# Power Grid State Estimation after a Cyber-Physical Attack under the AC Power Flow Model

Saleh Soltan, Gil Zussman
Department of Electrical Engineering
Columbia University, New York, NY
Email: {saleh,gil}@ee.columbia.edu

***Abstract*—In this paper, we present an algorithm for estimating the state of the power grid following a cyber-physical attack. We assume that an adversary attacks an area by: (i) disconnecting some lines within that area (*failed lines*), and (ii) obstructing the information from within the area to reach the control center. Given the phase angles of the buses outside the attacked area under the AC power flow model (before and after the attack), the algorithm estimates the phase angles of the buses and detects the failed lines inside the attacked area. The novelty of our approach is the transformation of the line failures detection problem, which is combinatorial in nature, to a convex optimization problem. As a result, our algorithm can detect any number of line failures in a running time that is independent of the number of failures and is solely dependent on the size of the network. To the best of our knowledge, this is the first convex relaxation for the problem of line failures detection using phase angle measurements under the AC power flow model. We evaluate the performance of our algorithm in the IEEE 118- and 300-bus systems, and show that it estimates the phase angles of the buses with less that $1\%$ error, and can detect the line failures with $80\%$ accuracy for single, double, and triple line failures.**

## I. Introduction

Power grids are vulnerable to cyber-physical attacks. A cyber-physical attack may sabotage the information flow to the control center as well as cause physical failure by remotely disconnecting some of the lines. The recent cyber-physical attack on the Ukranian grid revealed the devastating consequence of such an attack on power grids [1].

We follow the model that we introduced in [2] to study cyber-physical attacks on power grids. Under this model, we assume that an adversary attacks an area by: (i) disconnecting some lines within that area (*failed lines*), and (ii) obstructing the information from within the area to reach the control center. Given the phase angles of the buses outside the attacked area (before and after the attack), our objective is to estimate the phase angles and detect the failed lines in the attacked area. Unlike [2] which is based on the simplistic DC power flow model, in this paper, we assume that the phase angles are given under the more realistic AC power flow model.

The problem of the line failures detection is combinatorial in nature, since the solution space is the discrete set of all possible line failures. Despite the complexities, we present the Convex OPtimization for Statistical State EStimation (COPSSES) Algorithm to estimate the phase angles of the buses and detect the failed lines inside the attacked area. The algorithm is based on a variation of the convex relaxation that was introduced in [2] for information recovery under the DC power flow model. Here, we adapt a similar idea in the COPSSES Algorithm and show that it can estimates the phase angles and detects the line failures accurately under the AC power flow model. The novelty of our approach is the transformation of the line failures detection problem to a convex optimization problem. Therefore, the COPSSESS Algorithm can be used to detect any number of line failures without affecting its running time.

We evaluate the performance of the COPSSES Algorithm in the IEEE 118- and 300-bus systems, and show that it estimates the phase angles of the buses with less that $1\%$ error, and can detect the line failures with $80\%$ accuracy for single, double, and triple line failures. The algorithm can detect failures beyond triple line failures. However, due to the page limit, we only provide simulations for up to triple line failures.

The considered problem is very similar to the problem of line failure detection using phase angle measurements [3], [4], [5], [6]. Up to two line failures detection, under the DC power flow model, is studied [3], [4]. Since the provided methods in [3], [4] are greedy-based methods that need to search the entire failure space, the running time of these methods grow exponentially as the number of failures increases. Hence, these methods cannot be generalized to detect higher order failures.

The problem of line failure detection in an internal system using the information from an external system was also studied in [6] based (again) on the DC power flow model. The proposed algorithm works for only one and two line failures, since it depends on the sparsity of line failures.

In a recent work [5], a linear multinomial regression model is proposed as a classifier for a single line failure detection using transient PMU data. Due to the time complexity of the learning process for more than a single line failure, this method is impractical for detecting higher order failures. To the best of our knowledge, our work is the first to provide a method for line failures detection under the AC power flow model that can be used to detect any number of line failures.

The remainder of this paper is organized as follows. Section II describes the model and provides definitions. In Section III, we present the COPSSES algorithm. Section IV presents numerical results, and Section V provides concluding

This work was supported in part by the U.S. DOE OE as part of the DOE Grid Modernization Initiative, DARPA RADICS under contract #FA-8750-16-C-0054, and DTRA grant HDTRA1-13-1-0021.

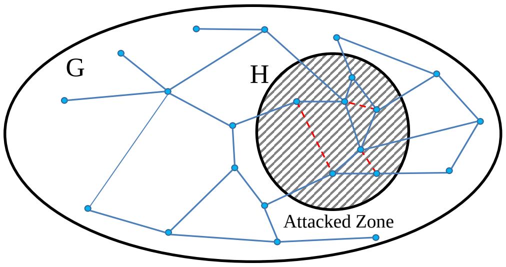


Fig. 1: $G$ is the power grid graph and $H$ is a subgraph of $G$ that represents the attacked zone. An adversary attacks a zone by disconnecting some of its lines (red dashed lines) and disallowing the information from within the zone to reach the control center.

remarks and directions for future work.

## II. Model and Definitions

### A. Power grid model

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 edges corresponding to the buses and transmission lines, respectively (direction of the edges are arbitrary). Each edge $e = (i, j)$ connects two nodes and is characterized by its *impedance*, $r_e + \mathbf{i}x_e$. In the AC power flow model, the status of each node $i$ is represented by its voltage $v_i = |v_i|e^{\mathbf{i}\theta_i}$ in which $|v_i|$ is the voltage magnitude, $\theta_i$ is the phase angle at node $i$, and $\mathbf{i}$ denotes the imaginary unit. In this paper, we consider the standard AC power flow equations (for the details of the equations see [7]).

The (node-edge) *incidence matrix* of $G$ is denoted by $\mathbf{D} \in \{-1, 0, 1\}^{|V| \times |E|}$ and defined as follows,

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

Define also $\mathbf{A} = \mathbf{DYD}^T \in \mathbb{R}^{|V| \times |V|}$,[1] in which $\mathbf{Y} := \text{diag}([1/x_{e_1}, 1/x_{e_2}, \ldots, 1/x_{e_m}])$ is a diagonal matrix with diagonal entries equal to the inverse of the reactance values. We use matrices $\mathbf{A}$ and $\mathbf{D}$ in the state estimation algorithm.

### B. Basic Graph Theoretical Terms

**Matching:** A *matching* in a graph is a set of pairwise nonadjacent edges. If $M$ is a matching, the two ends of each edge of $M$ are said to be *matched* under $M$, and each vertex incident with an edge of $M$ is said to be *covered* by $M$.

**Cycle:** A *cycle* in a graph is a sequence of its distinct nodes $u_1, u_2, \ldots, u_k$ such that for all $i < k$, $\{u_i, u_{i+1}\} \in E$, and also $\{u_k, u_1\} \in E$. A graph with no cycle is called *acyclic*.

### C. Attack Model

We assume that an adversary attacks an area by: (i) disconnecting some lines within that area (*failed lines*), and (ii) obstructing the information (e.g., status of the lines and phase angle measurements) from within the area to reach the control center. We call this area, the *attacked zone*.

[1] Matrix $\mathbf{A}$ is mostly used in the DC power flow equations as *the admittance matrix* and is equivalent to the *weighted Laplacian matrix* of the graph.

We assume that either disconnecting lines within a zone does not make $G$ disconnected, or the control center is aware of the supply/demand values after separation of the grid into islands. In the latter case, without loss of generality, one can assume that the supply/demand values before the attack were as in after the attack by recomputing the power flows before the attack given the new set of supply/demand values.

Fig. 1 shows an example of an attack on the zone represented by $H = (V_H, E_H)$. We denote the set of failed lines in zone $H$ by $F \subseteq E_H$. Upon failure, the failed lines are removed from the graph and the flows are redistributed according to the AC power flows. Our objective is to estimate the phase angles and detect the failed lines inside the attacked zone using the changes in the phase angles of the nodes outside the zone.

Detecting line failures after such an attack is crucial in maintaining the stability of the grid, since it may result in further line overloads and failures if the proper load shedding mechanism is not applied. An effective load shedding requires the exact knowledge of the topology of the grid.

**Notation.** We denote the complement of the zone $H$ by $\bar{H} = G \backslash H$. $\mathbf{D}_H \in \{-1, 0, 1\}^{|V_H| \times |E_H|}$ is the submatrix of $\mathbf{D}$ with rows from $V_H$ and columns from $E_H$. $\vec{\theta}_H$ and $\vec{\theta}_{\bar{H}}$ are the vectors of phase angles of the nodes in $H$ and $\bar{H}$, respectively. If $X, Y$ are two subgraphs of $G$, $\mathbf{A}_{X|Y}$ denotes the submatrix of $\mathbf{A}$ with rows from $V_X$ and columns from $V_Y$. We use the prime symbol ($'$) to denote the values after an attack.

For a column vector $\vec{y}$, $\|\vec{y}\|_1 := \sum_{i=1}^n |y_i|$ is its $l_1$-norm, $\|\vec{y}\|_2 := (\sum_{i=1}^n y_i^2)^{1/2}$ is its $l_2$-norm, and $\text{supp}(\vec{y}) := \{i | y_i \neq 0\}$ is the index set of its nonzero elements.

## III. State Estimation

We can formulate the state estimation problem after a cyber-physical attack as follows: Given the attacked zone $H$, $\vec{\theta}_H$, $\vec{\theta}_{\bar{H}}$, and $\vec{\theta}'_{\bar{H}}$, the objective is to estimate $\vec{\theta}'_H$ and detect $F$. To address this problem, we use and build on the idea that we first introduced in [2]. We proved in [2] that if the phase angles of the nodes are given under the DC power flow equations, there exist vectors $\vec{x} \in \mathbb{R}^{|E_H|}$ and $\vec{\delta}_H \in \mathbb{R}^{|V_H|}$ satisfying following optimization problem for $\epsilon_1 = \epsilon_2 = 0$, and that $\text{supp}(\vec{x}) = \{i | e_i \in F\}$ and $\vec{\delta}_H := \vec{\theta}_H - \vec{\theta}'_H$:

$$\begin{aligned} &\min \|\vec{x}\|_1 \text{ s.t.} \\ &\|\mathbf{D}_H\vec{x} - \mathbf{A}_{H|H}\vec{\delta}_H - \mathbf{A}_{H|\bar{H}}\vec{\delta}_{\bar{H}}\|_2 \leq \epsilon_1 \\ &\|\mathbf{A}_{\bar{H}|H}\vec{\delta}_H + \mathbf{A}_{\bar{H}|\bar{H}}\vec{\delta}_{\bar{H}}\|_2 \leq \epsilon_2. \end{aligned} \tag{1}$$

where $\vec{\delta}_{\bar{H}} := \vec{\theta}_{\bar{H}} - \vec{\theta}'_{\bar{H}}$.

The novelty of this method is that it provides a convex relaxation for the line failures detection problem which is combinatorial in nature. Notice that for $\epsilon_1 = \epsilon_2 = 0$, the optimization problem (1) is a Linear Program (LP). We proved that under several conditions on $H$, the solution to (1) is unique, therefore the relaxation is exact and the state of the grid can be recovered by solving (1) ($\text{supp}(\vec{x})$ gives the failed lines and the phase angles can be computed as $\vec{\theta}'_H = \vec{\theta}_H - \vec{\delta}_H$). In particular, when $H$ is acyclic and there is a matching

**Algorithm 1:** Convex OPtimization for Statistical State EStimation (COPSSES)

**Input:** A connected graph $G$, attacked zone $H$, $\vec{\theta}$, and $\vec{\theta}'_{\bar{H}}$

**for** $\epsilon_1 = s_1$ **to** $t_1$ **do**
    **for** $\epsilon_2 = s_2$ **to** $t_2$ **do**
        Compute $\vec{x}, \vec{\delta}_H$ the solution to (1) by second order cone programming;
        Compute $\vec{\theta}_H - \vec{\delta}_H$ as the estimated phase angles;
        Compute $F = \{e_i | i \in \text{supp}(\vec{x})\}$;
        $\mathcal{F} = [\mathcal{F}, F]$;
        $\Theta'_H = [\Theta'_H, \vec{\theta}_H - \vec{\delta}_H]$
Compute the appearance frequency of each line in $\mathcal{F}$ to form an appearance frequency table $P_F$;
Compute the row mean and variance of $\Theta'_H$ as $\vec{\mu}_H, \vec{\sigma}_H$;
**return** $P_F, \vec{\mu}'_H, \vec{\sigma}'_H$;

between the nodes in $H$ and $\bar{H}$ that covers $H$, the solution to (1) is unique for any set of line failures.[2]

Since the DC power flows only provides an approximation of the phase angles, it is obvious that if the phase angles of the nodes are given under the AC power flows, the optimization problem (1) for $\epsilon_1 = \epsilon_2 = 0$ is no longer feasible. One way to overcome this challenge is to relax the exact conditions by selecting $\epsilon_1, \epsilon_2 > 0$.

It is easy to see that if $\epsilon_1, \epsilon_2 > 0$, the optimization problem (1) becomes a *second-order cone program* that can still be efficiently solved using gradient decent methods.

The only challenge in using (1) for state estimation is that $\epsilon_1$ and $\epsilon_2$ need to be determined. To overcome this challenge, we present the Convex OPtimization for Statistical State EStimation (COPSSES) Algorithm. The idea is to change $\epsilon_i$ from $s_i$ to $t_i$ and compute the solution to (1) for each setup. If $\mathcal{F}$ is an array that contains all the detected line failures for each setup, then the appearance frequency of each line in $\mathcal{F}$ gives a rough probability that the line is failed. $P_F$ denotes the appearance frequency table of the lines in $\mathcal{F}$. Moreover, the computed vector $\vec{\theta}_H - \vec{\delta}_H$ in each iteration is an estimate of the phase angles inside the attacked zone. By computing the mean and variance of all the estimated phase angle vectors in each iteration, it can improve this estimation. We refer to the mean and variance of the estimated phase angles in $H$ as $\vec{\mu}'_H, \vec{\sigma}'_H$. The COPSSES Algorithm is summarized in Algorithm 1.

## IV. Numerical Results

In this section, we evaluate the performance of the COPSSES Algorithm. We use the *CVX* [8] for solving the optimization problem (1) and use MATPOWER [9] to compute $\vec{\theta}$ and $\vec{\theta}'$ under the AC power flow model.

We use the IEEE 118- and 300-bus benchmark systems as the test networks [10] and consider the attacked zones $H_1$ and $H_2$ within these networks, repectively. Fig. 2 shows the topology of the attacked zone $H_1$ within the 118-bus system. Fig. 3 also shows the topology of the attacked zone $H_2$ within the 300-bus system. It is easy to see from Figs. 2 and 3 that $H_1$ and $H_2$ are both acyclic. For both of the attacked zones

[2]We proved in [2] that the solution to (1) is unique under less restricted conditions. However, due to the page limit, we focus on the simplest case.

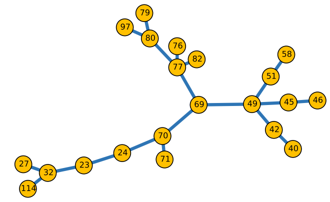


Fig. 2: Topology of the attacked zone in the 118-bus system with 21 nodes and 22 lines (referred to as $H_1$).

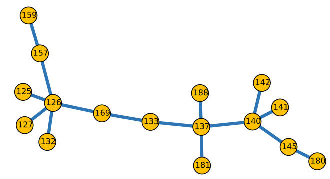


Fig. 3: Topology of the attacked zone in the 300-bus system with 16 nodes and 15 lines (referred to as $H_2$).

there is also a matching between the nodes in $H_i$ and $\bar{H}_i$ that covers $H_i$.

To compute the error in the estimated phase angles, we compute $\|\vec{\mu}'_{H_i} - \vec{\theta}'_{H_i}\|_2 / \|\vec{\theta}'_{H_i}\|_2 \times 100$. Recall that $\vec{\mu}'_{H_i}$ is the vector of average estimated phase angles of the nodes in the attacked zone obtained by the COPSSES Algorithm and $\vec{\theta}'_{H_i}$ is the vector of the actual phase angles of the nodes.

To quantify the performance of the COPSSES Algorithm in detecting the failed lines, we can use the appearance frequency table $P_F$ to detect the most likely failed lines using a threshold value $t$. The idea is that if a line is detected as failed in at least $(t \times 100)\%$ of the settings, then we consider that line as a line that is most likely failed. If $t = 0.5$, then the solution is similar to the *maximum likelihood* set of failures based on $P_F$. We consider $t = 0.2, 0.5, 0.8$.

In this section, in the COPSSES Algorithm, we use $s_1 = 3$, $t_1 = 7$, $s_2 = 1$, and $s_2 = 20$. Therefore, the COPSSES Algorithm estimates the state under 100 different settings for $\epsilon_1$ and $\epsilon_2$. Notice that increasing the intervals $[s_i, t_i]$ increases the accuracy at the expense of the running time.

### A. *Single line failures*

In this subsection, we consider all possible single line failure scenarios in zones $H_1$ and $H_2$ as the failed lines.

*1) 118-bus System:* For all single line failure scenarios, the error in the estimated phase angles using the COPSSES Algorithm is below 1%.

To show the results for the detected line failures, we use a heatmap matrix as in Fig. 4. As can be seen, in most of the cases, the correct line is detected as the most probable failed line by the COPSSES Algorithm. For example:

- For $i = 1$, in 95% of the settings, line 1 is detected as the only failed line. In 5% of the settings, however, no failure is detected.
- For $i = 15$, in 100% of the settings, line 15 is detected as the failed line. However, in 3, 12, 16, 20, 20, 23, and 20% of the times, lines 12, 14, 16, 17, 18, 19, and 20 are also detected as the failed lines.

Fig. 5 also shows the number of false negatives and positives if we use the appearance frequency table $P_F$ and a threshold

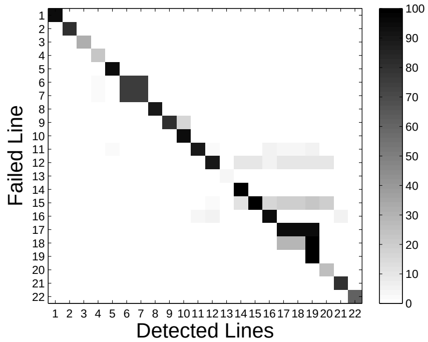


Fig. 4: Detected line failures after all single line failures in zone $H_1$ within the IEEE 118-bus system. The color intensity of each $(i, j)$ square shows the number of times line $j$ is detected as failed when only line $i$ is actually failed.

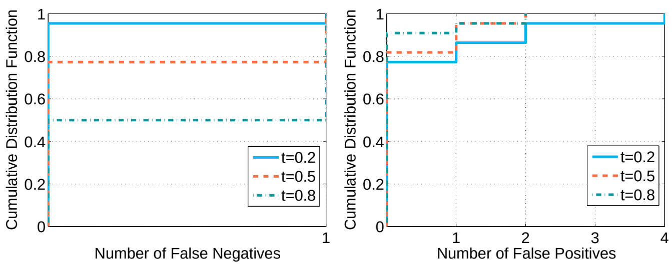


Fig. 5: The CDF of the number of false negatives and positives in detecting single line failures in $H_1$ within the 118-bus system using the COPSSES Algorithm and a threshold value $t$.

value $t$ to detect the most likely failed lines. As can be seen, for $t = 0.5$, for almost 80% of the cases there are no false negatives or false positives. For $t = 0.2$, for almost 95% of the cases, there are no false negatives while for almost 80% of the cases there are no false positives either.

*2) 300-bus System:* In this case, the phase angle estimation is also very accurate. For all single line failure scenarios, the error in the estimated phase angles using the COPSSES Algorithm is below 2%.

Fig. 6 shows the heatmap matrix of the failed line and detected failed lines. As can be seen, in this case also the actual failed line is detected as the most probable failed line by the COPSSES Algorithm, most of the time. For example:

- For $i = 5$, in 95% of the settings, line 5 is detected as the only failed line. In 5% of the settings, however, the optimization problem (1) is infeasible.
- For $i = 7$, in 75% of the settings, line 7 is detected as the failed line. However, in 40 and 13% of the settings, lines 3 and 5 are also detected as the failed lines, respectively. In 25% of the settings, (1) is infeasible.

As in the 118-bus system case, Fig. 7 shows the number of false negatives and positives using the appearance frequency table $P_F$ and a threshold value $t$. As can be seen, for $t = 0.2$ and $t = 0.5$, the number of false positive and negatives is zero for most of the cases.

### B. Double line failures

In this subsection, we consider all double line failure scenarios in zones $H_1$ and $H_2$ as the failed lines.

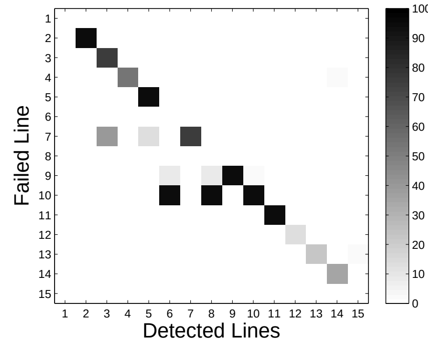


Fig. 6: Detected line failures after all single line failures in zone $H_2$ within the IEEE 300-bus system. The color intensity of each $(i, j)$ square shows the number of times line $j$ is detected as failed when only line $i$ is actually failed.

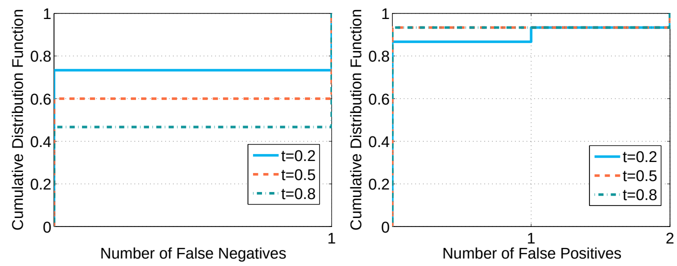


Fig. 7: The CDF of the number of false negatives and positives in detecting single line failures in $H_2$ within the 300-bus system using the COPSSES Algorithm and a threshold value $t$.

*1) 118-bus System:* In this case, as in the single line failure, the phase angle estimation is very accurate. For all double line failure scenarios, the error in the estimated phase angles using the COPSSES Algorithm is below 2%.

Since there are many double line failure cases, we cannot show the failed lines detection results as a matrix heatmap. However, as in the previous subsection, we can show the number of false negatives and positives if we use the appearance frequency table $P_F$ and a threshold value $t$ to detect the most likely failed lines. As can be seen in Fig. 8, for $t = 0.2$ for more than 80% of the cases there is no false negative. Moreover, for more than 80% of the cases there is less than a single false positive line detection.

*2) 300-bus System:* In this case, as in the single line failure scenario and the 118-bus system, the phase angle estimation is very accurate. For all double line failure scenarios, the error in the estimated phase angles is below 2%.

To show the performance of the COPSSES Algorithm in detecting failures, as in the 118-bus case, we compute the number of false negative and positive failure detections using $P_F$ and a threshold value $t$. As can be seen in Fig. 9, in this case also for $t = 0.2$, the detection is relatively accurate. In almost 70% of the cases, there are no false negatives while in 80% of the cases, there is no false positives either.

### C. Triple line failures and beyond

Here, due to the page limit, we only consider up to 3 line failures in our numerical results. However, the COPSSES

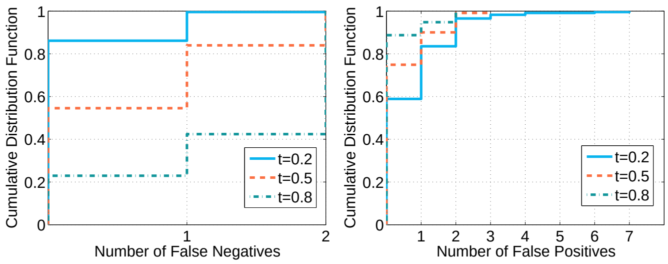


Fig. 8: The CDF of the number of false negatives and positives in detecting double line failures in $H_1$ within the 118-bus system using the COPSSES Algorithm and a threshold value $t$.

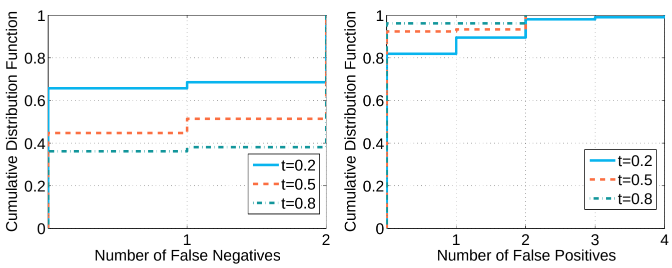


Fig. 9: The CDF of the number of false negatives and positives in detecting double line failures in $H_2$ within the 300-bus system using the COPSSES Algorithm and a threshold value $t$.

Algorithm can be used to estimate the state in the attacked zone for any number of line failures.

In this subsection, we consider 100 randomly sampled triple line failures from all possible triple line failures in $H_1$ and $H_2$ as the failed lines.

*1) 118-bus System:* The phase angle estimation is nearly perfect for the triple line failure scenarios as in the previous cases. The error for the estimated phase angles is less that $1\%$ for all the sampled triple line failure cases.

Fig. 10 shows the number of false negatives and positives, if we use the appearance frequency table $P_F$ and a threshold value $t$ to detect the most likely failed lines. As can be seen, for $t = 0.2$, in $80\%$ of the times there is no false negatives while in $80\%$ of the times there is at most 1 false positive.

*2) 300-bus System:* The phase angle estimation of the COPSSES Algorithm is surprisingly perfect for the triple line failures scenarios in the $H_2$. The error for the estimated phase angles is $0\%$ for all the sampled triple line failure cases.

As in the previous cases, to evaluate the performance of the COPSSES Algorithm in detecting failures, we compute the number of false positives and negatives by selecting different threshold values $t$. As can be seen in Fig. 11, for $t = 0.2$, the Algorithm performs relatively well. In $70\%$ of the cases, there is no false negatives while for more than $80\%$ of the cases, there is no false positives either.

## V. Conclusion

We provided an algorithm to estimate the state of the grid following a cyber-physical attack under the AC power flow model. We studied its performance under different scenarios (single, double, and triple line failures) in IEEE 118- and 300-bus systems and showed that it can estimate the phase angles almost perfectly (with less that $1\%$ error) in these

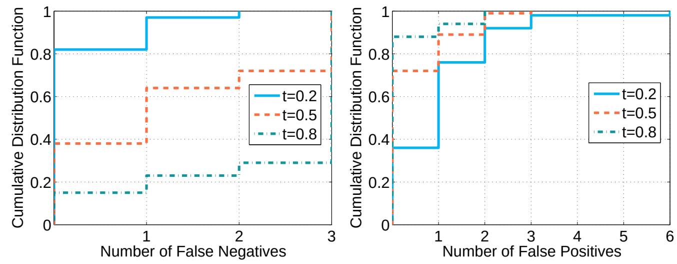


Fig. 10: The CDF of the number of false negatives and positives in detecting triple line failures in $H_1$ within the 118-bus system using the COPSSES Algorithm and a threshold value $t$. 100 randomly sampled triple line failures are considered.

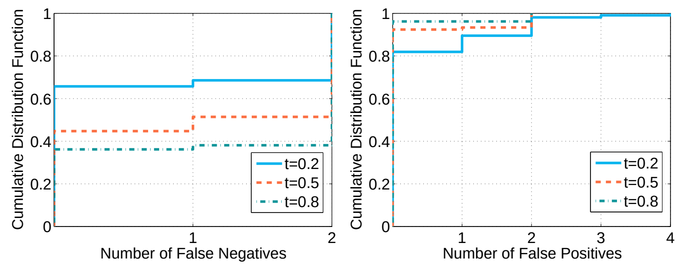


Fig. 11: The CDF of the number of false negatives and positives in detecting triple line failures in $H_2$ within the 300-bus system using the COPSSES Algorithm and a threshold value $t$. 100 randomly sampled triple line failures are considered.

scenarios. Moreover, we showed that our algorithm can detect line failures with less that $20\%$ chance of producing false positives and negatives.

We believe that the COPSSES Algorithm can accurately estimate the state for less constrained attacked zones as well. Moreover, our method can be used in different context such as false data detection. Exploring these directions is part of our future work.

## References

[1] P. Fairley, "Cybersecurity at U.S. utilities due for an upgrade: Tech to detect intrusions into industrial control systems will be mandatory [news]," *IEEE Spectrum*, vol. 53, no. 5, pp. 11–13, May 2016.

[2] S. Soltan, M. Yannakakis, and G. Zussman, "Power grid state estimation following a joint cyber and physical attack," *in IEEE Trans. Control Netw. Syst. (to appear)*, 2016.

[3] 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, 2008.

[4] ——, "Double line outage detection using phasor angle measurements," in *Proc. IEEE PES'09*, July 2009.

[5] 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, 2016.

[6] 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, 2012.

[7] A. R. Bergen and V. Vittal, *Power Systems Analysis*. Prentice-Hall, 1999.

[8] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[9] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "Matpower: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, 2011.

[10] "IEEE benchmark systems," available at http://www.ee.washington.edu/research/pstca/.