

Controlling Cascading Failures in Interdependent Networks under Incomplete Knowledge.

Diman Zad Tootaghaj[†], Novella Bartolini^{*}, Hana Khamfroush[†], Thomas La Porta[†]

[†]The Pennsylvania State University (USA), ^{*}Sapienza University (Italy)

{dxz149, hkham, tlp}@cse.psu.edu, {bartolini}@di.uniroma1.it

Abstract—Vulnerability due to inter-connectivity of multiple networks has been observed in many complex networks. Previous works mainly focused on robust network design and on recovery strategies after sporadic or massive failures in the case of complete knowledge of failure location. We focus on cascading failures involving the power grid and its communication network with consequent imprecision in damage assessment. We tackle the problem of mitigating the ongoing cascading failure and providing a recovery strategy. We propose a failure mitigation strategy in two steps: 1) Once a cascading failure is detected, we limit further propagation by re-distributing the generator and load's power. 2) We formulate a recovery plan to maximize the total amount of power delivered to the demand loads during the recovery intervention. Our approach to cope with insufficient knowledge of damage locations is based on the use of a new algorithm to determine consistent failure sets (CFS). We show that, given knowledge of the system state before the disruption, the CFS algorithm can find all consistent sets of unknown failures in polynomial time provided that, each connected component of the disrupted graph has at least one line whose failure status is known to the controller.

Index Terms—Interdependent networks; Cascading failures; Power Grids

I. INTRODUCTION

Needless to say, power grids are one of the most critical infrastructure in our everyday lives. Large-scale blackouts in the power grid due to propagating failures, natural disasters or malicious attacks, can severely affect the operation of other interconnected critical infrastructures and cause catastrophic economic and social disruptions.

In September 2003, a large cascading blackout, in Italy, led to the shortage of 6400 MW of power, which caused a complete system collapse. The cascade began when a tree flashover caused a 380-kV line to fail between Italy and Switzerland [1]. The cascade lasted approximately several minutes, a time sufficient for enabling countermeasures, which could have mitigated and limited the blackout propagation. The main cause of most cascading failures including 2003 Italian and Northeast US-Canada blackout is reported to be inadequate training, planning and operations studies to respond to the emergency [1, 2]. This highlights the necessity of a holistic power control strategy that utilizes real-time monitoring to detect, predict and prevent possible failures. Furthermore, it is crucial to have a strategic recovery plan that ensures effective use of the available resources during the recovery process.

The functionality of the electric power grid and its damage

assessment rely on the operation of a monitoring system. Such a monitoring system utilizes communication lines to interact with power grid controllers, to notify them of detected damage involving overloaded power lines. When a cascading failure affects the power grid, the monitoring system and the communication network are also likely to fail, inevitably compromising the completeness and reliability of damage detection and assessment.

Previous works addressed the problem of cascading failures involving the power grid and the communication network. The majority of these works aimed at characterizing the residual functionality of the networks subject to failure, on the basis of network topology, size and location of the initial damage which caused the cascading phenomenon. Recovery was mostly considered only in the unrealistic case of complete knowledge of the damage, and with interventions aimed at restoring network functionality under the assumption that failure propagation has ended.

In this paper, we address, for the first time, the study of mitigating an ongoing cascade of failures in a power grid and maximizing the provided energy by recovering damaged network elements while the cascade is still in progress and knowledge of the network damages is only partial. Uncertainty of the exact location of the disrupted network components poses a new challenge that has never been successfully tackled. We study the impact of cascading failures in power grids and propose a mitigation strategy in two phases that (1) stops the cascade when the system is still in transient state, and (2) provides a recovery schedule that maximizes the total amount of power delivered to demand loads over all the steps of the recovery process.

In the following, we summarize the most important **contributions**:

- We tackle the problem of mitigating an ongoing cascade (first phase) by formulating the minimum cost flow assignment (*Min-CFA*) problem as a linear programming optimization. *Min-CFA* aims at finding a DC power flow setting that stops the cascading failure at minimum cost. We define the total cost, the total weighted amount of reduced power due to the re-distribution of the power in the generators and loads without violating the overload constraint at each line.
- We study the problems related to the interdependency of the power grid and its communication network and show

that, in the absence of complete knowledge of failure locations, classic cascade prevention approaches may not work as they should.

- We address the recovery phase (second phase) formulating the problem of maximizing the restored accumulative flow (*Max-R*). We show that *Max-R* is NP-hard and propose a heuristic recovery strategy which works under partial knowledge of damage locations by calculating consistent failure sets to locate failures.
- We performed an experimental evaluation, considering cascading failures in a power grid and its monitoring communication network. We use real data from the Italian high-voltage transmission grid (HVIET) and its communication network (GARR) [3, 4, 5]. The experiments show that when 60% of the network is disrupted, our cascade prevention approach (Min-CFA) finds the optimal solution with 54.39% of the demand satisfied. While, without a cascade prevention algorithm, the whole system fails. Furthermore, our backward recovery approach on average delivers 20% more power to the loads with respect to a shadow-pricing approach inspired by the work in [6].

While our recovery approach is proposed for a case study of a power grid and a communication network, our approach invites further work on recovery of other interdependent networks.

The remainder of this paper is organized as follows. Section II discusses the background and motivation behind this work. In section III, we explain the *Min-CFA* and *Max-R* optimization problems and show that *Max-R* is NP-Hard. Section IV describes our algorithms. Section V shows our evaluation methodology and experimental results and Section VI concludes the paper with a summary.

II. BACKGROUND AND MOTIVATION

Most of the research on large-scale failure management has concentrated on the recovery of a single network. Bartolini et al. [7], Al Sabeh et al. [8], Tootaghaj et al. [9] and Wang et al. [10] jointly address the progressive recovery of a single data communication network after a large-scale disruption.

In complex networks however, multiple heterogeneous networks may be interconnected and interdependent. Because of the interdependency between different components, perturbations caused by failures, physical attacks or natural disasters may propagate across the different networks. To study the interactions in a complex network, graph-based models are typically used, where nodes are the system components and edges model the interactions or dependencies between different components of the same or of different networks. A cascading failure may propagate across the nodes of the complex network traversing the dependency edges across a same network or multiple networks, possibly accelerating and eventually resulting in a potentially total failure of the system.

Cascading failures in interdependent networks have been studied in several works [11, 12, 13, 14, 15, 16]. The existing works on interdependent networks can be broadly classified

into three categories: 1) those which study the interaction through percolation theory [14, 15, 16, 17], 2) works which try to identify most vulnerable nodes and design failure resilient networks [11, 18, 19, 20, 21], 3) and the works which try to find the root cause of failures [22, 23]. To the best of our knowledge, the problem of mitigating and recovering from cascading failures, during the transient regime of the propagation process, has not been studied extensively. Percolation/epidemic-based approaches depend on having a prior knowledge about the probabilistic model of failure propagation, which is hard to obtain. In addition, real systems usually have a deterministic failure propagation. For example, if a power line fails, a certain number of communication routers will stop working. Finding the root cause of the propagating failure is shown to be NP-Hard [22], but is the key to design restoration algorithms. Identifying the most vulnerable nodes and root cause of failures helps to design failure-resilient systems but does not provide a mitigation solution when the failure happens in the system.

Cascading failures in power grids can be due to a permanent short circuit, e.g. a tree falls on a transmission line etc., or due to a temporary failure, e.g. a temporary short circuit in a transmission line. When a short circuit happens in one of the transmission lines, the controller sends a "trip" signal to the breakers and the breakers set open. The controller tries to connect the breaker multiple times before the line fails. In case of a permanent failure, the breaker stays open circuit. After a line fails in the system, the power re-distributes according to Kirchhoffs and Ohms laws. This can cause other lines to be overloaded and trigger new failures. The cascaded failures can trigger multiple times and spread over the entire network. Unlike the approach proposed in [18], that re-distributes the power flow evenly over all transmission lines, we use the DC power flow model [24, 25] which is widely used in studies of cascading failures.

The operation and reliability of today's power grid is highly dependent on the operation of the communication network that provides the necessary information needed by the supervisory control and data acquisition (SCADA) system to respond to emergency situations. The required data is measured and gathered at the substations from the intelligent electronic devices (IEDs), control circuit breakers and phasor measurement units (PMUs) [26, 27]. While the security of the control system is itself an important challenge on the reliability of the power grids (e.g. a compromised controller can send a trip signal to disrupt the power grid) [28, 29], we focus on the interdependency between the operation of the monitoring system and the controller to avoid the cascaded failure.

III. PROBLEM DEFINITION

We consider a complex system for which some failures are detected while the propagation is still in the transient regime. We propose a mitigation strategy to avoid further cascade and a recovery plan to maximize the total operability of the network during K steps of recovery. We define the power grid

TABLE I: Summary of notations.

Notation	Explanation
$G_p = (V_p, E_p)$	undirected graph modeling the power grid. V_p is the set of nodes and E_p is the set of links
$G_c = (V_c, E_c)$	undirected graph modeling the communication network. V_c is the set of nodes and E_c is the set of links
$G_i \in V_p$	generator node $G_i \in V_p$ where the power is inserted
$L_i \in V_p$	load node $L_i \in V_p$ where the power is extracted
$J_i \in V_p$	junction node $J_i \in V_p$ where the power just flows by
$E_p^{B,t} \subseteq E_p$	set of broken edges in the red area
$E_p^{U,t} \subseteq E_p$	set of edges in the grey area whose failure patterns is unknown
$E_p^{W,t} \subseteq E_p$	set of edges in the green area which are known to be working correctly
F_{ij}^t	power flow in line (ij) at time t
θ_i^t	voltage angle of node i at time t
x_{ij}	series reactance of line (ij)
P_i^t	power generated/consumed at node i at time t
B^t	nodal admittance matrix at time t
w_{G_i}	weighted cost of increasing the power in generator G_i
w_{L_i}	weighted cost of shedding the power of load L_i
$P_{G_i}^{max}$	maximum power that can be generated in G_i
$P_{L_j}^{demand}$	demand load at L_j
F_{ij}^{max}	maximum capacity of the line (ij)
E_k^R	set of restored edges at iteration k
$\delta_{(ij),k}$	decision variable to repair $(ij) \in E_p^{B,t}$ at the k th iteration
r_{ij}	resources needed for repairing (ij)
R_k	available resource at iteration k of the recovery

operability to be the accumulative amount of power delivered to satisfy the demand load over the K recovery steps. Our approach can be extended to the use of other operability measures such as the total number of working power lines etc. Table I shows the notation used in this paper.

The power and communication networks are modeled as undirected graphs $G_p = (V_p, E_p)$ and $G_c = (V_c, E_c)$ respectively. Transmission lines are monitored by several sensors deployed nearby that area. The aggregated data are then sent to closest communication node and to the control center. Also the control commands are sent to the closest communication node. Therefore, each power line is monitored and controlled through the closest communication node. Each node $i \in V_p$ in the power grid can be 1) a generator G_i , where the power is inserted, 2) a load L_i , where the power is extracted, or 3) a junction J_i where power flows by. As transformer and generator failures are extremely unlikely, we hereby assume that failures only occur in power lines (E_p). Further, we consider the inter-dependency between the power grid and the communication network such that failures in the communication network would lead to lack of information in the control center. We assume that the communication network gets power from an emergency source in case of failures in the power grid and ignore the ping pong failures between the two networks. The edges in the power grid graph G_p may be in three different states:

- 1) the set $E_p^{B,t} \subseteq E_p$ is the set of certain **broken** edges

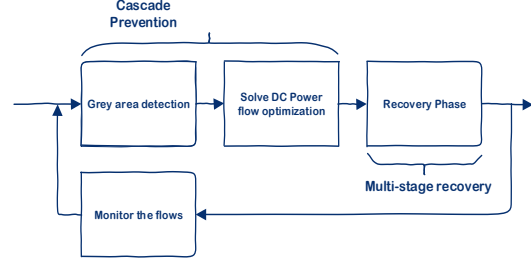


Fig. 1: Recovery Process: 1) Re-distribution of power, 2) Recovery phase.

- (hereby denoted as red edges) at time t ¹.
- 2) the set $E_p^{U,t} \subseteq E_p$ is the set of edges of unknown working status (denoted as grey edges) at time t ,
- 3) the set $E_p^{W,t} \subseteq E_p$ is the set of certain working edges (denoted as green edges) at time t .

A. 2-phase Recovery approach: Power grid case study

In this section, we study the mitigation of cascading failure and related recovery process in a power grid. Figure 1 illustrates the two phases of this process: 1) mitigation of the cascade using a combination of load shedding and adjustment of the generated power, and 2) recovery phase.

1) *Cascade mitigation (Min-CFA)*: We model the cascading failure in a power system using a DC load flow model [24]. The DC power flow model provides a linear relationship between the active power flowing through the lines and the power generated/consumed in the nodes, which can be formulated as follows:

$$F_{ij}^t = \frac{\theta_i^t - \theta_j^t}{x_{ij}}, \quad (1)$$

where, F_{ij}^t is the power flow in line (ij) at time t , x_{ij} is the series reactance of line (ij) and θ_i^t and θ_j^t are the voltage angles of node i and j at time t . The power flow of node i can be found by summing up the power flows of all its adjacent power lines:

$$P_i^t = \sum_j F_{ij}^t \quad (2)$$

We can re-write the power flow model as a linear system of equations as follows:

$$P^t = B^t \theta^t \quad (3)$$

where B^t is nodal admittance matrix at time t , $b_{ij}^t = -\frac{1}{x_{ij}^t}$ for $i \neq j$ and $b_{ii}^t = \sum_k \frac{1}{x_{ik}^t}$.

Once a transmission line trips, the power is redistributed according to Equation (3) and if the power exceeds the

¹Notice that in order to be able to assess an edge damage, the edge must be connected to a working communication node in G_c . The working communication node provides the failure status of the edge to the central controller and can send power adjustment commands to the connected loads or generators.

maximum threshold on another line (ij) , the transmission line (ij) will also disconnect unless we reduce the total load or redistribute the generated power.

Theorem 1. *The power flow model (Equation 3) is always solvable for each connected component of the power graph.*

Proof. The nodal admittance matrix, B , of a connected graph with n nodes is always $\text{rank}(B) = n - 1$ because one can construct a graphic matroid from a given graph where the nodal admittance matrix is a weighted incident matrix. It is known that the rank of a weighted incident matrix is equal to the rank of any basis (tree) in the graph which is $n - 1$ [30, 31]. To make this equation solvable, one of the equations is removed and the node associated with that equation is chosen as a reference angle $\theta_1 = 0$. If the graph has c connected components, the rank of its admittance matrix is $n - c$. Therefore, the DC power flow model for each connected component of the graph has a unique solution. \square

Once we detect an outage of the transmission line, we readjust power and load according to the optimization problem described in the following. The Minimum Cost Flow Assignment (*Min-CFA*) optimization problem minimizes the total cost of reducing the load or generator's power. Let w_{G_i} be the weighted cost of reducing the power in generator G_i and w_{L_i} be the weighted cost of decreasing the power of load L_i . The *Min-CFA* problem to avoid the cascaded failures can be formulated as follows:

$$\begin{aligned}
& \text{minimize} && \sum_{G_i, L_j \in V_p} w_{G_i}(P_{G_i}^0 - P_{G_i}^t) - w_{L_j}(P_{L_j}^t - P_{L_j}^0) \\
& \text{subject to} && 0 \leq P_{G_i}^t \leq P_{G_i}^0, \quad \forall G_i \in V_p^t \\
& && 0 \leq P_{L_j}^t \leq P_{L_j}^{\text{demand}}, \quad \forall L_j \in V_p^t \\
& && -F_{ij}^{\text{max}} \leq F_{ij}^t \leq F_{ij}^{\text{max}}, \quad \forall (ij) \in E_p^t \\
& && \sum_{G_i, L_j \in V_p} P_{G_i}^t + P_{L_j}^t = 0. \\
& && P_{G_i}^t = \sum_j F_{ij}^t, \quad \forall G_i \in V_p^t, (ij) \in E_p^t \\
& && P_{L_i}^t = \sum_j F_{ij}^t, \quad \forall L_i \in V_p^t, (ij) \in E_p^t \\
& && P_{G_i}^t = B^t \theta^t, \quad \forall G_i \in V_p^t \\
& && P_{L_j}^t = B^t \theta^t, \quad \forall L_j \in V_p^t \\
& && F_{ij}^t = \frac{(\theta_i^t - \theta_j^t)}{x_{ij}}, \quad \forall (ij) \in E_p^t
\end{aligned} \tag{4}$$

The first constraint indicates that the power generated at each generator cannot exceed the initial power at each generator. If we had full knowledge about the location of failures, we could have a more relaxed constraint to increase the power at some of the generators without violating a maximum threshold. However, under uncertain failure we reduce our solution space to decrease the possibility of consequent cascades due to unknown knowledge. The second constraint shows that the

reduced load cannot exceed the demand. The third constraint shows that the power flowing through each line cannot exceed the maximum capacity of the line. The fourth constraint is the power conservation condition, i.e. the total power generated in the generators should be equal to the total power consumed in the loads. The fifth and sixth constraints show that the total power generated/consumed at each node should be equal to the total power flow through its edges. The last three constraints reflect the DC power flow model.

2) *Recovery Phase (Max-R)*: In the general cascading failure model, suppose that recovery of each failed power line $(ij) \in E_p^{B,t}$ leads to the restoration of $\sum P_{L_j}^k(\text{Rep}_k)$ power units in the loads' demand. Where $\text{Rep}_k = \{(i, j) \in E_p^{W,k}\}$ is the set of restored and working power lines at iteration k . Also, suppose that at each iteration k of the recovery R_k resources are available and repairing (ij) needs r_{ij} resources. The maximum recovery (*Max-R*) optimization can be modeled as a mixed integer programming where we maximize the accumulative delivered power over K steps of the algorithm. Assuming that at each iteration we have enough resources to repair at least one disrupted edge, we set K to be the total number of disrupted edges. Let E_k^W be the set of lines which have been restored or are working up to time step k and let E_k^R be the set of restored edges up to iteration k . The *Max-R* recovery problem is formulated as follows:

$$\begin{aligned}
& \text{maximize} && \sum_{k=1}^K \sum_{L_j \in V_p} P_{L_j}^k(\text{Rep}_k), \\
& \text{subject to} && \sum_{m=1}^k \sum_{(ij) \in E_k^R} \delta_{(ij),m} \cdot r_{ij} \leq \sum_{m=1}^k R_m \quad k = 1, \dots, K, \\
& && \sum_{k=1}^K \delta_{(ij),k} \leq 1, \quad \forall (ij) \in E_k^R \quad k = 1, \dots, K, \\
& && \delta_{(ij),k} \in \{0, 1\}, \quad \forall (ij) \in E_k^R \quad k = 1, \dots, K,
\end{aligned} \tag{5}$$

where $\delta_{(ij),k}$ is the decision variable to repair $(ij) \in E_p^B$ at the k th iteration of the algorithm. The first constraint indicates that at iteration k of the recovery, R_k resources are available; if the resources are not used in the k -th iteration of the recovery, the unused resources can be used in the following steps. The second constraint shows that each broken line can only be repaired once. Note that the total delivered power in the objective function changes with respect to the recovery schedule. The objective function is the accumulative power flow measured at the loads in the K steps of execution of the algorithm. With $P_{L_j}^k(\text{Rep}_k)$ we denote the power received by load L_j when the recovery decision $\delta_{(ij),k}$ is made up to step k leading to the restoration of the power lines Rep_k . One needs to re-solve the DC power flow optimization problem to find $\sum P_{L_j}^k(\text{Rep}_k)$ since the set of working lines, Rep_k , at time step k changes based on the current and previous decisions of the recovery schedule $\delta_{(ij),k}$. Note that in the recovery phase, we remove the generator's power reduction constraint and the generator and load's power increases

gradually until all demand loads are satisfied.

Theorem 2. *The problem of Max-R is NP-Hard.*

Proof. We prove the NP-hardness of the Max-R problem showing that it generalizes the Knapsack problem. We recall that the Knapsack problem considers a set of items I , each item $i \in I$ has a size S_i and a value $V_i > 0$. The problem is to find a subset $I' \subseteq I$ such that $S(I') \leq S$ and $V(I')$ is maximized, where $S(I') = \sum_{i \in I'} S_i$ and $V(I') = \sum_{i \in I'} V_i$.

In the following we show how we can build, in polynomial time, an instance of a single stage ($K = 1$) of Max-R problem whose solution corresponds to the solution of the generic formulation of the Knapsack problem given above.

Since we consider a single stage of the Max-R problem, we assume R resources are available to repair all disrupted lines $(ij) \in E_p^{B,t}$. We also assume that we have complete information about the disrupted lines. Let us consider a set of generators I , each generator corresponding to an element $i \in I$ of the Knapsack problem, producing a flow equivalent to the value V_i of the element. Each generator $i \in I$ is connected to a unique common load L with a broken line, whose repair cost is equivalent to the size S_i of the corresponding Knapsack element. We also assume that the load L has a demand of at least the summation of all flows ($\sum_{i \in I} V_i$). We set the recovery budget of Max-R equal to S , the size of the Knapsack. This instance of Max-R can be defined in polynomial time starting from any instance of Knapsack. Solving this instance of Max-R, corresponds to finding a list of links to be recovered with cost limited by S , such that the flow reaching the common load L is maximized, which is equivalent to selecting the Knapsack subset $I' \subseteq I$ with maximum value, and bounded size S , which completes the proof that any instance of the Knapsack problem can be polynomially reduced to the solution of an instance of Max-R, which implies the NP-hardness of Max-R. \square

As Max-R is NP-hard, we consider two polynomial time heuristics, (Max-R-shadow-pricing) and (Max-R-Backward) in Section IV.

Remark: Note that the maximum recovery problem is a combinatorial optimization and the total flow that each line can add to the final solution of the problem is unknown in advance and depends on the recovery schedule of other lines. The marginal flow that each line can add to the current solution of the problem can be found by solving the *Min-CFA* problem introduced in section III-A1 which itself is a linear programming optimization. We call the marginal utility (flow) added by recovery of each line the "shadow price" referring to the amount of flow assigned to the currently unknowable value of the flow that can be added to the final solution by repairing a broken line.

We now consider an example where the underlying communication network is disrupted and therefore, the controller fails to make appropriate decision to stop the cascade. We then propose a consistent failure set (CFS) algorithm in

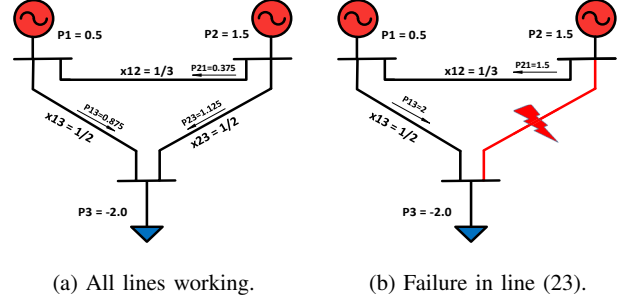


Fig. 2: An example of a 3-bus network where active power and reactances are in pu.

Section IV-A to cope with lack of knowledge.

An illustrative example: Consider the network given in Figure 2, using the DC power flow model to calculate the power flows in the lines, where the reference angle is $\theta_1 = 0$, we have:

$$\begin{pmatrix} \theta_2^0 \\ \theta_3^0 \end{pmatrix} = \begin{pmatrix} 5 & -2 \\ -2 & 4 \end{pmatrix}^{-1} \begin{pmatrix} 1.5 \\ -2 \end{pmatrix} = \begin{pmatrix} 0.125 \\ -0.4375 \end{pmatrix} \quad (6)$$

The power flow through each line is then computed as follows:

$$F_{12}^0 = \frac{\theta_{12}^0}{x_{12}} = 3 \times (0 - 0.125) = -0.3750, \quad (7)$$

$$F_{13}^0 = \frac{\theta_{13}^0}{x_{13}} = 2 \times (0 - (-0.4375)) = 0.875, \quad (8)$$

$$F_{23}^0 = \frac{\theta_{23}^0}{x_{23}} = 2 \times (0.125 - (-0.4375)) = 1.125. \quad (9)$$

If the power line 23 gets disrupted as in Figure 2b, the power redistributes according to DC power flow model, where $F_{21}^1 = 1.5$ and $F_{13}^1 = 2$. Suppose that the maximum power that each line can tolerate is $F_{ij}^{max} = 1.3$. Therefore, after the first line gets disrupted, the whole system collapses and the demand load cannot be satisfied. However, if we know the exact location of the failure, the controller may reduce the generator's power to satisfy a degraded quality of service. One trivial solution of *Min-CFA* to this problem is to reduce the second generator's power to $P_2^1 = 0.8$ and reduce the load to $P_3^1 = -1.3$ without violating the maximum power on each line. However, under the uncertainty of the exact location of the failure, the controller fails to make appropriate decisions and the whole network collapses.

IV. METHODOLOGY

In this section, we first describe the consistent failure set approach to detect the status of grey lines. Then, we describe two heuristic algorithms to solve Max-R. Inspired by the proposed approach in [6] that finds a progressive recovery schedule in a data communication network, we first propose a shadow price-based approach with polynomial time complexity and then propose a polynomial time backward approach that solves a single stage of the problem and traces back until it finds the recovery schedule for all stages.

A. Finding a Consistent Failure Set (CFS)

In order to detect the grey area, we use an algorithm, which starts with the nodes that have the smallest number of grey edges.

Lemma 1. *In the power grid graph G_p , if there exists a node $n_i \in V_p$ which has only one grey neighbor link $e = (n_i, n_j) \in E_p^t$, the exact status of the grey edge e can be discovered.*

Proof. The exact status of a single grey edge attached to a node n_i can be determined using the power flow equation 2, i.e. the power generated/consumed at node n_i can be found by summing the power flow of all its adjacent power lines. \square

Lemma 2. *If the grey area does not contain a cycle and there exists at least one edge in the power grid graph G_p whose status is known, the exact status of all grey edges can be found in $O(|E_p^{U,t}|)$.*

Proof. If the grey area does not contain any cycles, there exists at least one node that has only one grey neighbor link e and therefore, according to lemma 1, the exact status of e can be found. This procedure can be repeated to find the status of all grey edges in $O(|E_p^{U,t}|)$. \square

For the case study of a graph G_p which has one or multiple cycles in its grey area, we propose a consistent failure set rule to detect the exact status of unknown transmission lines. We assume the power generated/consumed in each generator/load or junction is known before the disruption. The algorithm starts by finding the status of grey edges, which are not within a cycle and are the only grey neighboring node of one of its end points. If all nodes have at least two grey edges in the graph, i.e. there exists a cycle in the grey area, and our algorithm picks a node within a cycle with the minimum number of adjacent grey edges and makes a decision tree. The algorithm tries to solve the unknown status of the grey edges by assuming one edge at each cycle to be working or not working, and solving the rest of DC power flow to see if the assumption is correct. If the assumption is not correct, the algorithm chooses another branch of the decision tree until it finds a consistent failure set. In cases where there exists multiple consistent failure sets, the algorithm performs a local inspection of an edge whose status is different from the possible solutions and picks the solution, which is consistent with the result of the local inspection. Algorithm 1 shows different steps of CFS.

Theorem 3. Complexity Analysis: *Assuming the grey area becomes a tree by removing C edges, CFS algorithm runs in $O(2^C |E_p^{U,t}|)$.*

Figure 3 shows an example of a network with 6 grey edges and shows different steps of the CFS algorithm. In the first step, the status of all edges with a single grey adjacent edge are designated. In the second step, the decision tree makes two branches to remove the cycle, and solves the DC flow

TABLE II: Average number of local inspections needed as the size of the grey area increases in the Italian power grid network.

Percentage of disrupted monitors	Average number of grey edges in the Italian power grid	Average # of grey edges within a cycle
10	25.25	3.74
20	62.49	7.15
30	92.06	10.04
40	124.16	13.35
50	157.6	16.84
60	193.29	21.52
70	227.59	26.95
80	265.49	32.71
90	303.5	40.06

Algorithm 1: Consistent Failure Set (CFS) algorithm.

Data: A set of grey lines $(ij) \in E_p^{U,t}$ whose failure status is unknown, the graph of the network $G_p = (V_p^t, E_p^t)$, the power generated at each generator $P_{G_i} \forall G_i \in V_p^t$, the power consumed at each load $P_{L_i} \forall L_i \in V_p^t$

Result: The status of edges in the grey area $(ij) \in E_p^{U,t}$, which can be failure or working.

- 1: $C =$ Number of edges in $E_p^{U,t}$ that need to be removed to make the grey area cycle-free
 - 2: **if** $C > 0$ **then**
 - 3: pick an edge at each cycle to generate a cycle-free grey area
 - 4: for all 2^C combination of the chosen edges at each cycle, run CFS-Cycle-Free($E_p^{U,t}, G_p, P_{G_i}, P_{L_i}$) to find a consistent failure set
 - 5: **else if** $C = 0$ **then**
 - 6: run CFS-Cycle-Free($E_p^{U,t}, G_p, P_{G_i}, P_{L_i}$).
 - 7: **end if**
 - 8: **return** $E_p^{B,t}, E_p^{W,t}$
-

optimization for each branch to find a consistent failure set. Assuming edge $(23) \in E_p^{B,t}$ was broken, we do not find a consistent feasible solution and therefore we assume $(23) \in E_p^{W,t}$ is working. The last graph shows a consistent failure set of broken and working edges.

In cases where we have multiple consistent failure sets, we perform a local inspection of the edges whose failure status is different from the possible consistent solutions, and pick the solution consistent with the local inspection.

Table II shows the average number of grey edges within a cycle, in the Italian power grid network [3, 4, 5] when the size of the disrupted communication network (garr) increases from 10% to 90% for 100 different random selection of disrupted communication nodes. Assuming all possible failures within a cycle are consistent with known information, we only need a maximum of 10% local inspection of the grey edges.

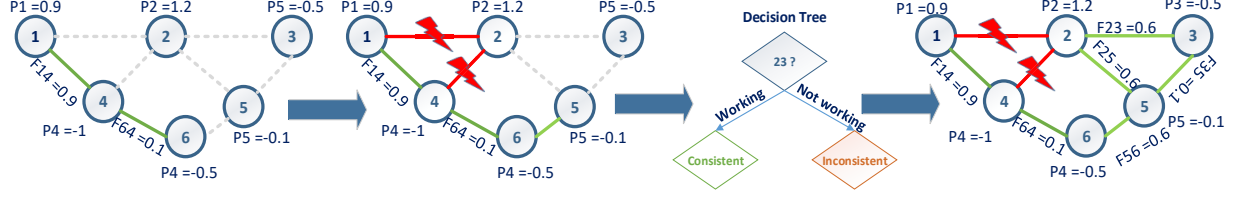


Fig. 3: An example of a 6-bus network with 6 grey edges and different steps of CFS algorithm.

Algorithm 2: CFS-Cycle-Free

```

1 Function CFS-Cycle-Free ( $E_p^{U,t}, G_p, P_{G_i}, P_{L_i}$ )
2    $greys = \operatorname{argmin} \{ (i,j) \in E_p^{U,t} \}$ ;
3   while  $greys = 1$  do
4     Select a node  $i \in V_p$  with one grey neighbor
5      $greys = \operatorname{argmin} \{ (i,j) \in E_p^{U,t} \}$ ;
6     detect whether  $(i,j) \in E_p^{U,t}$  is working or not
7     using equation 2.;
8     if there exists no solution from equation 2 then
9       return inconsistent;
10    break ;
11    if  $(i,j) \in E_p^{U,t}$  is working then
12       $E_p^{W,t} = E_p^{W,t} \cup (i,j)$  and  $E_p^{U,t} = E_p^{U,t} \setminus (i,j)$  ;
13    else
14       $E_p^{B,t} = E_p^{B,t} \cup (i,j)$  and  $E_p^{U,t} = E_p^{U,t} \setminus (i,j)$ ;
15  return consistent,  $E_p^{B,t}, E_p^{W,t}$  ;

```

B. Identifiability of voltage phasors

When the network is divided into a known and unknown part we can re-write the DC power flow equations as follows:

$$\begin{pmatrix} B_{known} \\ B_{unknown} \end{pmatrix} \times \begin{pmatrix} \theta_{known} \\ \theta_{unknown} \end{pmatrix} = \begin{pmatrix} P_{known} \\ P_{unknown} \end{pmatrix} \quad (10)$$

Therefore, the unknown voltage phasors can be found as follows:

$$B_{known} \times \begin{pmatrix} \theta_{known} \\ \theta_{unknown} \end{pmatrix} = P_{known} \quad (11)$$

Let $N = \operatorname{Null}(B_{known})$ denote the null space of B_{known} , i.e., for any vector $n \in \operatorname{Null}(B_{known})$, $B_{known} \cdot n = 0$.

Theorem 4. Voltage phasor θ_i is identifiable, if and only if $\forall n \in N$ we have $n_i = 0$.

Therefore, in order to find a set of identifiable voltage phasors $\theta_i \in \theta$, we can first compute the null space of B_{known} and find all indices with zero values in the null space. The null space of B_{known} gives the number of identifiable voltage angles. If the value of the voltage phasor of θ_i is not identifiable, we have to perform a local inspection to find the value of voltage angles for non-identifiable nodes.

C. Identifying the failures

After identifying all voltage phasors, one can identify the unknown admittance matrix if the grey area does not contain any cycles.

$$B_{unknown} \times \begin{pmatrix} \theta_{known} \\ \theta_{unknown} \end{pmatrix} = P_{unknown} \quad (12)$$

Note that the value of the $P_{unknown}$ is determined from the previous state of the disruption. We assume the powers at the generators and loads are only controlled through the central controller unit and therefore since the controller has not increased or reduced the power $P_{unknown}^t = P_{known}^{t-1}$. Therefore, we can find the state of the network for all grey edges, which are not inside a cycle. In case of having a grey cycle we use the consistent failure set algorithm to remove the cycles and find a consistent set. If the consistent failure set algorithm finds multiple solutions, we pick one by performing a local inspection.

D. Max-R-shadow-pricing

Since the total value of the flow that each repaired line can add to the solution is not known in advance, we use a shadow pricing technique, which is used to assign values to the unknown value of repaired edges in the power grid graph. At each stage k , the shadow-pricing algorithm, repairs the transmission lines $(i,j) \in E_p^{B,t}$, which add the maximum to the total delivered power over the required resource, i.e. $\operatorname{argmax}_{(i,j)} (F_{ij}/r_{ij})$, until the total available resources for stage k are used. Algorithm 3 shows different steps of the *Max-R-shadow-pricing* algorithm. The algorithm starts with the disrupted network and computes the value of the flow added to the current state of the network divided by the total number of resources it needs, and repairs the power line that maximizes this value. This procedure repeats until there are no more resources left to repair additional lines for the current stage.

E. Max-R-Backward

As an alternative to compute a more accurate solution of the *Max-R* problem, we use *Max-R-Backward*. The algorithm starts by solving a single stage of the problem assuming $R = R_1 + \dots + R_K$ resources are available. The solution of this

Algorithm 3: Max-R-shadow-pricing recovery algorithm.

Data: A set of failed lines $(ij) \in E_p^{B,t}$, A set of demand loads $L_j \in V_p$ and generators $G_i \in V_p$, limit on the tolerable power of each transmission line F_{ij}^{max} , the nodal admittance matrix B , the required resources to repair each line r_{ij}

Result: The recovery schedule of the failed transmission lines $\delta_{(ij),k}$

```
1:  $R = 0$ 
2: for  $k \in \{1, \dots, K\}$  do
3:    $R = R + R_k$ 
4:   while  $\exists (ij) \in E_p^{B,t}$  that  $r_{ij} \leq R$  do
5:     Select an un-repaired line  $(ij)^* = \operatorname{argmax}_{ij} \frac{F_{ij}}{r_{ij}}$ 
6:      $\delta_{(ij),k} = 1$ 
7:      $R = R - r_{(ij)^*}$ 
8:   end while
9: end for
10: return  $\delta_{(ij)^*,k}$ 
```



(a) HVIET.



(b) GARR.

Fig. 4: a) The Italian high-voltage (380 kV) transmission grid (HVIET), and b) its communication network (GARR).

single stage algorithm is added to the set Rep_K , showing the set of edges, which should be repaired up to stage k . Then, the single stage is solved assuming $R = R_1 + \dots + R_{K-1}$ resources are available, which gives the solution set Rep_{K-1} . The repaired lines, which are in the solution set of Rep_K and not in Rep_{K-1} will be added to the repair schedule of stage K . This procedure repeats until the repair schedule of all stages is found.

V. EVALUATION

In this section, we compare our *Min-CFA* cascade prevention approach presented in section III, with a baseline algorithm, which does not include cascade prevention. We also compare the recovery performance of *Max-R-shadow-pricing* and *Max-R-backward* recovery approaches. We use the Italian power grid network shown in Figure 4a consisting of 310 nodes, 113 generators and 97 demand loads. The network has 361 power. For the communication network we use the GARR network, shown in Figure 4b consisting of 39 nodes and 50 edges [4, 5]. We implement our cascade prevention and recovery algorithms in python and used the

Algorithm 4: Max-R-Backward recovery algorithm.

Data: A set of failed lines $(ij) \in E_p^{B,t}$, A set of demand loads $L_j \in V_p$ and generators $G_i \in V_p$, limit on the tolerable power of each transmission line F_{ij}^{max} , the nodal admittance matrix B , the required resources to repair each line r_{ij}

Result: The recovery schedule of the failed transmission lines $\delta_{(ij),k}$

```
1: solve DC power flow model to find  $F_{ij}$ , assuming all
   lines are working
2:  $Rep_K = E_p^{B,t}$ 
3: for  $k = K - 1$  downto  $k = 1$  do
4:    $R = \sum_{m=1}^k R_m$ 
5:    $Rep_k = Rep_{k+1}$ 
6:   while  $\sum_{(ij) \in Rep_{k+1}} r_{ij} > R$  do
7:     Select a line with minimum flow per cost
8:      $(ij)^* = \operatorname{argmin}_{ij} \frac{F_{ij}}{r_{ij}}$ 
9:      $\delta_{(ij),k+1} = 1$ 
10:     $Rep_k = Rep_{k+1} \setminus (ij)^*$ 
11:  end while
12: solve DC power flow model to find  $F_{ij}$ , assuming
    $(ij) \in Rep_k$  are working.
13: end for
14: return  $\delta_{(ij)^*,k}$ 
```

Gurobi optimization toolkit, on a 120-core, 2.5 GHz, 4TB RAM cluster [32].

In the following experiments, we compare the total cost of failure and delivered power in cases where 1) there is no cascade prevention, 2) the cascade prevention can only turn a load on/off and 3) where we can reduce the load's demand continuously. For each scenario, we randomize the results running 10 different trials, where we vary the random selection of failed transmission lines.

A. Preventing the cascade (Min-CFA)

In the first set of simulations, we compare the performance of the *Min-CFA* cascade prevention algorithm with respect to the total cost and total delivered demand power. Similar to [24], we assume all loads have the same priority and give a high penalty for not being able to satisfy the demand. We assume the weighted cost of decreasing power of load L_j is 100, i.e. $w_{L_j} = 100 \quad \forall L_j \in L$, while the normalized weighted cost of generators is 1, i.e. $w_{G_i} = 1 \quad \forall G_i \in G$.

In the first set of simulations we disrupt 60% of the transmission lines and run *Min-CFA* to find the optimal flow assignment. The *Min-CFA* algorithm finds the optimal solution with 54.39% of the demand satisfied. On the other hand, if we do not run a cascade prevention algorithm, the failed transmission lines lead to more lines failing and this process continues until the whole system fails. Figure 5a shows the total delivered power during different time steps of the

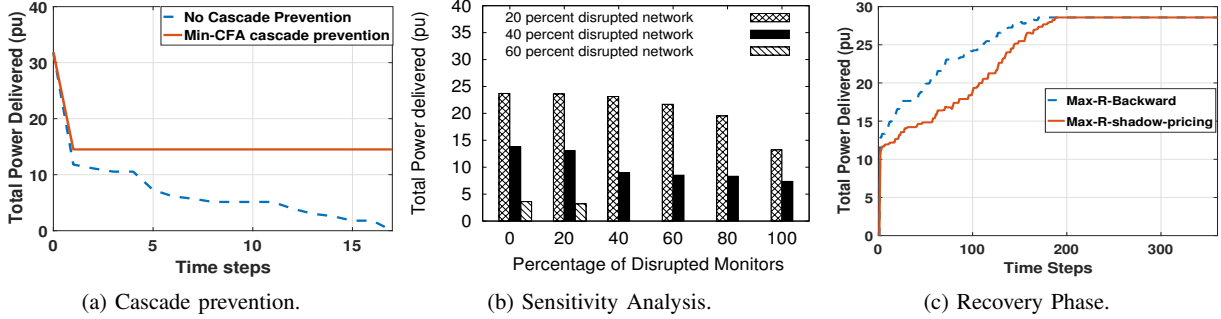


Fig. 5: a) Total delivered power (pu) during time when we use Min-CFA cascade prevention algorithm and without any cascade prevention in Italian power grid network, b) Total delivered power (pu) versus the percentage of failures on the monitoring nodes, c) Total delivered power (pu) flow over time for Max-R-Backward and Max-R-shadow-pricing in the Italian power grid.

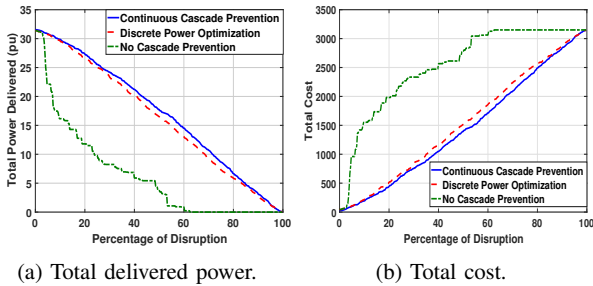


Fig. 6: a) Total delivered power (pu), and b) total cost versus the percentage of disruption in the Italian power grid.

algorithms with Min-CFA cascade prevention and without it. As shown Min-CFA can save 45% of total power that could be delivered if the network was not disrupted.

In the next set of simulations, we use a continuous cascade prevention, meaning that $P_{L_j}^t$ in equation 4 can be decreased continuously. Then, we consider a discrete cascade prevention scenario, where each load's demand power should be satisfied or turned off; and finally, we consider a scenario, where there is no monitoring technique to reschedule the power flow or avoid the cascade and the failed transmission lines can trigger multiple cascade. Figures 6b and 6a show the simulation results for the 3 cases versus the percentage of disrupted network. As shown, the continuous cascade prevention approach saves more power compared to the discrete power optimization and when there is no information from the monitoring network no power can be delivered when 60% of the power lines are disrupted.

B. Sensitivity Analysis

In this section, we investigate the impact of incomplete knowledge about the exact location of failures. We consider a destroyed graph and make $x\%$ of the network uncontrollable (where we lose monitoring information). We then run the detection algorithm to remove the grey cycle-free edges. Next, we assume that the total grey area within the cycle is

working (controller's belief about the grey area within the cycle which might not be correct), and then we run Min-CFA algorithm (to adjust the powers). Figure 5b shows the simulation results of this experiments. It is shown that when 20% of the network get disrupted, the total delivered power can drop by 44.20% when all the monitors get disrupted. Assuming the maximum unitary profit of 26.6 €/MW according to [33], the total profit loss, due to uncertainty can be as high as 209076 € = $10.48pu \times 750MW/pu \times 26.6 €/MW$ which could be avoided using a detection algorithm and a cascade prevention approach.

C. Recovery phase (Max-R)

In the next set of experiments, we compare the recovery performance of the proposed heuristics (Max-R-shadow-pricing and Max-R-Backward). Figure 5c shows the total delivered power flow over different steps of the algorithm when using the two algorithms. As shown, the shadow-pricing algorithm does not consider the correlation between different steps of the recovery approach and tries to maximize the added flow at each iteration step. On the other hand, the backward algorithm solves the problem using all repair resources in the beginning and removes the repair edges with less profit ($F_{ij}/r_{(ij)}$) from the schedule of previous stage until all repair schedules are determined. Therefore, Max-R-Backward performs better than the Max-R-shadow-pricing approach with larger total area behind the curve in Figure 5c.

VI. CONCLUSION

This paper studies the combined impact of large-scale failures on a power grid and its monitoring network. We propose a 2-phase mitigation strategy that 1) avoids further cascade while the system is in the transient state and 2) provide a maximum power flow recovery approach. We show that the maximum flow recovery problem (Max-R) is NP-Hard and intractable. Due to high complexity of the recovery problem, we propose two heuristic approaches (i) a shadow-pricing heuristic and (ii) a backward algorithm. It is shown that since

the shadow-pricing heuristic does not consider the combined impact of repaired component, it performs poorly compared to the backward algorithm.

We also propose a consistent failure set (CFS) algorithm to cope with the uncertainty due to the failure of the dependent communication network that provides the information about the status of power lines being overloaded. We show that CFS can find all failure sets given the information from the previous state of the network before the disruption and the incomplete information about the status of the lines. Our recovery approach and detection mechanism with incomplete information due to failure of the monitoring network is one of the first steps towards understanding the cascaded failures under uncertainty and opens up the area of power grid reliability approaches under incomplete or noisy information.

ACKNOWLEDGEMENT

We thank, Nilanjan Ray Chaudhuri and Vittorio Rosato for their feedback on earlier drafts of this paper and providing the power grid datasets. This research is supported in part by the Defense Threat Reduction Agency under Grant HDTRA1-10-1-0085 and in part by the U.S. Army Research Laboratory under Agreement W911NF-14-0610.

REFERENCES

- [1] D. Bienstock. *Electrical Transmission System Cascades and Vulnerability: An Operations Research Viewpoint*. SIAM, 2015.
- [2] S. Abraham et al. *Final report on the august 14, 2003 blackout in the united states and canada: Causes and recommendations*. US-Canada Power System Outage Task Force, 2004.
- [3] M. Parandehgheibi, E. Modiano, and D. Hay. Mitigating cascading failures in interdependent power grids and communication networks. In *IEEE SmartGridComm*, 2014.
- [4] V. Rosato, L. Issacharoff, F. Tiriticco, S. Meloni, S. Porcellini, and R. Setola. Modelling interdependent infrastructures using interacting dynamical models. *International Journal of Critical Infrastructures*, 2008.
- [5] V. Rosato, L. Issacharoff, G. Gianese, and S. Bologna. Influence of the topology on the power flux of the italian high-voltage electrical network. *arXiv preprint arXiv:0909.1664*, 2009.
- [6] S. Ciavarella, N. Bartolini, H. Khamfroush, and T. La Porta. Progressive damage assessment and network recovery after massive failures. In *INFOCOM*. IEEE, 2017.
- [7] N. Bartolini, S. Ciavarella, T. F. La Porta, and S. Silvestri. Network recovery after massive failures. In *Dependable Systems and Networks (DSN)*, 2016.
- [8] K. Al Sabeh, M. Tornatore, and F. Dikbiyik. Progressive network recovery in optical core networks. In *7th International Workshop on Reliable Networks Design and Modeling (RNDM)*. IEEE, 2015.
- [9] D. Z. Tootaghaj, H. Khamfroush, N. Bartolini, S. Ciavarella, S. Hayes, and T. La Porta. Network recovery from massive failures under uncertain knowledge of damages. In *IFIP Networking*, 2017.
- [10] J. Wang, C. Qiao, and H. Yu. On progressive network recovery after a major disruption. In *IEEE INFOCOM*, 2011.
- [11] M. Parandehgheibi and E. Modiano. Robustness of interdependent networks: The case of communication networks and the power grid. In *IEEE GLOBECOM*, 2013.
- [12] C. D. Brummitt, R. M. DSouza, and EA Leicht. Suppressing cascades of load in interdependent networks. *Proceedings of the National Academy of Sciences*, 109(12):E680–E689, 2012.
- [13] A. Sturaro, S. Silvestri, M. Conti, and S. K. Das. Towards a realistic model for failure propagation in interdependent networks. In *ICNC*. IEEE, 2016.
- [14] R. Parshani, S. V. Buldyrev, and S. Havlin. Interdependent networks: reducing the coupling strength leads to a change from a first to second order percolation transition. *Physical review letters*, 105(4):048701, 2010.
- [15] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin. Catastrophic cascade of failures in interdependent networks. *Nature*, 2010.
- [16] H. Khamfroush, N. Bartolini, T. La Porta, A. Swami, and J. Dillman. On propagation of phenomena in interdependent networks. *IEEE Transactions on network science and engineering*, 2016.
- [17] L. M. Shekhtman, M. M. Danziger, and S. Havlin. Recent advances on failure and recovery in networks of networks. *Chaos, Solitons & Fractals*, 2016.
- [18] E. Chatziafratis, Y. Zhang, and O. Yagan. On the robustness of power systems: optimal load-capacity distributions and hardness of attacking. In *ITA*, 2016.
- [19] A. Bernstein, D. Bienstock, D. Hay, M. Uzunoglu, and G. Zussman. Power grid vulnerability to geographically correlated failures analysis and control implications. In *INFOCOM*, 2014.
- [20] D. Z. Tootaghaj, F. Farhat, M. R. Pakravan, and M. R. Aref. Game-theoretic approach to mitigate packet dropping in wireless ad-hoc networks. In *Consumer Communications and Networking Conference (CCNC)*. IEEE, 2011.
- [21] D. Z. Tootaghaj, F. Farhat, M. R. Pakravan, and M. R. Aref. Risk of attack coefficient effect on availability of ad-hoc networks. In *Consumer Communications and Networking Conference (CCNC)*. IEEE, 2011.
- [22] A. Das, J. Banerjee, and A. Sen. Root cause analysis of failures in interdependent power-communication networks. In *MILCOM*. IEEE, 2014.
- [23] A. Das, C. Zhou, J. Banerjee, A. Sen, and L. Greenwald. On the smallest pseudo target set identification problem for targeted attack on interdependent power-communication networks. In *MILCOM*. IEEE, 2015.
- [24] B. A. Carreras, V. E. Lynch, I. Dobson, and D. E. Newman. Critical points and transitions in an electric power transmission model for cascading failure blackouts. *Chaos: An interdisciplinary journal of nonlinear science*, 2002.
- [25] M. J. Eppstein and P. DH Hines. A random chemistry algorithm for identifying collections of multiple contingencies that initiate cascading failure. *IEEE Transactions on Power Systems*, 2012.
- [26] N. Kayastha, D. Niyato, E. Hossain, and Z. Han. Smart grid sensor data collection, communication, and networking: a tutorial. *Wireless communications and mobile computing*, 2014.
- [27] D. Pendarakis, N. Shrivastava, Z. Liu, and R. Ambrosio. Information aggregation and optimized actuation in sensor networks: enabling smart electrical grids. In *INFOCOM*. IEEE, 2007.
- [28] S. Sridhar, A. Hahn, and M. Govindarasu. Cyber-physical system security for the electric power grid. *Proceedings of the IEEE*, 2012.
- [29] S. Soltan, M. Yannakakis, and G. Zussman. Joint cyber and physical attacks on power grids: Graph theoretical approaches for information recovery. In *SIGMETRICS*. ACM, 2015.
- [30] J. G. Oxley. *Matroid theory*, volume 3. Oxford University Press, USA, 2006.
- [31] A. M. Kettner and M. Paolone. On the properties of the power systems nodal admittance matrix. *arXiv preprint arXiv:1702.07235*, 2017.
- [32] Gurobi optimization, inc. gurobi optimizer reference manual. <http://www.gurobi.com/>, accessed in, 2012.
- [33] M. Dicatoro, A. Minoia, R. Sbrizzai, and M. Trovato. A simulation tool for studying the day-ahead energy market: the case of italy. In *Power Engineering Society Winter Meeting*. IEEE, 2002.