	78	3	488
33	8	228	11	637	8	92	3	568
34	13	260	15	543	8	92	4	521
36	8	252	10	583	4	68	2	559
37	7	214	11	622	6	78	3	486
38	10	253	13	531	11	143	6	495
39	8	232	12	652	8	92	3	516
44	5	197	7	642	8	92	3	617
45	8	250	9	499	5	78	3	473
46	11	248	15	563	8	92	4	519
48	9	279	9	616	8	92	1	550
50	5	234	7	444	7	83	2	378
57	8	245	11	641	7	92	2	560
60	8	267	7	544	6	101	5	503
66	6	237	8	535	6	101	2	502
68	5	232	7	434	7	103	1	421
70	5	203	9	538	6	78	3	526
71	11	255	11	624	6	78	3	626
72	13	301	11	632	6	78	2	621
75	7	228	10	637	3	55	2	569
85	9	240	12	612	3	55	3	549
88	11	268	13	639	8	92	2	616
90	10	234	12	698	5	78	1	658
91	9	242	12	709	5	78	2	645
93	6	219	9	629	6	78	3	629
Av	7.9	238.0	10.1	576.9	6.5	86.5	2.7	530.6

7.3 Algorithm Behavior - 20 scenarios, no MRSP

H	OSCO				EXPL-OSCO			
	IV	*IT*	*AV*	*FT*	*IV*	*IT*	*AV*	*FT*
2	12	398	11	1011	6	102	1	928
5	7	260	9	928	5	86	2	944
10	4	311	4	914	6	102	1	887
15	7	326	7	1087	6	109	0	1096
21	9	357	9	938	5	86	1	949
29	10	332	9	995	7	128	1	1050
31	7	250	9	855	5	83	2	898
33	6	244	6	961	7	120	1	1036
34	6	293	7	890	5	86	1	882
39	11	354	9	1142	6	102	1	1106
44	8	288	8	1146	6	102	1	1087
53	6	282	7	883	5	86	1	934
54	8	263	8	1002	6	118	2	964
67	11	367	10	1112	6	118	2	1079
68	9	376	8	985	5	83	1	1046
76	5	272	6	1289	7	128	1	1170
90	5	308	2	1206	3	55	1	1281
91	9	349	5	1126	3	55	1	1289
95	8	353	9	924	7	120	2	925
98	7	309	7	892	5	86	2	964
Av	7.8	314.5	7.48	1013.6	5.6	97.8	1.2	1025.6

7.4 Algorithm Behavior - 40 scenarios, no MRSP

H	OSCO				EXPL-OSCO			
	IV	*IT*	*AV*	*FT*	*IV*	*IT*	*AV*	*FT*
0	9	318	8	1032	7	92	1	1019
2	9	349	11	903	7	83	0	953
7	9	330	8	986	4	68	1	995
10	14	371	14	943	12	143	1	899
11	18	419	15	825	7	83	0	906
12	13	399	11	1102	8	92	3	1196
14	9	355	9	939	5	78	2	1005
20	13	354	14	974	6	78	1	987
22	15	368	15	1004	7	92	1	932
23	17	427	13	1025	5	78	0	972
24	14	375	12	1085	7	92	1	1132
25	18	406	14	1122	7	83	0	1170
26	11	314	11	914	4	68	0	853
27	6	374	6	1215	6	78	2	1096
30	12	359	10	896	7	103	2	855
32	11	354	11	1042	5	78	2	960
33	7	256	7	1059	8	92	1	999
34	10	335	11	892	8	92	2	900
36	13	348	10	1039	4	68	0	1022
37	10	386	9	1200	6	78	0	1036
38	10	368	9	1006	11	143	1	1035
39	11	360	12	1219	8	92	3	1105
44	14	397	12	1148	8	92	0	1124
45	6	356	4	975	5	78	0	966
46	15	366	14	957	8	92	3	910
48	12	397	8	1082	8	92	1	1009
50	15	421	15	943	7	83	2	946
57	16	408	15	1129	7	92	2	1045
60	10	346	8	1070	6	101	1	1185
66	8	287	8	942	6	101	1	984
68	22	451	17	1006	7	103	3	1033
70	11	356	10	1018	6	78	3	982
71	14	440	10	1216	6	78	1	1134
72	15	492	8	1168	6	78	0	1157
75	21	429	20	1164	3	55	3	1124
85	14	366	10	928	3	55	0	976
88	9	346	7	1188	8	92	2	1040
90	11	397	8	1213	5	78	0	1170
91	14	436	9	1182	5	78	1	1176
93	15	439	11	1250	6	78	4	1210
Av	12.5	376.5	10.7	1050.4	6.5	86.5	1.3	1029.8