Price Modification Attack and Protection Scheme in Smart Grid

Subhankar Mishra, Xiang Li, Tianyi Pan, Alan Kuhnle, My T. Thai, *Member, IEEE*, and Jungtaek Seo

***Abstract*—Smart grid addresses the problem of existing power grid's increasing complexity, growing demand, and requirement for greater reliability through two-way communication and automated residential load control among others. These features also make the smart grid a target for a number of cyber attacks. In this paper, we study the problem of price modification attack (PMA) through fabrication of price messages, which induces changes in load profiles of individual users and eventually causes major alteration in the load profile of the entire network. Combining with cascading failure, it ends up with a highly damaging attack. We prove that the problem is nondeterministic polynomial-time-complete and provide its inapproximability. We devise two approaches for the problem, the former deals with maximizing failure of lines with the given resource and then extending the effect with cascading failure, while the later takes cascading potential into account while choosing the lines to fail. We formulate new protection strategy against PMA and this includes two new algorithms, namely bi-level programming with new branching method and an effective heuristic to improve the running time. Empirical results on both IEEE bus data and real network help us evaluate our approaches under various settings of grid parameters.**

***Index Terms*—Smart grids, power system security, optimization.**

I. Introduction

ONE OF the most critical infrastructures of the present human civilization is the power grid. Protection of these critical infrastructures is priority for the governments. However some recent blackouts in the power grid along with the normal interference in day-to-day activities result in losses up to billions of dollars. For example, a huge blackout was triggered in the power grid of the United States and Canada, resulting in power cut for over 50 million people in August 2003 [1]. To enhance the reliability and efficiency of these existing power grid systems, smart grid [2] was proposed which included many advancements such as smart metering, two-way communication capabilities, distributed intelligence and automation of home systems. They also end up in creating opportunities for the attackers by opening up new vulnerabilities in power infrastructures. The basic sectors of the power system, i.e., generation, distribution and control, and consumption are open to a wide range of damaging cyber attacks [3] and an cyberintrusion [4] attempt may target any sector. Attack on the generation and distribution sectors need much more sophisticated and significant resources as compared to the consumption sector. Therefore, the consumption sector requires a much more attention on the counter-measures of various cyber-attacks.

Among all attacks towards the consumption sector, cyber-intrusion attacks [5] that fabricate price signals or messages through the Internet become very crucial due to the following reasons: 1) with the help of automated and distributed software intruding agents, this attack becomes much easier to launch. Furthermore, because of the load control and automated energy consumption scheduling (ECS) features of the smart grid, these attacks can be very effective. Given the price information and energy consumption needs of the users, ECS units accordingly schedule the timing and amount of energy consumption for each household appliance. Decisions are primarily based on minimizing the cost of energy. As the price information is obtained through the Internet, false price injection can trigger potential load altering attacks exposing the automated residential load control. 2) More importantly, this class of cyber attacks will eventually increase the load at most crucial locations in the grid causing circuit overflow or other malfunctioning that can immediately bring down the grid or cause significant damage to the power transmission and user equipments. This combined with the cascading failure can lead to major blackouts and collapse of the entire system. Unfortunately the counter measure against the price alteration attack is indeed very challenging, given the dependence on the Internet and its vulnerabilities and also the numerous private distributors.

Due to the above challenges, in this paper, we first attempt to identify the set of most vulnerable nodes along with their respective alteration in the rates, which when attacked lead to maximum number of line failures in the system. We also take into consideration the impact of cascading failure after the initial failure of lines. With this we give two perspectives of approaching to the problem. The first one deals

Manuscript received June 28, 2015; revised October 16, 2015; accepted November 21, 2015. Date of publication January 7, 2016; date of current version June 19, 2017. This work was supported in part by the National Science Foundation Career Award under Grant 0953284, in part by the Defense Threat Reduction Agency under Grant HDTRA1-14-1-0055, and in part by the Korea National Security Research Institute. Paper no. TSG-00743-2015. *(S. Mishra and X. Li contributed equally to this work.)*

S. Mishra, X. Li, T. Pan, and A. Kuhnle are with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32603 USA (e-mail: mishra@cise.ufl.edu; xixiang@cise.ufl.edu; tianyi@cise.ufl.edu; kuhnle@cise.ufl.edu).

M. T. Thai is with the Division of Algorithms and Technologies for Networks Analysis, Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam, and also with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32603 USA (e-mail: mythai@cise.ufl.com).

J. Seo is with the Attached Institute of Electronics and Telecommunications Research Institute, Daejeon 305-600, Korea (e-mail: seojt@ensec.re.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2015.2509945

with maximizing failure of lines initially and then extending the effect with cascading failure while the second one takes cascading potential into account while choosing the lines to fail.

Although protection of smart grid has been studied previously [8]–[10], they only focus on the single stage protection or do not counter attacks targeted at consumption sector. To overcome these shortcomings, we devise the protection strategy from the protector's perspective modeled as a one leader (protector), one follower (attacker) Stackelberg game. Both the attacker and protector can choose their own optimal strategy to maximize their respective gains. Unfortunately this game structure is very challenging, given the inner problem is nondeterministic polynomial-time (NP)-Hard and the existing solutions for Stackelberg game may not be applicable. The protection scheme also gets challenging due to the cascading nature of the failure. To tackle the above challenges, we have proposed two effective techniques called branch merging and filtering and an heuristic algorithm to improve the running time.

Our contributions are summarized as follows:

- We define a new problem of Price Modification Attack (PMA) and prove its hardness and inapproximability of $O(m^{1-\eta})$, m being the number of edges and $\eta > 0$.
- We propose two approaches to the problem, namely MaxL and CasL. (1) MaxL tries to fail as many lines as possible and then calculate the total number of failures using the cascading effect. (2) In CasL we rank the lines on the basis of their cascading potential and then fail those lines one by one till we exhaust the resource.
- We propose the protection schemes Two Stage Branching Algorithm (TSBA) and Protect Most Critical Nodes Algorithm (PMCNA) to protect the smart grid against PMA.
- We experimentally evaluate our proposed algorithms in various settings, from which we infer many insights on the power network behavior to price modification attack.

Section II describes the smart grid structure and model and the problem definition. The complexity and inapproximability proofs are given in Section III. We next provide the attacking methods for causing maximum line failures in Section IV. Protection schemes are discussed in Section V. Performance evaluation of the proposed algorithms is presented in Section VI. The related works are discussed in Section VII, which is then followed by conclusion in Section VIII.

NOMENCLATURE

$G(V, E)$	Power grid network with vertices and edges.
P	Set of power generation nodes.
D	Set of demand nodes.
p_i	Power generation output of i.
d_i	Load demand of i.
B_i	Billing profile/cost of i.
r_i	Rate of electricity at i.
ρ_i	Maximum rate change of i.
c_i	Cost of attacking i.
k_i	Sensitivity of i towards billing.
f_{ij}	Power flow between i and j.
x_{ij}	Reactance of edge ij.
u_{ij}	Capacity of edge ij.
R_A	Attacking resource constraint.
R_P	Protection resource constraint.
γ_i	Maximum protection resource for i.
y_i	Protection plan for i.
$\mathcal{A}$	Set of single-link attacking plans.

II. NETWORK MODEL AND PROBLEM DEFINITION

In this section, we describe the network model of the Smart Grid and the associated DC power flow model to understand the power flow dynamics in the power grid.

A. Smart Grid

We consider a smart grid represented by a directed graph $G(V, E)$, where each node in V represents either the power generation stations, intermediate power transformer stations, or the consumption sector (houses, industries and data centers etc.) and edges E represent the transmission power lines between nodes. The set of nodes V includes power generation nodes $P \subset V$, other intermediate nodes $O \subset V$ with no power generation or demand and consumers $D \subset V$.

Each node $i \in P$ has power generation output given by p_i. Every user has a demand d_i and a billing profile B_i. Every user/node i receives the electricity rate r_i from the Internet which we assume to be accessible to the attacker to manipulate or alter. A certain portion of the rate is allowed to change for each user given the baseline constraints set up or hard coded by electricity distribution companies. Hence, we have the maximum rate change (MRC) ρ_i, which represents the maximum change in the rate that the attacker is allowed. Note that we use price modification and rate alteration interchangeably. The attacker can choose what percentage of the MRC it wants to change denoted by z_i which lies between 0 and 1. Given the automated demand side management, one of the "smart" features of the smart grid, the rate change causes automated increase in the use of the demand for the same household or the company, such as starting up the laundry, more frequent use of the heating/cooling devices, etc. There is a cost associated with the change done by the attacker. Here we consider a linear cost function $c(.)$ for simplicity. The attacker is restricted by the maximum resource R_A, i.e., the attacker is bounded by a given total cost to alter the rates at various positions in the grid.

As explained in previous section about automation of smart grids, and considering the sensitivity of the users to the billing, the total bill is given by $(1 + k_i)B_i$ where B_i represents the user's targeted billing amount and k_i represents the sensitivity of the user towards billing amount. $k_i = 0$ indicates that the user is not willing to pay anymore than the targeted billing amount. For simplification we allow the k_i with a maximum value of 1. The line (i, j) fails when the power flow through the transmission line goes over its capacity.

Every transmission line is a directed edge $(i, j) \in E$ connecting node i to node j, which has a maximum capacity u_{ij}. The power flow is given by f_{ij}. When the power through a

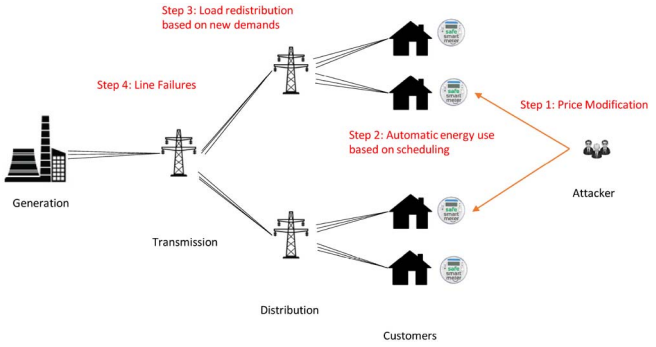


Fig. 1. Attacking Model : Step 1: Attacker changes price in the smart meters. Step 2: The automatic scheduling algorithm realizing that the price is low, starts using electricity for appliances that are scheduled only for the low price time. Step 3: These new increased demands cause load redistribution in the transmission lines. Step 4: With a successful number of customers under attack, transmission lines are overloaded and start failing.

transmission line goes over the capacity, it breaks due to thermal heating. As we try to balance the total power generation to the total demand of the consumers, the erratic demands can lead to power flow through a transmission line increasing beyond the capacity of the line; which leads to the failure of the line.

B. Attacking Model

In smart grid, the basic purpose is to create an automated, widely distributed energy delivery network that uses smart meters to facilitate two-way flows of information and electricity between energy consumers and providers. This transformation enables greater support for demand response and provides more flexibility in demand shaping through time-dependent pricing (TDP).

In a smart grid infrastructure, electricity providers can send pricing information from their pricing databases to the Energy Consumption Controller (ECC) unit located at the consumer's smart meters, as shown in Fig. 1. The ECC can monitor and control a consumer's energy consumption by scheduling device activities at periods of lower prices. An increasing number of devices, such as vacuum cleaners (e.g., Roomba), smart washing machines (e.g., Miele), and smart ovens (e.g., LG Thinq), are becoming more intelligent and can be scheduled, either manually or automatically by the ECC, to switch on or off depending on the prices at different times of the day. Such innovations further enable electricity providers to effectively use dynamic pricing to match their cost to revenues by flattening out peak demand and achieving better resource utilization. Enterprise markets offer additional opportunities in addition to consumer markets.

Existing literature already considers user interaction with the smart grid and uses this interaction to determine electricity prices for the electricity providers [6]. The studies include modeling user interaction to dynamic pricing and real trial studies, tracking the schedule of the consumer's electricity consumption to quantitatively predict the future use, using a feedback loop between the distributer and the consumer to determine real time pricing in order to provide stability to pricing scheme. They also include the auction system for the electricity distributors giving rise to dynamic pricing from the providers and responses from the consumers. The paper [7] deals with joint scheduling of consumers with shared information in order to reduce the peak load times and reducing overall cost to consumers.

All the above pricing schemes although charge their consumers giving them the choice of automatic scheduling for handling major devices such as PHEVs, EVs, washing machines etc., as well controlling other major devices. And although this helps in flattening the peak demand, the entire scheduling operation depends on the price that is sent to the smart meter by the distributing company irrespective of the total load on the system. Here the attacker increases the prices on smart meters, by fabricating the price signals from the electricity distributing company. This is followed by the running of major appliances which were otherwise scheduled for the 'real' low price time-line. This leads to increase in demands in consumer section of the smart grid system, thereby causing the massive load shifts and load redistribution in the power grid. The transmission lines under this heavy pressure when get overloaded start heating up and eventually leading to line failures. This attack will keep on distributing the load across the power grid, causing cascading failures leading to catastrophic blackouts.

C. DC Power Flow

The common modeling of the behavior of the power grids is done using a system of non-linear, non-convex equations which describe the physics of AC power flows [15]. The power flow problem constitutes of active and reactive power flow and can be put together in a formulation by using four variables per node [23]. Those variables are active power injection, reactive power injection, voltage magnitude and voltage angle. Active power losses depend on the voltage profile and active power injection pattern, hence they are not known in advance. This and other interdependent variables make the problem non-linear and hence is linearized often. The losses are re-evaluated at each loop depending on all other variables and the solution is calculated using successive linearized loops. Newton-Raphson algorithm is used by the modern power analysis tools. The Newton-Raphson method has a quadratic convergence and computing time only increases linearly with system size.

The power flow problem is simplified by converting the system to linear in order to reduce the running time. To make the linearization feasible, the following assumptions are taken into consideration:

- Phase angle differences are small and hence we can use $sin\theta_{ij} = \theta_{ij}$ and $cos\theta_{ij} = 1$, where θ_{ij} is the phase angle between the voltages at the two nodes i and j.
- Lossless lines; i.e., the resistance of the arcs/lines is negligible.
- Voltage profile is kept flat.

But the above assumptions are not natural. The voltage profile varies with respect to buses and hence is not flat always. Resistance is not always negligible. The impact of resistance increases with the decrease in voltage. Hence the distribution

sub system of the smart grid would not follow this rule and only the high voltage transmission lines can accommodate this condition. All the above assumptions, affect the quality of the solution, and introduce alterations in the accuracy of the solution, thus the DC power flow is less accurate compared to the full, AC power flow solution.

Even though AC power flow is more accurate, the Newton-Raphson algorithm may fail to converge under extreme operating conditions. Even though Newton-Raphson is relatively fast it may be too slow when a large volume of power flow computations is required. For these and other reasons, researchers normally prefer and rely on the linearized or DC power flow approximation. This is solved far more quickly and is proved to be accurate under good operating conditions and thus we have adopted this model for our paper.

We briefly describe the linearized or DC power flow model. In the linearized approximation, we are given a power grid represented by a directed graph G, where:

- Each node $i \in V$ corresponds to either a power generator (i.e., a supply node), or to a load (i.e., a demand node), or to a node that neither generates nor consumes power.
- If node i is a generator, then there are values $0 \leq P_i^{min} \leq P_i^{max}$. If the generator is operated, then its output must be in the range $[P_i^{min}, P_i^{max}]$; if the generator is not operated, then its output is zero. In general, $P_i^{min} > 0$.
- If node i is a demand, then the "nominal" demand is given by D_i^{nom}. The set of demands or demand nodes is denoted by D.
- The edges E represent power/transmission lines. For each line (i, j), two parameters are given, i.e., $x_{ij} > 0$ (the resistance or reactance) and u_{ij} (the capacity).

Now, given a set P of operating generators, the linearized power flow is a solution to the system of constraints given in the following set of the equations. For each edge (i, j), f_{ij} represents the power flow on the edge (transmission line) (i, j). In the case where $f_{ij} < 0$, power is effectively flowing from j to i. Additionally, the phase angle at node i is given by the variable θ_i. Given a node i, $\delta^+(i)(\delta^-(i))$ is the set of lines oriented out of (into) node i. The power flow equations are given below:

$$\sum_{(i,j)\in\delta_i^+} f_{ij} - \sum_{(j,i)\in\delta_i^-} f_{ji} = \begin{cases} p_i & i \in P \\ -d_i & i \in D \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$\theta_i - \theta_j - x_{ij} f_{ij} = 0, \quad \forall (i,j) \in E \quad (2)$$

$$p_i^{min} \leq p_i \leq p_i^{max}, \quad \forall i \in P \quad (3)$$

$$0 \leq d_j \leq d_j^{nom}, \quad \forall j \in D \quad (4)$$

D. Problem Definition

We are now ready to formally define our problem as follows.

Definition 1 (Price Modification Attack (Lines) PMA): Given a smart grid system $G(V, E)$, $P \subset V$ being the set of power generators, and $D \subset V$ being the set of demand nodes, maximum attack resource R_A, and electricity billing price r_i, billing constraints $(1 + k_i)B_i$ and maximum rate change ρ_i, for each demand node i. Compute an attacking strategy $z = \{z_i\}, i \in D, z_i \in [0, 1]$ that alter the rates of those demand nodes, such that the total number of the ***line*** failures is maximized.

III. Complexity

In this section, we first prove the NP-Completeness of PMA. We next study its inapproximability which shows the best approximation ratio that one can ever achieve.

Theorem 1: The price modification attack (PMA) is NP-Complete.

Proof: First, we define the decision version of price modification attack problem.

Definition 2 (Decision Version of PMA (Lines)): Given a system $\{G(N, P), P, D, R_A, r_i, k_i, B_i, c_i, \rho_i\}$ PMA asks whether or not there is an attacking strategy $z = \{z_i\}, i \in D, z_i \in [0, 1]$ that alter the rates of those demand nodes will result in failure of at least m lines.

We first prove price modification attack problem is in NP. Given the set of demands which rates will be altered in the system, we can verify if number of failed lines is greater than m in polynomial time.

To prove the NP-completeness, we reduce from the maximum coverage problem (MC). The decision version of MC is defined as following.

Definition 3 (Decision Version of MC): Given a set $U = \{S_1, S_2, S_3, \ldots, S_n\}$, the space $E = \cup_{S_i \in U} S_i$ and a number k, MC asks whether or not there exists a subset $S' \in U$ such that $|S'| \leq k$ and the number of covered elements $|\cup_{S_i \in S'} S_i|$ is greater than m $(m < |E|)$.

Here we show how to reduce MC to PMA (see Fig. 2). Let (E, U, k) be an instance of MC, and assign one node g as a generator. For each set $s_i \in S$, create a user node u_{s_i}. For each element $e \in E$, create a node n_e, and add edges (g, n_e) to the generator. Now, for each u_{s_i} add edges (n_e, u_{s_i}) iff $e \in S_i$. Set $B_i = |S_i| - 1$, $r_i = 1$, $k_i = 0$. Also, suppose that each node u_{s_i} initially has demand $d_i = |S_i| - 1$, so that the initial flow of edges (n_e, u_{s_i}) will be $1 - \frac{1}{|S_i|}$. Let $\rho_i = \frac{1+\epsilon}{|S_i|+\epsilon}$ and the capacity of lines (n_e, u_{s_i}) to be 1 (all the other lines are assumed not to broken in this attack). So that when we choose to decrease the rate of user node i to its lowest $(r_i - \rho_i)$, the new demand will be $d_i' = |S_i| + \epsilon$. Also, assume that the cost $c_i = 1$ and the budget $R_A = k$. In addition, let the reactance x_{ij} be equal to 1. It is easy to show that this construction takes polynomial time.

Then we will prove if MC has a solution T, $|T| \leq k$ that guarantee a coverage of m elements, PMA has a rate alternation strategy to fail m lines with budget k. Also, if PMA has a solution z to fail m lines, there is a solution for MC to cover m elements.

$\Longrightarrow$ Assume MC has a solution $T(|T| \leq k)$ that cover at least m elements. In PMA, if we choose to alter the rates of all the user nodes corresponding to the sets in T to the lowest, i.e., set $z_i = 1$, $s_i \in T$, then the demand of each selected user nodes will increase to $d_i' = |S_i| + \epsilon$. We keep all the other user nodes unchanged.

So that the flow on edges (n_e, u_{s_i}) will be $1 + \frac{\epsilon}{|S_i|}$ and those edges connected to set nodes in T will be broken. Since there are at least m nodes covered by the sets in T, we are able to fail at least m lines based on this strategy.

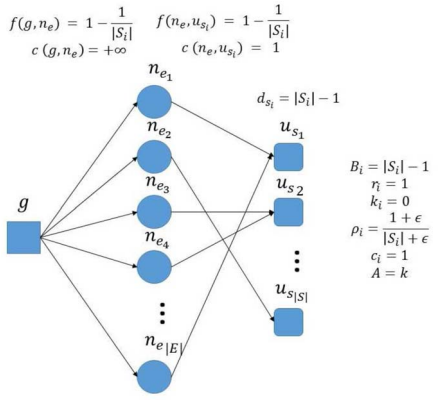


Fig. 2. PMA reduced from MC.

$\Longleftarrow$ Now assume *PMA* has a solution to fail m lines with altering the rates of at most k user nodes. Since all the edges connected to those altered user nodes with $z_i = 1$ must be failed and all the other edges cannot be failed, we know for sure that the m failed lines are all connected to one of the nodes with $z_i = 1$. Thus, if we choose the corresponding sets of those user nodes as a solution of MC, we can cover at least m elements. ■

Theorem 2: There is no $O(m^{1-\eta})$-approximation algorithm for the cascading edge failure problem unless $P = NP$, where m is the number of edges, for any $\eta > 0$.

Proof: Let an instance $(X, \mathcal{S}, k)$ of the set cover problem be given, with $|X| = g$, $|\mathcal{S}| = h$ (See Fig. 3). For each $S \in \mathcal{S}$, create generator vertex u_S, demand vertex v_S and transmission line (u_S, v_S) with capacity $1/(|S| + 1) - \epsilon$ and reactance r_1. For each $x \in X$, create vertices u_x, v_x and a line (u_x, v_x) with capacity

$$\min_{S:x\in S} 1/(|S| + 1) - \epsilon,$$

and reactance r_1. For every set S containing x, add edges (u_S, u_x), (v_x, v_S) with reactance r_1 and capacity 2. Add vertices u_0, v_0 and edges (u_x, u_0), (v_x, v_0) for all $x \in X$ with reactance r_2 and capacity 2. Finally, for $1 \le i \le l$, add extra lines (u_0, u_i), (u_i, v_0), each with capacity $k/l - \epsilon$ and reactance r_2. Choose l so that $k/l < 1/g$. It is possible to choose r_1, r_2 such that for each set S, unless all element lines corresponding to elements of S are broken, only a negligible amount of current from u_S to v_S flows over lines of reactance r_2.

Initially, the demand at each vertex v_S is set to 0, with maximum possible demand equal to 1. We will show that the $2l$ extra lines can be broken with budget k iff there is a set cover of size k.

Suppose there is a set cover SC of size k. For each $S \in SC$, raise the demand of v_S to its maximum value 1. This breaks every element line; hence the extra lines receive flow k/l, which causes them to break. So in total $k + g + 2l$ lines break.

Suppose there is no such cover. The most element lines that a feasible solution could break would be $g - 1$. So the extra lines have flow at most $k/(l+1) < k/l - \epsilon$. Hence they remain intact. Thus, the largest possible number of lines broken would be at most $k + g - 1$.

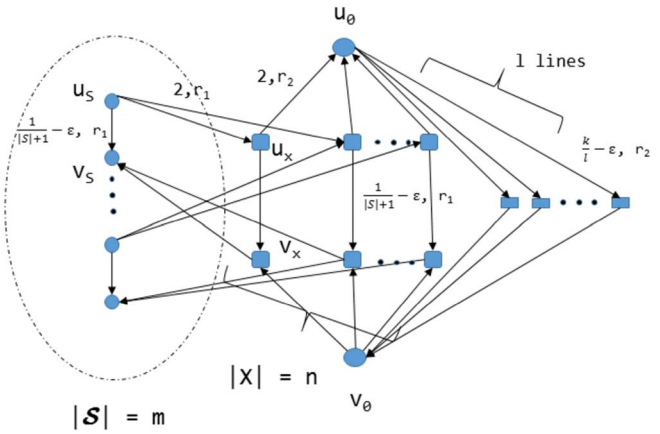


Fig. 3. Inapproximability proof using set cover. The labels on the edges represent (capacity,reactance) respectively.

Now the total number of lines $m = c_1 g + c_2 h + 2l$. There is a constant K such that

$$Km^{1-\eta} < (g + k + 2l)^{1-\eta}.$$

Suppose

$$l \ge \frac{\left(\frac{g+k}{K}\right)^{1/\eta} - g - k}{2}.$$

Suppose we have an $m^{1-\eta}$-approximation algorithm $\mathcal{A}$. If there is a set cover of size k, $\mathcal{A}$ produces a solution breaking at least

$$\frac{g + k + 2l}{m^{1-\eta}} \ge K\frac{g + k + 2l}{(g + k + 2l)^{1-\eta}} = g + k.$$

edges, while if there is no such cover, it produces a solution of size less than $g + k$. ■

IV. Attacking Schemes

In this section, we provide attacking schemes using the price modification attack. Two approaches are considered:

- The first aim is to initially fail as many lines as possible without considering the cascading failure. An integer program is formulated for this case. After initial failure of lines, the cascading effect of the same is taken into account. This is discussed in Section IV-A.
- The second approach is explained in Section IV-B and consists of ranking the edges according to their cascading potential and then failing the highest ranked with the minimum cost. The process is repeated until the maximum resource available to the attacker gets exhausted.

A. Maximizing the Line Failures (MaxL)

We formulate the maximization of number of line failures for the given problem. Let a binary variable y_{ij} indicate the failure of the transmission line $(i, j) \in E$. When $y_{ij} = 1$, the line fails and is 0 otherwise. Our goal is to maximize the total of number of line failures given by $\sum_{e(ij)\in E} y_{ij}$.

The formulation is given as follows:

$$\max \quad \sum_{e(ij)\in E} y_{ij} \tag{5}$$

$$s.t. \quad (1+k_i)B_i \geq d_i \cdot (r_i - z_i \cdot \rho_i) \qquad \forall i \in D \tag{6}$$

$$\sum_{i\in D} c_i(z_i) \leq A \tag{7}$$

$$y_{ij} < 1 + \frac{f_{ij} - u_{ij}}{u_{ij}} \qquad \forall e(ij) \in E \tag{8}$$

$$Eqs.(1) - (4) \tag{9}$$

$$y_{ij} \in \{0, 1\} \qquad \forall e(ij) \in E \tag{10}$$

$$z_i \in [0, 1] \qquad \forall i \in D \tag{11}$$

where constraints (6), (7), (9) represent the billing calculation resulting in the change in the demand values, constraint on the resource available to the attacker and linearized DC power flow equations respectively. Constraint (8) represents the line breaking scenario, that is, the transmission line breaks when the total power flow through the line exceeds the capacity of the corresponding transmission line.

1) Cascading Effect: There are a lot of different models of cascading failures in the complex networks and power grids. In [16], the authors showed that the failure of a single node in the system can cause the load redistribution to the other nodes. This process potentially could cause large global failures because of the continuous redistribution of the load to its neighbors. This has been used in numerous papers [17]–[19] to model cascading failures in various complex networks and power grid systems. However the above model fails to capture the physics of power flow in the power grid systems rather treats them as normal network flow systems. Hence we adopt the cascading failure model from [20] (extension of model in [21]) as described in Alg. 1. Line failures are often associated with the cascading effect. Failure of lines changes the power flows in other lines and results in failure of more lines and often leads to black outs of entire region.

At the steady state, G is connected and total supply is equal to the demand. When there is a failure, some edges/lines are removed from the graph G (i.e., gets disconnected). The total supply and total demand are now adjusted within each component by decreasing the demand and supply at loads and generators respectively. Using the power flow equations, the power flow is recalculated. The new flows may exceed the capacity and as a result, the corresponding lines will become overheated. The outages are modeled by moving average of the power flow $\tilde{f}_{ij}^t$: $\tilde{f}_{ij}^t = \alpha f_{ij} + (1-\alpha)\tilde{f}_{ij}^{t-1}$. The moving average approximates thermal effects, including heating and cooling from prior states to first order.

B. Cascade Potential Ranking (CasL)

In this section, instead of breaking the lines and then using the cascading template to calculate the total number of lines, we use the cascade algorithm to rank the edges according to their cascading potential. Cascading potential of the edge $e*$ is calculated by calculating the numbers of other edges that fail due to load distribution following the edge e^* failure.

Algorithm 1: Cascade Failure Template

Data: Connected Power Grid Network $G(V, E)$
Result: S_1: Lines which failed
S_2: Nodes which failed
while *Network is not stable* **do**
  Adjust the total demand to the total supply within each island.
  Use equations (1)-(4) to calculate power flows in G.
  For all lines computer the moving average $\tilde{f}_{ij}^t = \alpha f_{ij} + (1-\alpha)\tilde{f}_{ij}^t$.
  Remove all lines that have moving average flows greater than the capacity $(\tilde{f}_{ij}^t > (1+\epsilon)u_{ij})$ and add to S_1.
  Add the failed nodes to S_2.
  If no more line fails, then network is stable, break the loop.
Return S_1, S_2

Algorithm 2: Cascade Potential Ranking Lines (CasL) Algorithm

Data: Connected Power Grid Network $G(V, E)$
Result: Total number of failed lines L
Initialize temporary cost $T = 0$
while $T < A$ **do**
  for *each* $e \in E$ **do**
    Remove line e
    Cascade_potential(e) = number of failed edges [by Alg. 1]
  Push edges to E' with decreasing order of Cascade_potential()
  for *each* $e \in E'$ *in the order* **do**
    if $MCB(e) + T < A$ **then**
      e* = e
      break
  $T = T + MCB(e*)$
  $L = L + Cascade\_potential(e*)$
  Remove the failed edges from E
Return L

Next, the objective is to fail the highest ranking edge with minimum cost of rate alteration given by $\sum_{i\in D} c_i(z_i)$. The rest of the constraints follow the same reasoning as the maximization of lines failure except constraint (14) which represent the failure of the edge (e^*) with highest cascade potential. The integer program for calculating the minimum cost for breaking the edge e^* is given by $MCB(e^*)$. The above process is repeated till the attacker runs out of the maximum resource A. Algorithm 2 states the pseudocode for the CasL algorithm.

Minimum cost to break e* [MCB(e*)]

$$\min \quad \sum_{i\in D} c_i(z_i) \tag{12}$$

$$s.t. \quad (1+k_i)B_i \geq d_i \cdot (r_i - z_i \cdot \rho_i) \qquad \forall i \in D \tag{13}$$

$$f_{ij} > u_{ij} \qquad ij = e^* \tag{14}$$

$$Eqs.(1) - (4) \tag{15}$$

$$z_i \in [0, 1] \qquad \forall i \in D \tag{16}$$

V. Protection Schemes

In this section, we turn our attention to the load protection scheme to counter PMA. Then we formulate LPuRA as a bi-level Mixed Integer Program and propose an exact algorithm, *Two Stage Branching Algorithm* (TSBA), to obtain optimal solution of LPuRA. Moreover, since TSBA can be time consuming, we propose a fast heuristic algorithm, *Protect Most Critical Nodes Algorithm* (PMCNA) as an alternative solution.

A. Load Protection Under Rate Alteration Problem

To minimize the impact of DA to the smart grids, we consider Load Protection under Demand Attack problem (LPuRA). In LPuRA, the protector assigns protection resources to demand nodes to make them more difficult (costly) to be attacked. Each demand node $i \in D$ can only accept limited protection resource, upper bounded by γ_i. The protection plan is denoted as $\mathbf{y} = \{y_1, y_2, \ldots, y_{|D|}\}$ where $y_i \in [0, 1], i \in D$. The cost to attack node i increases by $\gamma_i y_i$ when the protection plan is executed. The cost to protect a demand node is denoted by the function $c_i'(y_i)$. The maximum protection resource is denoted as R_P.

The definition to the protection problem is provided as follows.

Definition 4 (Load Protection Under Rate Alteration (LPuRA)): Given a smart grid network $G(N, E)$ with a set of generators P, a set of demand nodes D, the problem asks us to compute a protection plan $\boldsymbol{y}$ under protection resource constraint R_P to protect D in order to minimize the total number of failures after Demand Attack.

B. Bi-Level MIP Formulation for LPuRA

The formulation of LPuRA is as follows.

$$\min \ g(\mathbf{y}) \tag{17}$$

$$s.t. \quad \sum_{i \in D} c_i^{'}(y_i) \leq R_P \tag{18}$$

$$y_i \in [0, 1] \qquad \forall i \in D \tag{19}$$

$$g(\mathbf{y}) = \max \sum_{(i,j) \in E} w_{ij} \tag{20}$$

$$s.t. \quad d_i \leq d_i^0 + \rho_i z_i \qquad \forall i \in D \tag{21}$$

$$\sum_{i \in D} (c_i(z_i) + \gamma_i y_i x_i) \leq R_A \tag{22}$$

$$x_i \leq M z_i, \qquad \forall i \in D \tag{23}$$

$$w_{ij} < \frac{f_{ij}}{u_{ij}} \qquad \forall (i,j) \in E \tag{24}$$

$$Eqs.\ (1) - (3) \tag{25}$$

$$d_j \geq 0, \qquad \forall j \in D \tag{26}$$

$$w_{ij} \in \{0, 1\}, \qquad \forall (i,j) \in E \tag{27}$$

$$z_i \in [0, 1], \qquad \forall i \in D \tag{28}$$

$$x_i \in \{0, 1\}, \qquad \forall i \in D \tag{29}$$

$$f_{ij} \geq 0, \qquad \forall (i,j) \in E \tag{30}$$

The objectives of the protector and the attacker are defined in (17) and (20) respectively, both focus on total number of line failures. It is assumed that a line fails if and only if its flow exceeds capacity. To model line failure, a binary variable w_{ij} is introduced for each line $(i, j) \in E$, which value is 1 when line (i, j) failed and 0 otherwise. When the flow exceeds capacity on line (i, j), the rhs of constraint (24) is greater than 1 and w_{ij} can reach 1. Otherwise, w_{ij} is always 0, based on constraint (27). Objective function (20) ensures that w_{ij} can reach 1 whenever possible. Therefore, constraints (24), (27) together with objective (20) correctly define the mechanism of line failure.

The resource constraints are (18) and (22) for protector and attacker, respectively. Together with constraint (23), the attacking resource constraint also defines the protection mechanism. The attacker is forced to pay an extra cost $\gamma_i y_i$ in order to launch an attack to protected node i. Constraint (21) ensures that the demand of node i can be higher with more assigned attacking resource. The flow equations are described in constraint (25).

Notice that such an MIP cannot depict the cascading failure in smart grids. Also, it is extremely difficult to be solved directly. However, it casts light on the core of the problem and serves as the base of the following algorithms, which do address the cascading failure impacts.

C. Two Stage Branching Algorithm (TSBA)

In a big picture, TSBA works in two stages. In the first stage, TSBA uses branching technique to find candidate feasible solutions for the attacking problem. In the second stage, a protector's problem is solved to make as many of the attacking problem's candidate solution infeasible as possible. We will discuss the two stages sequentially as follows.

1) Stage 1: Solution to the Attacking Problem: For the attacking problem, instead of solving the MIP directly, the candidate attacking plans (lists of targeted lines) are populated using branching method and the best feasible one is chosen. The feasibility of any solution is based on whether the given set of lines can be failed within the budget. The quality of any solution is based on how many lines the attacker can make them fail. To accurately cast the nature of power grid network, the effect of cascading failure is considered, as described in [21]. Given that a set of edges has already failed in the attack, the algorithm proposed by Bernstein *et al.* in [20] can be used to calculate the total number of failed lines. Notice that the cascading failure is extremely hard to be integrated into mathematical formulation, while TSBA can taken it into consideration. In the remaining of this part, the structure of the branching scheme and the feasibility measures are discussed sequentially.

The branching is done using a tree structure. Each node of the branching tree is a fixed solution to the attacking problem. The root node is a null solution which attacks no nodes. In the branching process, a child node will attack one additional link compared to its parent node. The first stage of TSBA starts from the root node and set it as unexplored. For each unexplored node, first its feasibility is checked. If it is feasible, it will be branched and all its possible children nodes are created. The number of cascading line failures based on this solution is also calculated to obtain the quality of the solution. If it is not feasible, the branching on that node will be stopped. Once there does not exist any unexplored nodes, the feasible solution with the best quality is selected as output. Since all feasible solutions are considered, the output solution must be optimal.

The feasibility of a solution is measured by a slightly modified model of the attacking problem. In which only attacking of the targeted lines in the solution is considered. If the model cannot fail those lines under resource

constraint (33), such a solution is infeasible, otherwise it is feasible. The flows of the targeted lines are forced to exceed the capacity by constraint (34). The formulation is as follows.

$$\min \quad \sum_{i\in D} c_i(z_i) \tag{31}$$

$$s.t. \quad d_i \le d_i^0 + \rho_i z_i, \qquad \forall i \in D \tag{32}$$

$$\sum_{i\in D} c_i(z_i) \le R_A \tag{33}$$

$$f_{ij} > u_{ij}, \qquad (i,j) \in E, e_{ij} = 1 \tag{34}$$

$$Eqs.\ (1) - (3) \tag{35}$$

$$d_j \ge 0, \qquad \forall j \in D \tag{36}$$

$$z_i \in [0,1], \qquad \forall i \in D \tag{37}$$

$$f_{ij} \ge 0, \qquad \forall (i,j) \in E \tag{38}$$

Due to the nature of the algorithm, the size of the branching tree will become larger once the problem scales up. Therefore, we propose the following two pruning methods to optimize the performance of the branching process.

a) Branch merging: Assume the current targeted list contains one line. Since the specific is made to fail by letting the whole network overload, it is highly possible that some other lines are also failed. If those lines are neglected, different branches will be created for them under the node of the current attacking plan. However, it is a waste of time and space since it can be known that those lines will fail for sure. Therefore, there is no need to attack them again and those unnecessary branches can be merged to the current plan.

b) Filtering in branching: Sometimes attacking one line can make attacking another one become unnecessary. For instance, 2 has 1 as its sole incoming and the node shared by two lines is not a generator. If 1 is chosen to be attacked, attacking 2 or not will not impact the cascading failure and the final optimization result since it is already disconnected. Then there is no need to waste a branch on this kind of line. Checks are performed before adding attacking plans to the tree to avoid such issues.

2) Stage 2: Integration of Protection Strategy: The general idea of the protection strategy is to assign protection resources to make the best attacking plans infeasible. We will first explain how to protect a set of attacking plans and then describe the method to get the best set of attacking plans to protect.

The feasibility of protection against a set of attacking plans is determined by solving a linear system. The linear system includes constraints (22)-(23) and (32)-(38) for each attacking plan. It also has a single protection resource constraint (18). When it is possible to find a solution to this system, the set of attacking plans can be protected with a sole protection strategy. Otherwise, the protection resource assigned is not enough to protect all of the attacking plans in the set.

Define Q as the set of all feasible attacking plans, descendingly sorted based on number of failed lines resulted from each plan and Q_k as the top k attacking plans in set Q. Then the problem to find the best set of attacking plans to protect can be stated as: find the largest number k^* that Q_{k^*} can be protected within resource limit L. Notice that for any number k', when $Q_{k'}$ can be protected based on the above formulas, $k^* \ge k'$. Otherwise, $k^* < k'$. Also, k^* should be between 1 and $|Q|$. So that binary search can be utilized to find k^*. When the number k^* is found, the top k^* attacking plans can be protected and the plan ranked at $k^* + 1$ remains unprotected. Also, k^* is the final solution to the bi-level problem. Since no solution can protect top $\tilde{k}$ plans with $\tilde{k} > k^*$ based on the structure of TSBA, the solution returned is optimal to LPuRA. The detail of TSBA is described in Alg. 3.

Algorithm 3: Two-Stage Branching Algorithm (TSBA)

Data: Connected Power Grid Network G(N,E), P, D
Result: Protection strategy $\boldsymbol{y} = \{y_i\}, i \in D$, Total number of failed lines F
Initialize branching tree.
Initialize root node as $\boldsymbol{e} = \{e_i = 0\}, i \in D$
Initialize queue of active nodes.
ActiveNodes.Enqueue(*root*)
Initialize list of feasible nodes *FeasibleNodes*.
while *ActiveNodes is not empty* **do**
    Node currentNode = *ActiveNodes.Dequeue*()
    Check feasibility of the solution in current node by solving the attacker's problem.
    if *feasible* **then**
        Populate all possible child nodes of current node.
        Add all child nodes to *ActiveNodes*
        FeasibleNodes.Add(*currentNode*)
        Get number of failed lines for the current node.
Sort all feasible nodes descendingly based on number of failed lines.
Use binary search to find the best set of attacking plans that can be protected Q_{k^*}
$\boldsymbol{y}$ = solution to the protection problem of Q_{k^*}
F = number of failed lines in the node ranked $k^* + 1$
Return $\boldsymbol{y}, F$

D. Protect Most Critical Nodes Algorithm (PMCNA)

Due to high time complexity of TSBA, PMCNA is proposed to balance the solution quality and the running time. It works in an iterative fashion. Each iteration of PMCNA determines the criticality of nodes based on the contribution of nodes to single-link attacking plans and then assign resources to the most critical nodes. As a final step, CasL algorithm is used to determine the number of failed lines in the protected network.

Denote $\mathcal{A} = \{A_1, A_2, \ldots, A_{|E|}\}$ as the set of single-link attacking plans. Assume attacking plan A_l can fail p_l lines by altering demands for nodes in a set S_l. The demand of each node i in S_l is changed by $z_i \times 100\%$ of its maximum rate change ρ_i. The contribution r_l^i of node i to attacking plan A_i is defined as

$$r_l^i = p_l \times \frac{\dfrac{\rho_i z_i}{c_i(z_i)}}{\sum_{j\in S_l} \dfrac{\rho_j z_j}{c_j(z_j)}} \tag{39}$$

The overall contribution of node i is then $r_i = \sum_{l\in\mathcal{A}} r_l^i$. Intuitively, a node has more contribution if it participates in attacking plans that cause more line failure and its demand can be altered with a low cost. The detail of PMCNA is described in Alg. 4.

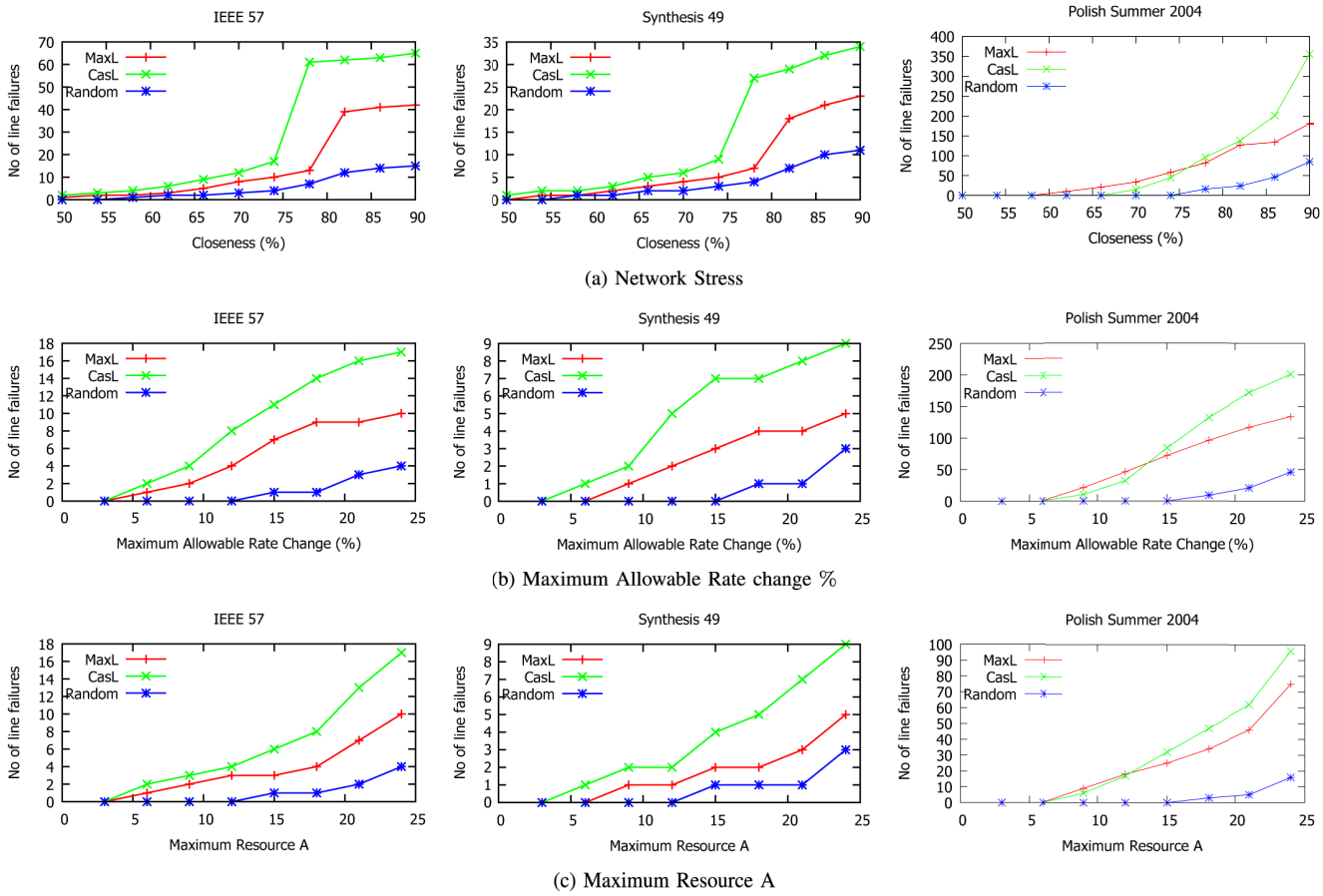


Fig. 4. Number of line failures.

Algorithm 4: Protect Most Critical Nodes Algorithm (PMCNA)

Data: Connected Power Grid Network G(N,E), P, D
Result: Protection strategy $\boldsymbol{y} = \{y_i\}, i \in D$, Total number of failed lines F
Initialize temporary cost $T = 0$
while $R_P - T > 0$ **do**
  Calculate cascading potential for each single line $e \in E$, based on algorithm in [20].
  Rank lines based on their cascading potential.
  Calculate the optimal attacking plans for all single line attack based on equation (31)-(38).
  Calculate contribution r_i for each node $i \in D$, denote the node with highest contribution as j.
  Assign $\min\{\gamma_j, R_P - T\}$ resource to node j. $T = T + \min\{\gamma_j, R_P - T\}$
  Set $y_j = 1$ in $\boldsymbol{y}$.
Calculate F based on CasL
Return $\boldsymbol{y}, F$

VI. Performance Evaluation

In this section, we evaluate the efficiency of the different algorithms we proposed. In the experiments reported in this section we used a 3.0 GHz Xeon machine with 2 MB L2 cache and 12 GB RAM. All experiments were run using a single core. The LP/IP solver was Cplex [24], with default settings. 50 runs of each cycle was run and averaged for consistency. We use the random algorithm as a baseline to compare our proposed algorithms.

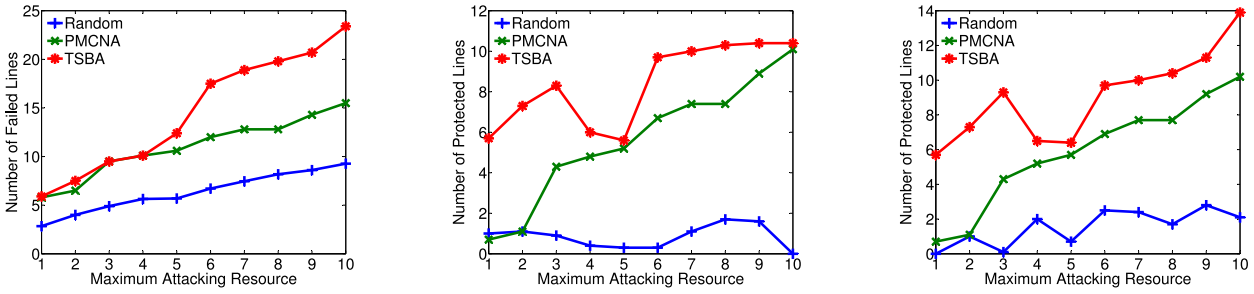
A. DataSet

For the experiments, we used the datasets of following types:

1) Two of the IEEE test cases [25]: the 57 bus case (57 nodes, 78 arcs, 4 generators and 38 demand nodes).
2) A simulated square grid network with 49 nodes and 84 lines, 5 generators and 11 demand nodes. The process to construct the related data about generators, demand nodes and lines follows. [22].
3) Polish system [2736sp]: Polish power flow system during Summer 2004. [26].

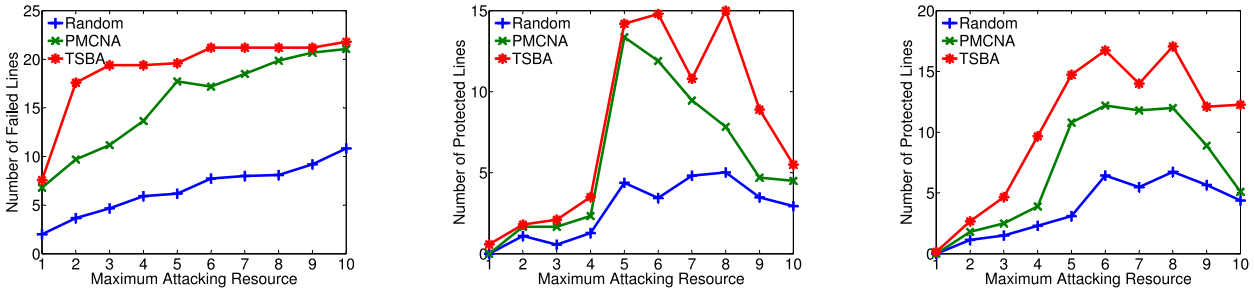
B. Number of Line Failures

We provide the comparative analysis of the behavior of the different proposed algorithms in three scenarios of datasets. In the Fig. 4a, the algorithms are tested against the network stress. To increase the network stress, the average demands of the users is increased over the whole network. This leads to unique solution in f_{ij}, such that the flows approach the capacity of their corresponding transmission lines. We vary the network stress (closeness) from 50% to 90% to check the response of the three algorithms. We see CasL break the network in a significant way at about $70-80\%$ of the network stress, while MaxL has a bump in the number of transmission line failures when the network is about $77-87\%$ stressed. CasL definitely



(a) No Protection Resource (b) Protection Resource = 3 (c) Protection Resource = 5

Fig. 5. Results From Square Grid.



(a) No Protection Resource (b) Protection Resource = 3 (c) Protection Resource = 5

Fig. 6. Results From IEEE 118 Bus.

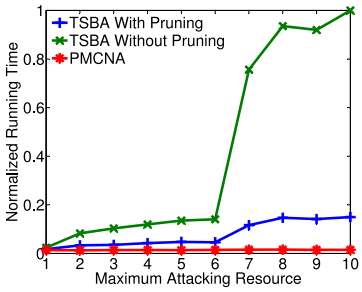


Fig. 7. Running Time Comparison.

performs better than MaxL and way better than Random. We also infer that the network stress is an important factor while considering this type of attacks.

Next the reaction of various algorithms while we change the maximum alterable rate change δ_i is evaluated. Note that even if there is a change in the maximum alterable rates, the increase in demand also depends on the billing profile of the user. And hence after a certain variation in maximum alterable rates, there is no impact on the demand of the user. δ_i is varied from 5% to 25% to observe the performance. In this case, we keep the maximum allowable resource constant at 25% to maintain consistency. As argued, the declining effect of the δ_i change, in Fig. (4b) it is evident that as the increase δ_i, the overall change in the number of transmission line failures although increases but at a smaller rate, implying that δ_i change does have diminishing impact because of baseline settings and user sensitivity. Finally we evaluate with the change of maximum resource allocated A to the attackers, which is basically how many houses and how much of rate change can be attacked. In Fig. 4c, it can be observed that as A is increased from 5% to 25%, the increase in the failure increases at a faster rate as the attacker has the opportunity to attack more users with a higher degree of rate change with a various combinations of those. Note that in case of the polish power system, CasL algorithm does not well in the less stressed or low resource environment as it depends a lot on the overall stress of the environment. As the stress on the power grid and attacking parameters such as maximum resource and maximum allowable rate change increases, we see CasL performing really well.

C. Evaluation of Protection Measures

1) Protection Results With Varying Attacking Resource: First we present the scenarios with varying R_A and protection is not allowed. Such scenarios provide an overview of how disastrous Demand Attack can be. Notice that in this case, the result of TSBA is only the result of Stage 1 and the best attacking plan is selected. Fig. 5a, 6a indicates that using TSBA, more than 20 lines can be failed in the worst case for both data sets, which corresponds to 11% and 25% of the total lines in IEEE 118 bus and simulated square grid, respectively. The performance of PMCNA is generally not far from TSBA and is close in some scenarios. Both TSBA and PMCNA performs better than the random algorithm.

Then we present the scenarios with fixed R_P and R_A. For each algorithm and each scenario, we consider number of failed lines before protection is applied (N_{bp}) and

the same number after protection (N_{ap}). Their difference, $N_p = N_{bp} - N_{ap}$, denotes the number of protected lines by the protection scheme and serves as the measurement of the efficiency of the protection algorithms. Fig. 5b, 5c displays the results with $R_P = 3$ and $R_P = 5$, respectively. TSBA performs better than the other two algorithms, as expected. In some cases, the performance of PMCNA is close to TSBA, which shows the quality of the heuristic algorithm. Normally, N_P increases with R_A since N_{bp} tends to be high with higher R_A and N_{ap} stays the same with a fixed protection plan. However, this does not always hold. As in the case when R_A is between 3 and 5 in Fig. 5b, 5c. It can be explained by the fact that increase in R_A enables the attacker to stay at the attacking strategy even if the cost to attack is higher. The same behavior is also observed in 6b, 6c when $R_A > 7$.

2) Running Time Comparison: Moreover, we evaluate the effectiveness of the pruning methods proposed in Section V-C1 and the efficiency of PMCNA. To test the pruning methods, we ran TSBA twice on the same simulated square grid network. The pruning methods were enabled in one run and disabled in the other. From Fig. 7, it can be concluded that the pruning methods are effective. They can save up to 85% of running time. Additionally, PMCNA is up to 10 times faster than TSBA with pruning. The times are normalized by setting the largest running time in the experiment to 1.

VII. Related Work

The studies [11], [12] expose the structural vulnerability of the power grid, where the power system is shown to be vulnerable against the hidden failures when the DC power flow system is considered. In [13], a new criterion of reliable strategies for defending power systems is derived and two allocation algorithms have been developed to seek reliable strategies for two types of defense tasks. Load redistribution attacks in Power Systems are modeled by Yuan *et al.* [14]. They develop the concept of load redistribution (LR) attacks, a special type of false data injection attacks, and analyze their damage to power system operation in different time steps with different attacking resource limitations. They identify the most damaging LR attack through a max-min attacker-defender model and provide a protection strategy. However, all the above approaches do not include the role of communication network in the new model of the smart grid.

A survey and study of Internet based load altering attacks against smart Grid was presented in [8]. Defense of consumption sector through protection of command and price signals, protection of smart meters, attack detection and learning of demand patterns, load shedding and load relocation are proposed along with cost-efficient load protection for Type-III attacks. However they do not analyze the effect of rate alteration through Internet and only single stage protection is considered. To the best of our knowledge, our paper is first of its kind to study the vulnerability of the power grid to price modification in the communication in the smart grid.

VIII. Conclusion

We define a novel problem of price modification attacks in the smart grid and show the hardness of the problem and inapproximability. Two approaches are studied to exploit the smart grid vulnerability to maximize the number of line failures. IP and cascading extension for the first approach and cascading ranking algorithm is provided as the second approach. Protection scheme against PMA was also devised that consisted of the bi-level IP formulation and an efficient heuristic algorithm PMCNA. Experimental results on both IEEE bus data and synthetic data along with real data for this new problem, give us insightful knowledge about the reaction of various approaches to different network settings and parameters.

References

[1] B. Liscouski and W. Elliot, *US-Canada Power System Outage Task Force, Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations*, U.S. Dept. Energy, Apr. 2004.

[2] *Energy Independence and Security Act of 2007*. [Online]. Available: http://www.gpo.gov/fdsys/pkg/PLAW-110publ140/html/PLAW-110publ140.htm, accessed Nov. 2015.

[3] N. Ye, J. Giordano, and J. Feldman, "Securing the electricity grid," *Commun. ACM*, vol. 44, no. 8, pp. 76–82, Aug. 2001.

[4] O. Kosut, L. Jia, R. J. Thomas, and L. Tong, "Malicious data attacks on smart grid state estimation: Attack strategies and countermeasures," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, Gaithersburg, MD, USA, Oct. 2010, pp. 220–225.

[5] S. Mishra *et al.*, "Rate alteration attacks in smart grid," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Hong Kong, 2015, pp. 2353–2361.

[6] D. Bienstock, "Optimal control of cascading power grid failures," in *Proc. PES Gen. Meeting*, 2011, pp. 2166–2173.

[7] P. Vytelingum, S. D. Ramchurn, T. D. Voice, A. Rogers, and N. R. Jennings, "Trading agents for the smart electricity grid," in *Proc. 9th Int. Conf. Auton. Agents Multiagent Syst.*, vol. 1. Toronto, ON, Canada, 2010, pp. 897–904.

[8] A.-H. Mohsenian-Rad and A. Leon-Garcia, "Distributed Internet-based load altering attacks against smart power grids," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 667–674, Dec. 2011.

[9] T. T. Kim and H. V. Poor, "Strategic protection against data injection attacks on power grids," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 326–333, Jun. 2011.

[10] P.-Y. Chen, S.-M. Cheng, and K.-C. Chen, "Smart attacks in smart grid communication networks," *IEEE Commun. Mag.*, vol. 50, no. 8, pp. 24–29, Aug. 2012.

[11] G. Chen, Z. Y. Dong, D. J. Hill, G. H. Zhang, and K. Q. Hua, "Attack structural vulnerability of power grids: A hybrid approach based on complex networks," *Physica A Stat. Mech. Appl.*, vol. 389, no. 3, pp. 595–603, 2010.

[12] G. Chen, Z. Y. Dong, D. J. Hill, and G. H. Zhang, "An improved model for structural vulnerability analysis of power networks," *Physica A Stat. Mech. Appl.*, vol. 388, no. 19, pp. 4259–4266, 2009.

[13] G. Chen, Z. Y. Dong, D. J. Hill, and Y. S. Xue, "Exploring reliable strategies for defending power systems against targeted attacks," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1000–1009, Aug. 2011.

[14] Y. Yuan, Z. Li, and K. Ren, "Modeling load redistribution attacks in power systems," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 382–390, Jun. 2011.

[15] A. R. Bergen and V. Vittal, *Power Systems Analysis*. Upper Saddle River, NJ, USA: Prentice-Hall, 1999.

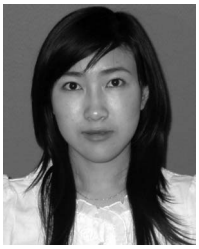
[16] P. Crucitti, V. Latora, and M. Marchiori, "Model for cascading failures in complex networks," *Phys. Rev. E*, vol. 69, no. 4, 2004, Art. ID 045104.

[17] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Phys. Rep.*, vol. 424, nos. 4–5, pp. 175–308, 2006.

[18] I. Dobson, B. A. Carreras, V. E. Lynch, and D. E. Newman, "Complex systems analysis of series of blackouts: Cascading failure, critical points, and self-organization," *Chaos Interdiscipl. J. Nonlin. Sci.*, vol. 17, no. 2, 2007, Art. ID 026103.

[19] J.-W. Wang and L.-L. Rong, "Cascade-based attack vulnerability on the US power grid," *Safety Sci.*, vol. 47, no. 10, pp. 1332–1336, 2009.
[20] A. Bernstein, D. Bienstock, D. Hay, M. Uzunoglu, and G. Zussman, "Power grid vulnerability to geographically correlated failures—Analysis and control implications," in *Proc. IEEE INFOCOM*, Toronto, ON, Canada, 2014, pp. 2634–2642.
[21] J. Chen, J. S. Thorp, and I. Dobson, "Cascading dynamics and mitigation assessment in power system disturbances via a hidden failure model," *Int. J. Elect. Power Energy Syst.*, vol. 27, no. 4, pp. 318–326, 2005.
[22] D. Bienstock and A. Verma, "The *N-k* problem in power grids: New models, formulations, and numerical experiments," *SIAM J. Optim.*, vol. 20, no. 5, pp. 2352–2380, 2010.
[23] D. Van Hertem, J. Verboomen, K. Purchala, R. Belmans, and W. L. Kling, "Usefulness of DC power flow for active power flow analysis with flow controlling devices," in *Proc. 8th IEE Int. Conf. ACDC Power Transm.*, 2006, pp. 58–62.
[24] (2014). *IBM ILOG CPLEX Optimization Studio*. [Online]. Available: http://www-03.ibm.com/software/products/en/ibmilogcpleoptistud
[25] P. Wong *et al.*, "The IEEE reliability test system—1996," *IEEE Trans. Power Syst.*, vol. 14, no. 3, pp. 1010–1020, Aug. 1999.
[26] MATPOWER. *A MATLAB Power System Simulation Package*. [Online]. Available: http://www.pserc.cornell.edu//matpower/, accessed Nov. 2015.

Subhankar Mishra received the B.Tech. degree from the National Institute of Technology, Rourkela, India. He is currently pursuing the Ph.D. degree in computer science with the Department of Computer and Information Science and Engineering, University of Florida, USA, under the supervision of Dr. M. T. Thai. His current research includes approximation algorithms and security in smart grid.



Xiang Li received the M.Sc. degree from the Academy of Mathematics Systems and Science, Chinese Academy of Sciences, Beijing, in 2012, and the M.Sc. degree in industrial and systems engineering from the University of Florida, where she is currently pursuing the Ph.D. degree with the Department of Computer and Information Science and Engineering. Her current research interests include online social networks, network vulnerability, algorithms, and security in smart grid.

Tianyi Pan is currently pursuing the Ph.D. degree with the Department of Computer and Information Science and Engineering, University of Florida, under the supervision of Dr. M. T. Thai. His research focuses on optimization problems in online social networks, smart grids, and cellular networks.

Alan Kuhnle is currently pursuing the Ph.D. degree in computer science with the University of Florida. His research interests include approximation algorithms for graph problems, online algorithms, and multiplex networks.

My T. Thai (M'06) received the Ph.D. degree in computer science from the University of Minnesota, in 2005. She is a Professor with the Department of Computer and Information Science and Engineering, University of Florida. Her current research interests include algorithms and optimization on network science and engineering. She was a recipient of several research awards, including a UF Provosts Excellence Award for Assistant Professors, a DoD YIP, and an NSF CAREER Award. She has engaged in many professional activities, such as being the PC Chair of the EEE IWCMC 2012, the IEEE ISSPIT 2012, and COCOON 2010. She is the Founding Editor-in-Chief of *Computational Social Networks*, an Associate Editor of *JOCO* and the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and a Series Editor of *SpringerBriefs in Optimization*.

Jungtaek Seo, photograph and biography not available at the time of publication.