This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems

Line Failure Detection after a Cyber-Physical Attack on the Grid Using Bayesian Regression

Saleh Soltan, *Member, IEEE*, Prateek Mittal, *Senior Member, IEEE*, H. Vincent Poor, *Fellow, IEEE*

***Abstract*—We study the problem of line failure detection following a cyber-physical attack. Since such attacks can result in line trippings (by remotely activating switches) as well as loss of measurement feeds, we consider an attack model in which an adversary attacks an area by: (i) disconnecting some lines within the attacked area, and (ii) blocking the measurements coming from inside the attacked area from reaching the control center. Hence, after the attack, voltage phase angles of the buses and status of the lines inside the attacked area become unavailable to the grid operator. We build upon a recently introduced convex optimization method for detecting line failures and exploit Bayesian regression to develop the novel PROBER Algorithm for probabilistically detecting line failures after an attack using partial noisy measurements. The PROBER Algorithm provides the probability that each line is failed inside the attacked area in a running time which is independent of the number of line failures. Hence, these probabilities can be efficiently computed and used to make the existing brute force search methods tractable (for detecting multiple-line failures) by significantly reducing their search space. We numerically demonstrate that such an approach hits a sweet spot in accuracy and efficiency.**

***Index Terms*—Power grid, state estimation, cyber-physical attacks, Bayesian regression, machine learning**

I. Introduction

Power grids have been repeatedly shown in recent years to be vulnerable to cyber attacks [2], [3], [4]. In the most consequential of such attacks, about 200 thousand people lost their electric power due to a cyber attack on Ukrainian grid in December 2015 [2]. Smaller scale attacks on regional power grids have also been shown in a recent report to be common and pervasive on the U.S. grid [3]. Moreover, it was recently discovered that hackers were able to access to part of U.S. grid control systems and had reached a point where they could have caused a blackout [4].

In order to improve the grid's readiness against future cyber attacks, in this paper, we build upon recent advancements in Bayesian data analysis to provide a new algorithm for detecting line failures following such attacks. Since a cyber attack can result in line trippings (by remotely activating switches) as well as measurement losses, we follow the work by Soltan et al. [5] and consider an attack in which an adversary attacks an area by: (i) disconnecting some lines within the attacked area, and (ii) blocking the measurements coming from inside the attacked area from reaching the control center in order to mask the failed lines (as shown in Fig. 1). Hence, after the attack, the phase angles of the buses (nodes) and the status of the lines inside the attacked area become unavailable to the grid operator. Our objective is to use the phase angle measurements before and partial measurements after the attack to detect the failed lines. Detecting failed lines is necessary for obtaining a correct understanding of the grid topology and estimating the power flows on the lines after the attack.

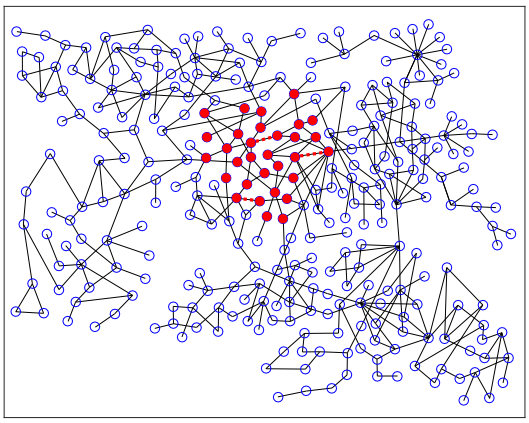


Fig. 1: An example of an attack on the IEEE 300-bus system. An adversary attacks an area by disconnecting some lines within the attacked area and blocking the measurements coming from inside the attacked area from reaching the control center in order to mask the failed lines. Buses in the attacked area and the failed lines are shown by the red filled nodes and the red dashed lines, respectively.

In this work, we build upon the efficient convex optimization method for detecting failed lines in noise-free scenarios by Soltan et al. [5] and use Bayesian regression [6] to detect failed lines probabilistically based on noisy measurements [7]. In Particular, we exploit Bayesian Regression with Automatic Relevance Determination (ARD) model provided by Drugowitsch [8] to develop the PRObabilistic line failure detection using BayEsian Regression (PROBER) Algorithm. The PROBER Algorithm provides the probability that each line is failed inside the attacked area after an attack, instead of 0-1 hard decision on the status of the lines, which reflects the uncertainty in line failure detection due to the measurements noise. *One of the main advantages of the PROBER Algorithm is that its running time, unlike previous methods that could deal with noisy measurements [9], [10], [11], is independent of the number of line failures.* Therefore, it remains tractable as the number of line failures increases.

The probabilities obtained by the PROBER Algorithm can

Authors are with the Department of Electrical Engineering at Princeton University, Princeton, NJ. Emails: {ssoltan,pmittal,poor}@princeton.edu

This work is supported in part by the Siebel Energy Institute, in part by the National Science Foundation under Grants DMS-1736417, ECCS-1824710, and CNS-1553437, and in part by the Office of Naval Research YIP Award. We also thank Ghazal Fazelnia for her helpful insights and comments. A partial and preliminary version of this work appeared in Proc. IEEE PES-GM'18 [1].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems

then be used for line failure detection by: (i) considering a probability threshold p and detecting all lines with probability of being failed greater than p as failed, and (ii) utilizing them to reduce the running times of the Brute Force Search (BFS) methods by limiting their search only to the lines with high probability of being failed. We numerically compare these two approaches with previously developed methods [5], [9], [10] in the IEEE 300-bus system and demonstrate that the second approach provides accuracy similar to (and sometimes better than) the BFS based methods but in an exponentially faster running time (e.g., in an attacked area with 15 lines, it runs 40 times faster). Hence, by utilizing the probabilities obtained by the PROBER Algorithm to improve the BFS running time, *our proposed method hits a sweet spot in accuracy and efficiency.*

The rest of this paper is organized as follows: Section II discusses the related work and Section III presents the attacked model and preliminary backgrounds. Section IV provides an overview of the previous line detection methods. In Section V, we present the PROBER Algorithm and in Section VI, we numerically evaluate its performance. Finally, Section VII provides concluding remarks and future directions.

II. Related Work

Power systems' vulnerability to failures and uncertainties has been widely studied in the past couple of years [12], [13], [14], [15]. In particular, false data injection attacks on power grids and anomaly detection have been studied using the DC power flows [16], [17], [18], [19], [20], [21], [22], [23]. These studies have focused on the attacks that have no physical components. Hence, they cannot be used in the scenarios studied in this paper for detecting *actual* line failures. Similar attack scenarios have also been recently studied on the power distribution networks [24].

The problem of line failure detection using phase angle measurements during the normal operation of the grid have been studied by Tate and Overbye [9], [25], Garcia et al. [26], and Rovatsos et al. [11]. However, since the developed methods by these works for detecting line failures are BFS based, their running time grows exponentially as the number of line failures increases. Hence, these methods cannot be generalized for detecting higher order failures. Similar approaches with likelihood detection functions have been studied to address the PMU placement problem under the DC power flow model [10], [27], [28], [29], [30].

The problem of line failure detection in an area based on the information from external nodes was first studied by Zhu and Giannakis [31] using sparse recovery methods. The proposed algorithm works for only one and two-line failures, since it depends on the sparsity of line failures. Moreover, as in the previous works for line failure detection [9], [25], the running time of the proposed algorithm grows exponentially with the number of line failures. In a recent novel work, Zhao et. al. [32] have proposed a Neural Network based approach for detecting line failures without searching over all possible set of line failures. However, since it is a supervised learning approach, it requires an extensive amount of data for training the classifier under different operating conditions. Moreover, it cannot be used in the scenarios studied in our this paper since the attacked area is not known in advance (which can vary based on the attack), unless the classifier is trained on all possible attacked areas which can make the training phase numerically intractable.

Cyber attacks that result in wrong topology estimation by the grid operator without any real *physical* failures have been widely studied before [17], [18], [19], [20], [33], [34], [35]. Attack scenarios similar to the one studied in this paper have been first introduced by Soltan et al. [36]. They suggested that the algebraic properties of the DC power flow equations and their relationship to the topology of the grid can be utilized to detect line failures in polynomial time (and independent of the number of line failures) by a Linear Program (LP) under some topological constraints. However, theoretical performance guarantees of their methods were limited to the noise-free scenarios. Moreover, the adaptation of their method to noisy scenarios may result in false positives and negatives when the noise-level is high [5]. Other recent works have studied similar cyber-physical attacks that can also be accompanied by false data injection under different assumptions on the attackers' knowledge of the system [37], [38], [39], [40], [41], [42], [43], [44]. These works, however, focus mostly on designing or detecting sophisticated false data injection attacks rather than computational complexity of detecting line failures after recovering the attacked area (which is the main focus of our work). In a recent work, Bienstock et. al. [45] have designed defenses against attacks that are proved to be undetectable right after an attack but can be detected if the grid operator actively changes the operating points of the generators in order to acquire more information regarding the state of the grid. In this work, however, we only rely on the measurements right after the attack as the only source of information for line failures detection.

III. Model and Preliminaries

A. DC Power Flow Model

We use the linearized DC power flow model, which is widely used as an approximation for the non-linear AC power flow model in studying the vulnerabilities of power grids. We represent the power grid by a connected directed graph $G = (V, E)$ where $V = \{1, 2, \ldots, n\}$ and $E = \{e_1, \ldots, e_m\}$ are the set of nodes and lines (with arbitrary orientation) corresponding to the buses and transmission lines, respectively. Each node $i \in V$ has a supply or demand value p_i and its *state* is determined by its voltage phase angle value θ_i. Each line $e_j = (u, v) \in E$ is characterized by its *reactance* $x_{uv} = x_{e_j}$ and carries a power flow $f_{uv} = f_j$.

Under the DC power flow model, the relationship between the vector of power supply/demand values (or *bus injections*) $\vec{p} \in \mathbb{R}^{|V|}$, the reactance values, the vector of power flows on the lines $\vec{f} \in \mathbb{R}^{|E|}$, and the vector of phase angles $\vec{\theta} \in \mathbb{R}^{|V|}$ are determined by the following matrix equations:

$$\mathbf{B}\mathbf{D}^t\vec{\theta} = \vec{f}, \tag{1}$$

$$\mathbf{A}\vec{\theta} = \vec{p}, \tag{2}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems

where $\mathbf{D} \in \{-1, 0, 1\}^{|V|\times|E|}$ is the *incidence matrix* of G defined as,

$$d_{ij} = \begin{cases} 0 & \text{if } e_j \text{ is not incident to node } i, \\ 1 & \text{if } e_j \text{ is coming out of node } i, \\ -1 & \text{if } e_j \text{ is going into node } i, \end{cases}$$

$\mathbf{B} := \text{diag}([1/x_{e_1}, 1/x_{e_2}, \ldots, 1/x_{e_m}])$ is a diagonal matrix with diagonal entries equal to the inverse of the reactance values, and $\mathbf{A} \in \mathbb{R}^{|V|\times|V|}$ is the *admittance matrix* of G defined as $\mathbf{A} := \mathbf{DBD}^T$. Given the vector of phase angles $\vec{\theta}$, the vector of power flows $\vec{f}$ and $\vec{p}$ can be determined uniquely based on (1)-(2). Therefore, the phase angle values are the power grid state variables.

Notation. If X, Y are two subgraphs of G, $\mathbf{A}_{X|Y}$ denotes the submatrix of $\mathbf{A}$ with rows from X and columns from Y. For any matrix $\mathbf{C}$, $\mathbf{C}^T$ denotes its transpose and $\mathbf{C}^+$ denotes its *Moore-Penrose pseudo-inverse*. For a vector $\vec{y}$, $\|\vec{y}\|_1 := \sum_{i=1}^{n} |y_i|$ denotes its l_1-norm, $\|\vec{y}\|_2 := (\sum_{i=1}^{n} y_i^2)^{1/2}$ denotes its l_2-norm, and $\text{supp}(\vec{y}) := \{i | y_i \neq 0\}$ denotes its support.

B. The Attack Model

Power grids are operated and controlled by the Supervisory Control And Data Acquisition (SCADA) systems [46]. The main components of a SCADA system are shown in Fig. 2. First, there are Intelligent Electronic Devices (IEDs) that measure the voltage magnitudes and phase angles every few seconds and can also be used for controlling the status of the circuit breakers—Phasor Measurement Units (PMUs) could also be used to measure the voltage magnitudes and phase angles with a higher rate but for the purpose of our work, the low rate IED measurements suffice. Then, there are Data Concentrators that collect the data from a group of IEDs and transfer it via the communication network to the control center. Each component of SCADA is vulnerable both to cyber and physical attacks.

We follow Soltan et al. [5] and assume that an adversary attacks an area $H = (V_H, E_H)$ (representing a subgraph of G) by: (i) disconnecting some lines within the attacked area, referred to as *failed lines* and denoted by F, and (ii) blocking the measurements coming from inside the attacked area to mask the status of the lines in H. Such an attack can be executed by an adversary in different ways. For example, an adversary can *physically* attack a set of lines within an area and simultaneously block the measurements by attacking the corresponding Data Concentrator or the communication links. Alternatively, an adversary can get access to a control center with a *cyber* attack (i.e., [2], [3], [4]) to hide the stream of data from the operator and simultaneously use its control over the IEDs to remotely activate the circuit breakers (as demonstrated in the cyber attack on the Ukrainian grid control system in December 2015 [2]). Any combination of such cyber or physical attacks can also result in a similar scenario.

After the attack, the phase angles of the nodes and the status of the lines in H become unavailable to the grid operator. The objective is to use the phase angle measurements before and partial measurements after the attack to detect the failed lines. We assume that phase angle measurements are available at all buses outside of the attacked area.[1] Detecting failed lines is necessary for obtaining a correct understanding of the grid topology, estimating the power flows on the lines after the attack, and for executing an effective control algorithm (e.g., power grid intentional islanding or load shedding) in order to stop the initial attack from affecting the entire grid through a cascade of failures and causing a major blackout.

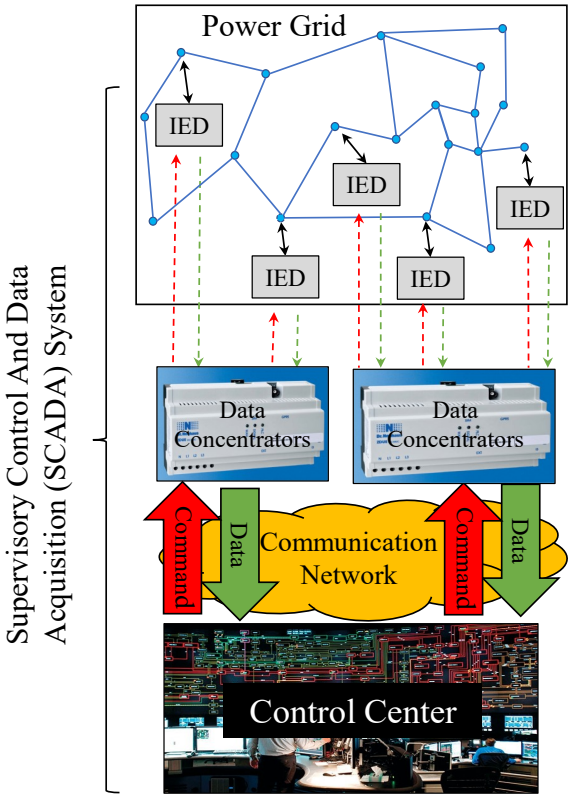


Fig. 2: The main components of the power grid's SCADA system. The IEDs measure voltage phase angles and send this information via Data Concentrators to the Control Center. Moreover, they can be used to remotely control the status of the circuit breakers.

We use the prime symbol ($'$) to denote the values after an attack. For example, $\mathbf{A}'$ and $\vec{\theta}'$ denote the admittance matrix and phase angle of the nodes after the attack. We denote the complement of the attacked area H by $\bar{H} = G \backslash H$. $\vec{\theta}_H$ and $\vec{\theta}_{\bar{H}}$ denote the vectors of phase angles of the nodes in H and $\bar{H}$, respectively. Without loss of generality, we also assume that $E_H = \{e_1, e_2, \ldots, e_{|E_H|}\}$.

Using this notation, we assume that the control center receives T noisy measurements $\vec{\theta}^{(1)}, \vec{\theta}^{(2)}, \ldots, \vec{\theta}^{(T)}$ in T different times for the phase angles of the nodes in the grid before the attack, and T noisy measurements $\vec{\theta}'^{(1)}_{\bar{H}}, \vec{\theta}'^{(2)}_{\bar{H}}, \ldots, \vec{\theta}'^{(T)}_{\bar{H}}$ in T different times for the phase angles of the nodes outside of the attacked area after the attack. The T measurements are from different but very close time intervals (e.g., within microseconds). The measurements after the attack are assumed to be associated with the time that the system has been stabilized after the attack.

[1]Notice that this assumption does not limit the scope of our work since all the buses with no measurement devices can simply be considered as part of the attacked area—which does not necessarily need to be connected.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems

The noise corresponds to the measurement noise [7] as well as fluctuations in the supply/demand vector $\vec{p}$. We assume that for each $1 \leq i \leq T$, the i^{th} measurement $\vec{\theta}^{(i)} = \vec{\theta} + \vec{e}^{(i)}$, in which $\vec{e}^{(i)} \sim \mathcal{N}(0, \beta\mathbf{I})$ has a multivariate Gaussian distribution with a zero mean and a diagonal covariance matrix $\beta\mathbf{I}$, and $\mathbf{A}\vec{\theta} = \vec{p}$.[2] We assume the same for the measurements after the attack, $\vec{\theta}'^{(i)}_{\bar{H}} = \vec{\theta}'_{\bar{H}} + \vec{e}^{(i)}_{\bar{H}}$, in which $\vec{e}^{(i)}_{\bar{H}} \sim \mathcal{N}(0, \beta\mathbf{I}_{\bar{H}})$, and $\mathbf{A}'\vec{\theta}' = \vec{p}$. To measure noise level, we define the Signal to Noise Ratio (SNR) based on the phase angles before the attack as $SNR := 20\log_{10}(\|\vec{\theta}\|_2/\sqrt{|V|\beta})$.

One way of using the T measurements to obtain more accurate measurements is to take an average of these vectors. Hence, we define $\vec{\theta}^{(\mu)} := (\sum_i \vec{\theta}^{(i)})/T$ and $\vec{\theta}'^{(\mu)}_{\bar{H}} := (\sum_i \vec{\theta}'^{(i)}_{\bar{H}})/T$.

C. Bayesian Regression

Given a data set $\mathcal{D} = \{\mathbf{X}, \mathbf{Y}\}$, with $\mathbf{X} = [\vec{x}_1, \ldots, \vec{x}_n] \in \mathbb{R}^{d\times n}$ and $\mathbf{Y} = [y_1, \ldots, y_n] \in \mathbb{R}^{n\times 1}$, in linear regression, we are interested in finding a vector $\vec{w} \in \mathbb{R}^d$ that minimizes $\|\vec{w}^T\mathbf{X} - \mathbf{Y}\|_2$. Such a vector $\vec{w}$ determines the relationship between in the input vectors $\mathbf{X}$ and outputs $\mathbf{Y}$.

The linear regression provides very little information on the uncertainties in the computed vector $\vec{w}$, especially when $\mathbf{Y}$ is noisy. One way of overcoming this issue is by adopting the Bayesian approach to regression [6]. In the Bayesian regression, instead of finding *the maximum likelihood* estimate for a vector $\vec{w}$ that describes the relationship between the inputs and the outputs, we are interested in computing a probability distribution on all possible vectors $\vec{w}$ that describe this relationship. This can be done by computing the *posterior* distribution on $\vec{w}$ using Bayes' rule, by assuming a *prior* distribution on $\vec{w}$ and an appropriate model for the way output data is related to the input. Due to the space constraints, for further details on Bayesian Regression see [6, Section 3.3].

Some sparsity constraints on the coefficients, as in the Lasso regression [6], can also be obtained in Bayesian regression by appropriate choice of the prior for the coefficients. In particular, in this paper, we follow the Bayesian Regression with Automatic Relevance Determination (ARD) model provided by Drugowitsch [8]. This model assumes a linear relationship between inputs $\vec{x}$ and outputs y, and constant-variance Gaussian noise, such that the likelihood is given by

$$\mathrm{P}(y|\vec{x}, \vec{w}, \tau) = \mathcal{N}(y|\vec{w}^T\vec{x}, \tau^{-1}) = \Big(\frac{\tau}{2\pi}\Big)^{1/2} \exp\Big(-\frac{\tau}{2}(y - \vec{w}^T\vec{x})^2\Big),$$

and the prior on $\vec{w}, \tau^{-1}$ is conjugate normal inverse-gamma

$$\mathrm{P}(\vec{w}, \tau|\vec{\alpha}) = \mathcal{N}\Big(\vec{w}|0, (\tau\mathrm{diag}(\vec{\alpha}))^{-1}\Big)\mathrm{Gam}(\tau|a_0, b_0), \quad (3)$$
$$\mathrm{Gam}(\tau|a_0, b_0) = \frac{b_0^{a_0}}{\Gamma(a_0)}\tau^{a_0-1}\exp(-b_0\tau),$$

[2]Notice that since any linear combination of independent Gaussian random variables still has a Gaussian distribution, it is easy to show that $\mathbf{A}(\vec{\theta} + \vec{e}_1) = \vec{p} + \vec{e}_2$ for vectors of independent Gaussian variables $\vec{e}_1, \vec{e}_2$ is equivalent to $A(\vec{\theta} + \vec{e}_3) = \vec{p}$ for a vector of independent Gaussian variables $\vec{e}_3$. So, the model that we considered here is equivalent to the model that adds separate Gaussian noise vectors to $\vec{\theta}$ and $\vec{p}$ to capture the measurement noise as well as fluctuations in the supply/demand vector.

where $\mathrm{diag}(\vec{\alpha})$ is a diagonal matrix with entries given by vector $\vec{\alpha} = [\alpha_1, \ldots, \alpha_d]$. The entries of $\vec{\alpha}$ are independent and the hyper-prior is given by

$$\mathrm{P}(\alpha) = \prod_i \mathrm{Gam}(\alpha_i|c_0, d_0). \quad (4)$$

Under this model, analytically computation of the posterior distribution $\mathrm{P}(\vec{w}, \tau, \vec{\alpha}|\mathcal{D})$ is intractable. Hence, the Variational Inference approach can be used to *approximate* the posterior distribution [8]. In this approach, it is assumed that the posterior distribution can be factorized as $\mathrm{Q}(\vec{w}, \tau)\mathrm{Q}(\vec{\alpha})$ which can be computed as (for further details on how to compute $\vec{w}_n, \mathbf{V}_n, a_n, b_n, c_n, d_{ni}$ see [8])

$$\mathrm{Q}(\vec{w}, \tau) = \mathcal{N}(\vec{w}|\vec{w}_n, \tau^{-1}\mathbf{V}_n)\mathrm{Gam}(\tau|a_n, b_n), \quad (5)$$
$$\mathrm{Q}(\vec{\alpha}) = \prod_i \mathrm{Gam}(\alpha_i|c_n, d_{ni}).$$

In Section V, we use this model to detect the failed lines after an attack described in the previous subsection.

IV. Detection Methods

The problem of failed lines detection using partial phase angle measurements is NP-hard in general [43]. However, in special cases, it is possible to efficiently solve this problem. In this section, we provide an overview of the two main approaches to failed lines detection using phase angle measurements.

In recent years, some efforts have been made to apply learning algorithms to this problem, such as the work by Garcia et al. [26], [32]. However, since detection using supervised learning (without considering underlying power flow equations) is limited to the state of the grid that they are trained on, such methods are not general enough for consideration in this paper (i.e., since the attacked area is not known is advance one cannot train a classifier for a particular attack before it happens).

A. Brute Force Search

The classical approach to the failed lines detection problem is the Brute Force Search (BFS) [9], [10]. The BFS based methods consider all possible set of failed lines and return the set with the maximum likelihood, based on the observed measurements.

Assume $\mathcal{M}$ is the set of all admittance matrices associated with the graphs that can be obtained by removing any number of lines in E_H from the graph G. One way of finding the set of failed lines is by searching the entire space of matrices $\mathcal{M}$ to find a matrix $\mathbf{C}$ that minimizes the following

$$\min_{\mathbf{C}\in\mathcal{M}} \|(\mathbf{C}^+\mathbf{A}\vec{\theta}^{(\mu)})_{\bar{H}} - \vec{\theta}'^{(\mu)}_{\bar{H}}\|_2. \quad (6)$$

To see why (6) is an intuitive approach for detecting the set of failed lines, assume $F_{\mathbf{C}} \subseteq E_H$ is the set of lines that their removal results in the admittance matrix $\mathbf{C}$. Notice that $\mathbf{A}\vec{\theta}^{(\mu)} \approx \vec{p}$. Therefore, $\mathbf{C}^+\mathbf{A}\vec{\theta}^{(\mu)}$ is the approximate phase angles that one would have expected to observe if $F_{\mathbf{C}}$ was the actual set of failed lines. By comparing the expected phase angle of the nodes outside of the attacked area $(\mathbf{C}^+\mathbf{A}\vec{\theta}^{(\mu)})_{\bar{H}}$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems

with the average observed phase angles $\vec{\theta}'^{(\mu)}_{\bar{H}}$, one can check how much the guessed set of failed lines $F_{\mathbf{C}}$ is consistent with the observed data.

Despite the simplicity and effectiveness of the BFS based methods, however, their running time grow exponentially with the number of lines in the attacked area. This makes them inapplicable to the scenarios that require fast decision making in order to localize the attack and reduce its consequences. In Section VI, we numerically investigate the limitations of the BFS based methods.

B. Convex Optimization

Another approach to the failed lines detection problem is to use the topological properties of the grid in order to find the failed lines more efficiently for certain topologies. Such methods and their connection to the topology of the grid were first fully explored by Soltan et al. [5]. However, similar approaches were studied before by Zhu and Ginnakis [31] without making the connection to the topology.

It is proved by Soltan et al. [5] that under some conditions on the topology of the attacked area, the optimal solution $\vec{t} \in \mathbb{R}^{|E_H|}$ and $\vec{z} \in \mathbb{R}^{|V_H|}$ of the following convex optimization problem is such that $\text{supp}(\vec{t}) = \{i | e_i \in F\}$ and $\vec{z} = \vec{\theta}'_H$:

$$\begin{aligned} &\min_{\vec{t},\vec{z},\epsilon} \|\vec{t}\|_1 + \lambda\epsilon \text{ s.t.} \qquad (7)\\ &\|\mathbf{A}_{H|H}(\vec{\theta}^{(\mu)}_H - \vec{z}) + \mathbf{A}_{H|\bar{H}}(\vec{\theta}^{(\mu)}_{\bar{H}} - \vec{\theta}'^{(\mu)}_{\bar{H}}) - \mathbf{D}_H\vec{t}\|_2 \le \epsilon \\ &\|\mathbf{A}_{\bar{H}|H}(\vec{\theta}^{(\mu)}_H - \vec{z}) + \mathbf{A}_{\bar{H}|\bar{H}}(\vec{\theta}^{(\mu)}_{\bar{H}} - \vec{\theta}'^{(\mu)}_{\bar{H}})\|_2 \le \epsilon. \end{aligned}$$

Therefore, by solving (7) the nonzero elements of $\vec{t}$ reveal the failed lines in a polynomial running time and independent of the number of line failures. It can be shown that in this case, if $e_i = \{j,k\} \in F$, then $t_i \approx -a_{jk}(\theta'_j - \theta'_k)$ which is the amount of power that the line e_i would have carried if it was not failed. The main advantage of this approach is that it can detect all the line failures simply by detecting nonzero elements of $\vec{t}$. Hence, its running time is *independent* of the number of line failures in oppose to the BFS based methods described in the previous subsection.

The main idea of (7) is that if there were no measurement noise, then $\mathbf{A}\vec{\theta} - \mathbf{A}'\vec{\theta}' = 0$. Hence, one could rewrite this equation as $\mathbf{A}(\vec{\theta} - \vec{\theta}') = (\mathbf{A}' - \mathbf{A})\vec{\theta}'$. By breaking this matrix equation by the rows associated with nodes in $\bar{H}$ and H, one would have $\mathbf{A}_{\bar{H}|G}(\vec{\theta} - \vec{\theta}') = (\mathbf{A}'_{\bar{H}|G} - \mathbf{A}_{\bar{H}|G})\vec{\theta}'$ and $\mathbf{A}_{H|G}(\vec{\theta} - \vec{\theta}') = (\mathbf{A}'_{H|G} - \mathbf{A}_{H|G})\vec{\theta}'$. Now, since all the line failures are within H, $\mathbf{A}'_{\bar{H}|G} - \mathbf{A}_{\bar{H}|G} = 0$. And from the way $\mathbf{A}$ is changed after a set of line failures, one can show that and $(\mathbf{A}'_{H|G} - \mathbf{A}_{H|G})\vec{\theta}' = \mathbf{D}_H\vec{t}$ for a vector $\vec{t}$ such that $t_i = -a_{jk}(\theta'_j - \theta'_k)$ if line i is failed, and $t_i = 0$ otherwise. Hence, $\mathbf{A}_{\bar{H}|G}(\vec{\theta} - \vec{\theta}') = 0$ which can be used to recover $\vec{\theta}'_H$ if $\mathbf{A}_{\bar{H}|H}$ is full-rank. And $\mathbf{A}_{H|G}(\vec{\theta} - \vec{\theta}') = \mathbf{D}_H\vec{t}$ which have a unique solution $\vec{t}$ such that $\text{supp}(\vec{t}) = F$ if $\mathbf{D}_H$ is full-rank. The idea can be extended to the case when the measurements are noisy and when $\mathbf{D}_H$ and $\mathbf{A}_{\bar{H}|H}$ are not full-rank as in (7).

In particular, when the noise is low, it is shown that (7) can detect the failed lines very well [5]. In particular, in noise-free case, when H is acyclic (i.e., $\mathbf{D}_H$ is full-rank) and there is a matching between the nodes in H and $\bar{H}$ that covers H (i.e., $\mathbf{A}_{\bar{H}|H}$ is full-rank), the feasible set of (7) contains only a single unique solution which $\vec{z} = \vec{\theta}'_H$ and $\text{supp}(\vec{t}) = F$ for any set of line failures. If $\mathbf{D}_H$ and $\mathbf{A}_{\bar{H}|H}$ are not full-rank, the feasible set of (7) still contains the solution that $\vec{z} = \vec{\theta}'_H$ and $\text{supp}(\vec{t}) = F$ but it may not be its optimal solution. General topological constraints on H and F for which the optimal solution of (7) is unique, $\vec{z} = \vec{\theta}'_H$, and $\text{supp}(\vec{t}) = F$ are provided in [5].

As the noise level increases, the detection based on the solution to (7) produces false negatives and false positives. The main challenge here is to determine the weight λ that makes the solution space small enough to contain only the actual solution. In Section VI, we numerically investigate the limitations of this method as well.

The main goal of this paper is to extend the idea of the optimization (7) and make it more robust to noisy measurements. In the next section, we demonstrate why Bayesian regression is a suitable approach for this purpose.

V. Bayesian Regression for Failed Lines Detection

In order to provide a robust method for failed lines detection, we use Bayesian regression, which is more suitable for dealing with uncertainties.

To see how Bayesian regression can be used here, recall that the key in (7) is that once it is solved, the failed lines can be detected by looking for the nonzero elements of vector $\vec{t}$. Now if the phase angle measurements are noisy, all we need to compute is the probability that each element of vector $\vec{t}$ is nonzero in the optimal solution. And this is where Bayesian regression can be used. Recall the regression notation from Subsection III-C and define

$$\mathbf{X} = [\vec{x_1}, \dots, \vec{x_n}] := \begin{bmatrix} \mathbf{A}_{H|H} & \mathbf{D}_H \\ \mathbf{A}_{\bar{H}|H} & 0 \end{bmatrix}^T, \qquad (8)$$

$$\mathbf{Y}^{(i,j)} = \begin{bmatrix} y_1^{(i,j)} \\ \vdots \\ y_n^{(i,j)} \end{bmatrix}^T := \begin{bmatrix} \mathbf{A}_{H|H}\vec{\theta}^{(i)}_H + \mathbf{A}_{H|\bar{H}}(\vec{\theta}^{(i)}_{\bar{H}} - \vec{\theta}'^{(j)}_{\bar{H}}) \\ \mathbf{A}_{\bar{H}|H}\vec{\theta}^{(i)}_H + \mathbf{A}_{\bar{H}|\bar{H}}(\vec{\theta}^{(i)}_{\bar{H}} - \vec{\theta}'^{(j)}_{\bar{H}}) \end{bmatrix}^T, \qquad (9)$$

where $\mathbf{Y}^{(i,j)}$ is the output vector using the i^{th} phase angle measurement before the attack and j^{th} phase angle measurement after the attack. In regression, we want to find the vector $\vec{w}$ such that for any $1 \le k \le |V|$ and any $1 \le i,j \le T$: $y_k^{(i,j)} \approx \vec{w}^T\vec{x_k}$. Notice that in this setting, the first $|V_H|$ elements of $\vec{w}$ are associated with vector $\vec{z}$ in (7) and the last $|E_H|$ elements are associated with vector $\vec{t}$.

To find a posterior distribution on $\vec{w}$, we use the Bayesian regression model with ARD as summarized in Subsection III-C. As mentioned in Subsection III-C, the prior distribution on $\vec{w}$ and the structure in this model is such that it promotes sparsity in $\vec{w}$ which can play the role of $\|\vec{t}\|_1$ minimizer in (7). Hence, this model can be considered as a Bayesian approach for solving (7).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems

Algorithm 1: PRObabilistic line failure detection using BayEsian Regression (PROBER)

Input: G, H, $\mathbf{A}$, $\vec{\theta}^{(1)}, \vec{\theta}^{(2)}, \ldots, \vec{\theta}^{(T)}$, $\vec{\theta}'^{(1)}_{\bar{H}}, \vec{\theta}'^{(2)}_{\bar{H}}, \ldots, \vec{\theta}'^{(T)}_{\bar{H}}$, and γ

1: Set $\mathbf{X}$ as in (8)
2: For each $1 \leq i, j \leq T$, set $\mathbf{Y}^{(i,j)}$ as in (9)
3: Set prior distributions on $\vec{w}, \tau, \vec{\alpha}$ as in (3) and (4)
4: Compute posterior distributions $Q(\vec{w}, \tau), Q(\vec{\alpha})$ as in (5) such that $\mathrm{P}(y_k^{(i,j)} | \vec{x_k}, \vec{w}, \tau) = \mathcal{N}(y_k^{(i,j)} | \vec{w}^T \vec{x_k}, \tau^{-1})$ for all $1 \leq i, j \leq T$ and $1 \leq k \leq |V|$
5: Set $\vec{t} = [t_1, \ldots, t_{|E_H|}]^T$ to be last $|E_H|$ entries of vector $\vec{w}$
6: Set $\vec{\mu}$ to be the last $|E_H|$ entries of the posterior mean vector $\vec{w}_n$
7: Set $\mathbf{\Sigma}$ to be the submatrix of matrix $\mathbf{V}_n$ associated with its last $|E_H|$ rows and columns
8: For each line e_i in H compute $\mathrm{P}(e_i \in F) \approx \frac{1}{2} - \frac{1}{2}\mathrm{erf}\Big(\frac{\mu_i - \gamma}{\sqrt{2b_n/a_n \Sigma_{ii}}}\Big)$
9: **return** $Q(\vec{w}, \tau), Q(\vec{\alpha})$ and $\mathrm{P}(e_i \in F)$ for all the lines in H

Once the posterior distribution on $\vec{w}$ is computed as in (5), it can be used to estimate the phase angles of the nodes inside the attacked area and compute the probability that each line is failed. Since the first $|V_H|$ elements of posterior $\vec{w}$ provide a distribution on the phase angles of the nodes inside the attacked area, the first $|V_H|$ elements of posterior mean vector $\vec{w}_n$ provide the maximum likelihood estimate for the phase angles of the nodes inside the attacked area *under this model*.

Also define $\vec{t} = [t_1, \ldots, t_{|E_H|}]^T$ to be last $|E_H|$ entries of vector $\vec{w}$, $\vec{\mu}$ to be the last $|E_H|$ entries of the posterior mean vector $\vec{w}_n$, and $\mathbf{\Sigma}$ to be the submatrix of matrix $\mathbf{V}_n$ associated with its last $|E_H|$ rows and columns. Then according to the model, $\mathrm{P}(\vec{t}|\tau) = \mathcal{N}(\vec{\mu}, \tau^{-1}\mathbf{\Sigma})$. Moreover, τ can be further approximated by its posterior mean a_n/b_n to obtain $\mathrm{P}(\vec{t}) \approx \mathcal{N}(\vec{\mu}, b_n/a_n\mathbf{\Sigma})$. Using this, we can approximate the probability that each line is failed by computing the probability that $|t_i|$ is greater than $\gamma > 0$. Hence, since $\mathbf{\Sigma}$ is diagonal, the probability that line e_i is failed according to this approach is

$$\mathrm{P}(e_i \in F) \approx \mathrm{P}(|t_i| \geq \gamma) \approx \frac{1}{2} - \frac{1}{2}\mathrm{erf}\Big(\frac{\mu_i - \gamma}{\sqrt{2b_n/a_n \Sigma_{ii}}}\Big), \quad (10)$$

where $\mathrm{erf}(.)$ is the Gaussian error function. We call this process PRObabilistic line failure detection using BayEsian Regression (PROBER) Algorithm (summarized in Algorithm 1). The threshold γ indicates the sensitivity of the detection algorithm to noise. If it is too small, then the PROBER Algorithm may assign a positive probability of failure to many lines. On the other hand, if γ is too large, then it only detects *significant* line failures (i.e., lines that would have carried large amount of power flows if they were not failed—as described in Section IV). In general, γ should be selected such that it is larger than the noise variance but it is smaller than the least amount of power flow on the lines. It is obvious that when the noise level is high, it is impossible for the algorithm to distinguish between a failure in an insignificant line and the noise.

Once the probabilities that each line in H is failed are computed using the PROBER Algorithm, there are two ways to turn these probabilities into hard decisions:

(i) Consider a probability threshold p and detect all lines with the probability of being failed greater than p as failed and use the probability values as the confidence in the detection. We refer to this method in Section VI for simplicity as PROBER.
(ii) Utilize the probabilities to sort the lines and only use the top k lines as the set of lines that are most probable to be failed in the search space of the BFS (6). This significantly reduces the running time of the BFS method by keeping the size of the search space constant as the size of the attacked area increases. We refer to this method in Section VI as PROBER-BFS.

In the next section, we compare the performance of these two methods in detecting line failures with previously proposed methods described in Section IV.

VI. Numerical Results

In this section, we numerically evaluate the performance of the PROBER Algorithm by first comparing it with that of the other methods described in Section IV, and then as both the noise-level and size of the attacked area increase.

A. Comparison between different failed line detection methods

In this subsection, we compare the PROBER and PROBER-BFS methods for failed lines detection, introduced in the previous section, with the methods described in Section IV.

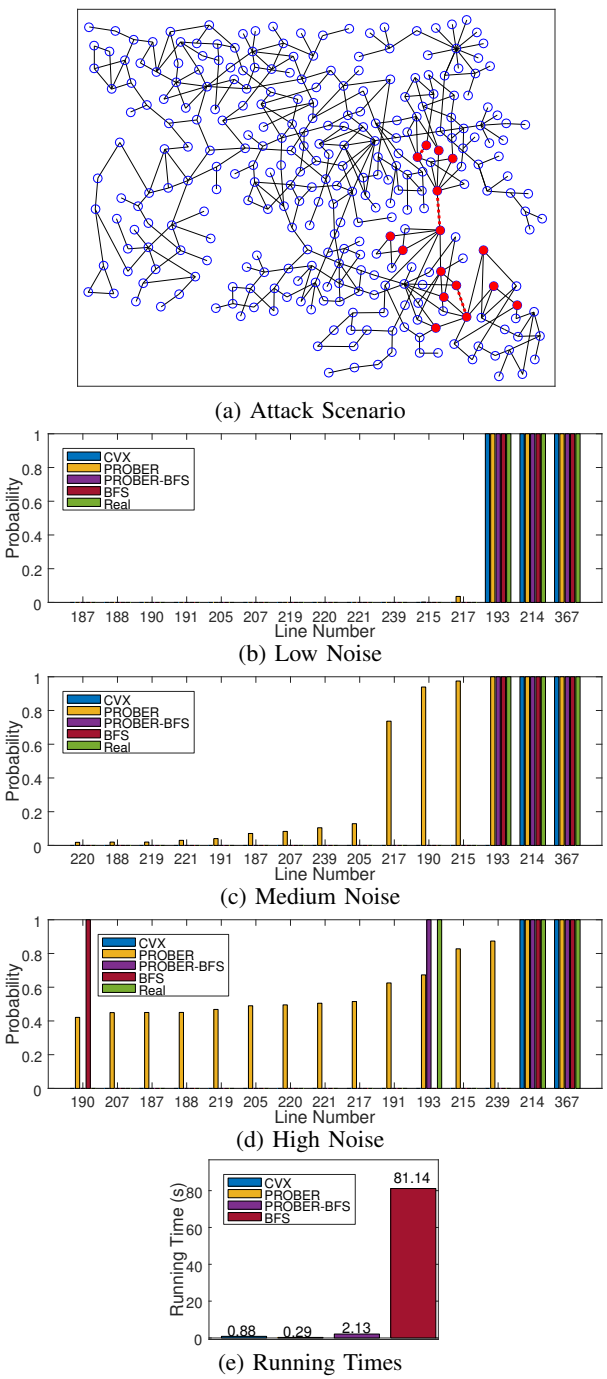
In the PROBER Algorithm, we select $\gamma = 0.1$, which means that we are interested in detecting the entries of $\vec{t}$ that can get a substantial value.[3] We assume $T = 10$ noisy measurements before and after the attack. In the BFS optimization (6), we limit the search to the set of failed lines of size at most 5. In the convex optimization (7), we set $\lambda = 1000$ and refer to the solution obtained by this method by *CVX*. Finally, we set $p = 0.9999$ as the threshold probability in PROBER, and $k = 7$ in PROBER-BFS.

Fig. 3 depicts an attack example and the detected failed lines using different methods. In this example, as depicted in Fig. 3(a), the phase angle measurements from the attacked area, shown by red filled nodes, are blocked and lines 193, 214, and 367, shown by the red dashed lines, are failed. For the attacked area H here, matrices $\mathbf{A}_{\bar{H}|H}$ and $\mathbf{D}_H$ are both full-rank. Hence, as mentioned in Section IV-B, the solution to the convex optimization (7) could exactly detect the line failures, if the measurements were noise-free. Here, however, we compare the results obtained by solving (7) and the PROBER Algorithm in the presence of measurement noise.

Figs. 3(b), 3(c), and 3(d) compare the probability that each line is failed, computed by different methods, under low, medium, and high noise-levels. The SNR in the low noise scenario is considered to be around $42.8dB$ which is equal to the noise level in the real-world PMU data as characterized and computed by Brown et al. [7]. Accordingly, the noise levels in the medium and high noise scenarios are 10 and 100 times stronger, respectively. Each color bar indicates the probability

[3]The threshold γ should be selected based on the active power nominal value. Here, based on the IEEE test system, the nominal value is $100MW$ which makes the 0.1 threshold in per unit, equal to $10MW$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems



(a) Attack Scenario

(b) Low Noise

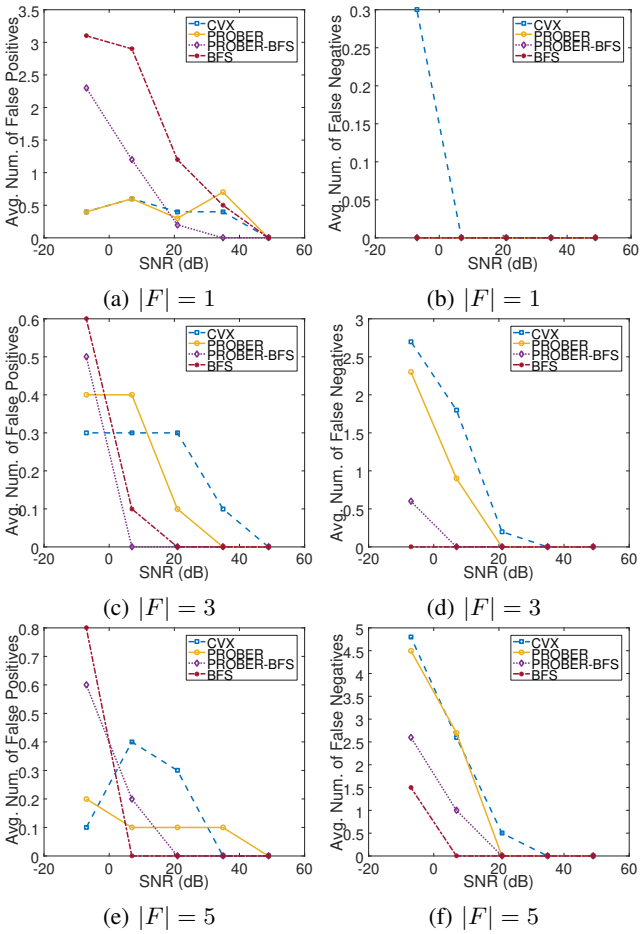
(c) Medium Noise

(d) High Noise

(e) Running Times

Fig. 3: An example of an attack and the failed lines detection using different methods. (a) The attacked area H shown by red filled nodes and three failed lines shown by red dotted lines in the IEEE 300-bus system, (b) the probability that each line is failed as computed by different methods when the noise is low (SNR $= 42.8dB$), (c) the probability that each line is failed as computed by different methods when the noise is medium (SNR $= 22.8dB$), (d) the probability that each line is failed as computed by different methods when the noise is high (SNR $= 3.7dB$), and (e) running times.

that a particular method assigns to a line. Since all the methods except PROBER make hard decisions, the probabilities they provide are either 0 or 1. Therefore, in these figures, the lines are ordered based on the probabilities obtained by the PROBER Algorithm.



(a) $|F| = 1$ (b) $|F| = 1$

(c) $|F| = 3$ (d) $|F| = 3$

(e) $|F| = 5$ (f) $|F| = 5$

Fig. 4: The average number of false positives and negatives using different detection methods under different SNR values and different number of failed lines.

When the noise-level is low, as can be seen in Fig. 3(b), all the four methods can detect the failed lines accurately. Also, when the noise is medium, all the methods except CVX (which misses line 193) can detect all the failed lines. When the noise-level is high, however, all the methods except the PROBER-BFS result in a false positive or a false negative, or both in their detections. For example, both CVX and PROBER with $p = 0.9999$ miss the failed line 193. The interesting result, however, is that BFS method incorrectly detects line 190 instead of line 193 as failed. On the other hand, PROBER-BFS by only searching over lines with a high probability of being failed avoids any false positives or negatives and detects all the failed lines correctly.

Fig. 3(e) compares the running times of the failed line detection methods. As mentioned in Section IV and can be seen in this figure, the BFS method is significantly slower than other methods which make it unsuitable for practical purposes. While CVX method is very fast, it may provide less accuracy as the noise-level increases. The PROBER Algorithm is the fastest among the four and by combining it with the BFS, as in PROBER-BFS, it hits a sweet spot in accuracy and efficiency.

To further investigate these observations, we performed more simulations under different attack scenarios (e.g., different number of failed lines and different noise-levels). The results are presented in Fig. 4. As can be seen, the results are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2019.2910396, IEEE Transactions on Power Systems

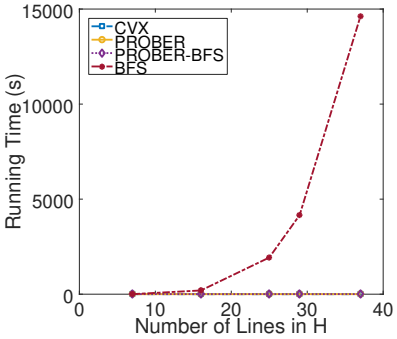


Fig. 5: The relationship between the running times of different methods and the number of lines in H.

consistent with the results that we presented in Fig. 3. While all the methods perform very well when the noise-level is low, the BFS and PROBER-BFS perform better than the CVX and the PROBER, especially in the number of false negatives. Hence, if the grid operator is confident about the low level of measurement-noise in the system, he can select either of the CVX or PROBER-BFS Algorithms to detect line failures. However, for high measurement-noise levels, PROBER-BFS is more accurate.

Finally, to show the main disadvantage of the BFS method more clearly, in Fig. 5, we compared the running times of the four methods as the number of lines in the attacked area increases. As can be seen, while the running times of the CVX, PROBER, and PROBER-BFS remain almost constant as the number of lines increases, the BFS method significantly slows down. Hence, Figs. 4 and 5 confirm the results we observed in Fig. 3 that *the PROBER-BFS provides an accuracy level similar to BFS in significantly lower running time.*

B. Performance evaluation of the PROBER Algorithm as the size of the attacked area and noise-level increase

In this subsection, we evaluate the performance of the PROBER Algorithm as the size of the attacked area and the noise-level increase. In particular, we want to determine how much it can help to reduce the search space for the BFS Algorithm. For this reason, we compute where the actual failed line is ranked between the most probable line failures by the PROBER Algorithm for any single line failure (i.e., *failed line probability rank*). For example, if the actual failed line is detected as the 3^{rd} most probable line to be failed by the PROBER Algorithm, then the line probability rank of that failed line is 3.

We consider five nested attacked areas $H_1, \ldots, H_5$, such that each attacked area H_i contains H_{i-1} plus some extra nodes as depicted in Fig. 6. In particular, H_i contains all the nodes in levels 1 to i. As the attacked area grows larger, matrices $\mathbf{A}_{\bar{H}|H}$ and $\mathbf{D}_H$ are no longer full-rank. Hence, as demonstrated in [5], the convex relaxation approach (7) may not be able to exactly detect the line failures in all scenarios (even in the absence of measurement noise). However, since the PROBER Algorithm provides the probabilities that each line is failed, as long as the line probability rank of the actual failed line is low enough, it will be detected by the BFS

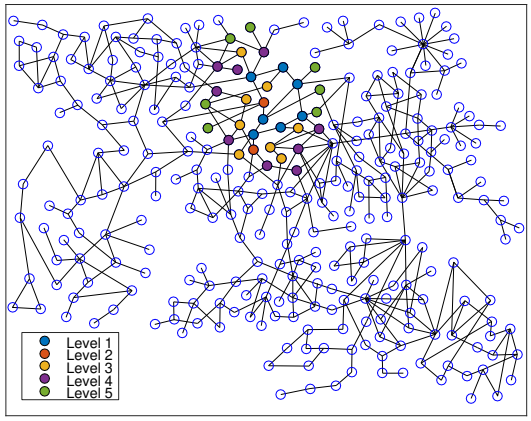


Fig. 6: Nodes that are added to the attacked area at each level in the IEEE 300-bus system.

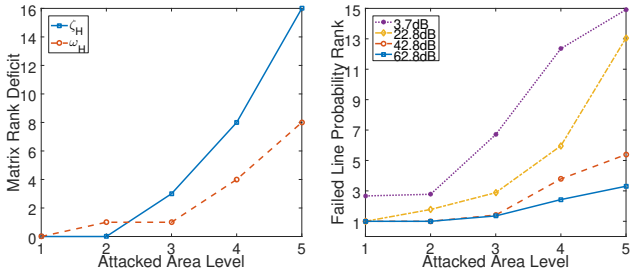


Fig. 7: (a) Matrices $\mathbf{A}_{\bar{H}|H}$ and $\mathbf{D}_H$ rank deficit from being full-rank for the attacked area H at each level. (b) Actual failed line average probability rank obtained by the PROBER Algorithm as the size of the attacked area and the noise-level increase.

Algorithm that is limited its search to only the lines with a high probability of being failed (i.e., PROPER-BFS).

In order to determine the difficulty of detecting line failures based on the rank of $\mathbf{A}_{\bar{H}|H}$ and $\mathbf{D}_H$, we define the matrix rank deficit of these two matrices as follows:

$$\zeta_H := n_H - \text{rank}(\mathbf{A}_{\bar{H}|H}),$$
$$\omega_H := m_H - \text{rank}(\mathbf{D}_H),$$

in which n_H and m_H denote the number of nodes and lines in the attacked area H, respectively. The rank deficits ζ_H and ω_H for attacked areas $H_1, \ldots, H_5$ are shown in Fig. 7(a). As can be seen, as the attacked area grows larger, both ζ_H and ω_H increase sharply. Therefore, we expect that it gets harder to detect line failures as the size of the attacked area increases at each level from H_1 to H_5.

To evaluate PROBER Algorithm performance, we considered all single line failures in $H_1, \ldots, H_5$ and computed the failed line probability rank using PROBER Algorithm under four SNR values ($3.7dB, 22.8dB, 42.8dB$, and $62.8dB$) in each case. The average failed line probability ranks for each attacked area and under each SNR value is shown in Fig. 7(b). As can be seen, when the SNR is high enough (e.g., $42.8dB$-$62.8dB$), the actual failed line is ranked among the top 5 most probable lines to be failed by the PROBER Algorithm, even as the attacked area grows larger. When the SNR is very low, the PROBER Algorithm still ranks the failed line among the

top 3 most probable lines to be failed as long as the attacked area is not too larger. Moreover, even in the worst studied scenario (i.e., the attacked area is H_5 and $SNR = 3.7dB$), the PROBER Algorithm still ranks the actual failed line among the top 15 most probable lines to be failed (out of 39 total number of lines).

Overall, the numerical results provided in this subsection demonstrate that the PROBER Algorithm can be used to reduce the search space of the BFS Algorithm (and therefore make it tractable) even when the noise-level is high and the attacked area is large. Therefore, by using Bayesian regression for probabilistically solving the convex optimization (7), its applicability for detecting line failures significantly increases (by performing BFS only on the lines with high probability of being failed) even when the rank deficits ζ_H and ω_H are quite high.

VII. Conclusion

In this work, we developed a new PRObabilistic line failure detection using BayEsian Regression (PROBER) Algorithm for detecting line failures following a cyber-physical attack. We numerically showed that the method that uses the PROPER Algorithm to find the set of lines with the highest probability of being failed and performs a limited Brute Force Search (BFS) on that small set of lines, hits a sweet spot in terms of accuracy and efficiency in detecting line failures after an attack.

Our work is one of the first works in applying Bayesian data analysis tools for power grid state estimation. We believe that these tools have a vast potential and can be applied for power grid state estimation not only during an attack but also during the daily operation of the grid. Providing a general framework for power grid state estimation using Bayesian regression is part of our future work.

Finally, in this work, we used the linearized DC power flows to describe the state of the grid. Although the considered high noise scenarios in simulations (with about 100 times stronger noise levels than normal) can capture the inaccuracy of the DC approximation of the AC power flows [47], the presented Bayesian regression approach can also directly be applied to the AC power flow equations using the extension of optimization (7) to the AC power flows provided in a recent work by Soltan et al. [48]. Moreover, they can be extended to the case where there are false data injections instead of data blocking, using recently introduced methods [43]. Exploring these directions under noisy measurements and fluctuations in the supply/demand values are also part of our future work.

References

[1] S. Soltan, M. Prateek, and H. V. Poor, "Bayesian regression for robust power grid state estimation following a cyber-physical attack," in *Proc. IEEE PES-GM'18*, Aug. 2018.

[2] "Analysis of the cyber attack on the Ukrainian power grid," Mar. 2016, http://www.nerc.com/pa/CI/ESISAC/Documents/E-ISAC_SANS_Ukraine_DUC_18Mar2016.pdf.

[3] N. S. Malik and R. Collins, "The cyberattack that crippled gas pipelines is now hitting another industry," 2018, https://www.bloomberg.com/news/articles/2018-04-04/cyberattack-bleeds-into-utility-space-as-duke-sees-billing-delay. Accessed: June. 2018.

[4] R. Smith, "Russian hackers reach U.S. utility control rooms, homeland security officials say," 2018, https://www.wsj.com/articles/russian-hackers-reach-u-s-utility-control-rooms-homeland-security-officials-say-1532388110. Accessed: June. 2018.

[5] S. Soltan, M. Yannakakis, and G. Zussman, "Power grid state estimation following a joint cyber and physical attack," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 1, pp. 499–512, Mar. 2018.

[6] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[7] M. Brown, M. Biswal, S. Brahma, S. J. Ranade, and H. Cao, "Characterizing and quantifying noise in PMU data," in *Proc. IEEE PES-GM'16*, July 2016.

[8] J. Drugowitsch, "Variational bayesian inference for linear and logistic regression," *arXiv preprint arXiv:1310.5438v3*, 2017.

[9] J. E. Tate and T. J. Overbye, "Line outage detection using phasor angle measurements," *IEEE Trans. Power Syst.*, vol. 23, no. 4, pp. 1644–1652, Nov. 2008.

[10] Y. Zhao, A. Goldsmith, and H. V. Poor, "On PMU location selection for line outage detection in wide-area transmission networks," in *Proc. IEEE PES-GM'12*, July 2012.

[11] G. Rovatsos, X. Jiang, A. D. Domínguez-García, and V. V. Veeravalli, "Statistical power system line outage detection under transient dynamics," *IEEE Trans. Signal Process.*, vol. 65, no. 11, pp. 2787–2797, June 2017.

[12] A. Pinar, J. Meza, V. Donde, and B. Lesieutre, "Optimization strategies for the vulnerability analysis of the electric power grid," *SIAM J. Optimiz.*, vol. 20, no. 4, pp. 1786–1810, Feb. 2010.

[13] T. Kim, S. J. Wright, D. Bienstock, and S. Harnett, "Analyzing vulnerability of power systems with continuous optimization formulations," *IEEE Trans. Net. Sci. Eng.*, vol. 3, no. 3, pp. 132–146, July 2016.

[14] I. Dobson, B. Carreras, V. Lynch, and D. Newman, "Complex systems analysis of series of blackouts: cascading failure, critical points, and self-organization," *Chaos*, vol. 17, no. 2, p. 026103, June 2007.

[15] D. Bienstock, *Electrical Transmission System Cascades and Vulnerability: An Operations Research Viewpoint*. SIAM, 2016.

[16] A. Teixeira, S. Amin, H. Sandberg, K. H. Johansson, and S. S. Sastry, "Cyber security analysis of state estimators in electric power systems," in *Proc. IEEE CDC'10*, 2010.

[17] J. Kim and L. Tong, "On topology attack of a smart grid: undetectable attacks and countermeasures," *IEEE J. Sel. Areas Commun*, vol. 31, no. 7, pp. 1294–1305, July 2013.

[18] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Trans. Inf. Syst. Secur.*, vol. 14, no. 1, p. 13, May 2011.

[19] G. Dán and H. Sandberg, "Stealth attacks and protection schemes for state estimators in power systems," in *Proc. IEEE SmartGridComm'10*, 2010.

[20] O. Vukovic, K. C. Sou, G. Dán, and H. Sandberg, "Network-layer protection schemes against stealth attacks on state estimators in power systems," in *Proc. IEEE SmartGridComm'11*, 2011.

[21] S. Li, Y. Yılmaz, and X. Wang, "Quickest detection of false data injection attack in wide-area smart grids," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2725–2735, Nov. 2015.

[22] J. Kim, L. Tong, and R. J. Thomas, "Subspace methods for data attack on state estimation: A data driven approach." *IEEE Trans. Signal Process.*, vol. 63, no. 5, pp. 1102–1114, Mar. 2015.

[23] R. B. Bobba, K. M. Rogers, Q. Wang, H. Khurana, K. Nahrstedt, and T. J. Overbye, "Detecting false data injection attacks on dc state estimation," in *Proc. SCS CPSWEEK'10*, vol. 2010, 2010.

[24] R. Deng, P. Zhuang, and H. Liang, "False data injection attacks against state estimation in power distribution systems," *to appear in IEEE Trans. Smart Grid*, 2018.

[25] J. E. Tate and T. J. Overbye, "Double line outage detection using phasor angle measurements," in *Proc. IEEE PES-GM'09*, July 2009.

[26] M. Garcia, T. Catanach, S. Vander Wiel, R. Bent, and E. Lawrence, "Line outage localization using phasor measurement data in transient state," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 3019–3027, July 2016.

[27] N. M. Manousakis, G. N. Korres, and P. S. Georgilakis, "Taxonomy of PMU placement methodologies," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 1070–1077, May 2012.

[28] K. Khandeparkar, P. Patre, S. Jain, K. Ramamritham, and R. Gupta, "Efficient PMU data dissemination in smart grid," in *Proc. ACM e-Energy'14 (poster description)*, June 2014.

[29] Y. Zhao, J. Chen, A. Goldsmith, and H. V. Poor, "Identification of outages in power systems with uncertain states and optimal sensor

locations," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 6, pp. 1140–1153, Dec. 2014.
[30] H. Zhu and Y. Shi, "Phasor measurement unit placement for identifying power line outages in wide-area transmission system monitoring," in *HICSS'14*, 2014, pp. 2483–2492.
[31] H. Zhu and G. B. Giannakis, "Sparse overcomplete representations for efficient identification of power line outages," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 2215–2224, Nov. 2012.
[32] Y. Zhao, J. Chen, and H. V. Poor, "A learning-to-infer method for real-time power grid topology identification," *arXiv preprint arXiv:1710.07818*, 2017.
[33] A. Ashok and M. Govindarasu, "Cyber attacks on power system state estimation through topology errors," in *Proc. IEEE PES-GM'12*, 2012.
[34] S. Wang, W. Ren, and U. M. Al-Saggaf, "Effects of switching network topologies on stealthy false data injection attacks against state estimation in power networks," *IEEE Systems Journal*, vol. 11, no. 4, pp. 2640–2651, Dec. 2017.
[35] Y. Zhou, J. Cisneros-Saldana, and L. Xie, "False analog data injection attack towards topology errors: Formulation and feasibility analysis," in *Proc. IEEE PES-GM'18*, 2018.
[36] S. Soltan, M. Yannakakis, and G. Zussman, "Joint cyber and physical attacks on power grids: Graph theoretical approaches for information recovery," in *Proc. ACM SIGMETRICS'15*, June 2015.
[37] J. Zhang and L. Sankar, "Implementation of unobservable state-preserving topology attacks," in *Proc. IEEE NAPS'15*, 2015.
[38] Z. Li, M. Shahidehpour, A. Alabdulwahab, and A. Abusorrah, "Bilevel model for analyzing coordinated cyber-physical attacks on power systems," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2260–2272, Sept. 2016.
[39] R. Deng, P. Zhuang, and H. Liang, "CCPA: Coordinated cyber-physical attacks and countermeasures in smart grid," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2420–2430, Sept. 2017.
[40] J. Zhang and L. Sankar, "Physical system consequences of unobservable state-and-topology cyber-physical attacks," *IEEE Trans. Smart Grid*, vol. 7, no. 4, July 2016.
[41] D. Bienstock and M. Escobar, "Computing undetectable attacks on power grids," *ACM SIGMETRICS Performance Evaluation Review*, vol. 45, no. 2, pp. 115–118, 2017.
[42] X. Liu and Z. Li, "Local topology attacks in smart grids," *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 2617–2626, Nov. 2017.
[43] S. Soltan, M. Yannakakis, and G. Zussman, "REACT to cyber attacks on power grids," *to appear in IEEE Trans. Netw. Sci. Eng.*, 2018.
[44] H.-M. Chung, W.-T. Li, C. Yuen, W.-H. Chung, Y. Zhang, and C.-K. Wen, "Local cyber-physical attack for masking line outage and topology attack in smart grid," *to appear in IEEE Trans. Smart Grid*, 2018.
[45] D. Bienstock and M. Escobar, "Stochastic defense against complex grid attacks," *arXiv preprint arXiv:1807.06707*, 2018.
[46] M. S. Thomas and J. D. McDonald, *Power system SCADA and smart grids*. CRC press, 2015.
[47] S. Soltan and G. Zussman, "Power grid state estimation after a cyber-physical attack under the AC power flow model," in *Proc. IEEE PES-GM'17*, July 2017.
[48] ——, "EXPOSE the line failures following a cyber-physical attack on the power grid," *to appear in IEEE Trans. Control Netw. Syst.*, 2018.

Saleh Soltan (M'15) is a postdoctoral research associate in the department of Electrical Engineering at Princeton University. In 2017, he obtained the Ph.D. degree in Electrical Engineering from Columbia University. He received B.S. degrees in Electrical Engineering and Mathematics (double major) from Sharif University of Technology, Iran in 2011 and the M.S. degree in Electrical Engineering from Columbia University in 2012. He is the Gold Medalist of the 23rd National Mathematics Olympiad in Iran in 2005 and the recipient of Columbia University Electrical Engineering Armstrong Memorial Award in 2012 and Jury Award in 2018.

Prateek Mittal is an Associate Professor in the Department of Electrical Engineering at Princeton University. He obtained his Ph.D. from University of Illinois at Urbana-Champaign in 2012. He is the recipient of the NSF CAREER award (2016), ARO YIP Award (2018), ONR YIP award (2018), M.E. Van Valkenburg award, Google Faculty Research Award (2016, 2017), Cisco Faculty research award (2016), Intel Faculty research award (2016, 2017), and IBM Faculty award (2017). He was awarded Princeton University's E. Lawrence Keyes Award for outstanding research and teaching, and is the recipient of multiple outstanding paper awards including ACM CCS and ACM ASIACCS.

H. Vincent Poor (S'72, M'77, SM'82, F'87) received the Ph.D. degree in EECS from Princeton University in 1977. From 1977 until 1990, he was on the faculty of the University of Illinois at Urbana-Champaign. Since 1990 he has been on the faculty at Princeton, where he is currently the Michael Henry Strater University Professor of Electrical Engineering. During 2006 to 2016, he served as Dean of Princeton's School of Engineering and Applied Science. He has also held visiting appointments at several other universities, including most recently at Berkeley and Cambridge. His research interests are in the areas of information theory and signal processing, and their applications in wireless networks, energy systems and related fields. Among his publications in these areas is the forthcoming book Advanced Data Analytics for Power Systems (Cambridge University Press).

Dr. Poor is a member of the National Academy of Engineering and the National Academy of Sciences, and is a foreign member of the Chinese Academy of Sciences, the Royal Society, and other national and international academies. Recent recognition of his work includes the 2019 ASEE Benjamin Garver Lamme Award, and a D.Eng. honoris causa from the University of Waterloo, also awarded in 2019.