Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

On the security of ANN-based AC state estimation in smart grid☆

*Tian Liu, Tao Shu**

Auburn University, 3101 Shelby Center, Auburn, Alabama, 36849, USA

ARTICLE INFO

Article history:
Received 15 November 2020
Revised 20 February 2021
Accepted 9 March 2021
Available online 5 April 2021

Keywords:
Smart grid
AC State estimation
False data injection attack
Adversarial learning
Cyber security

ABSTRACT

With the deployment of new elements in the smart grid, traditional state estimation methods are challenged by growing dynamics and system size. Artificial neural network (ANN) based AC state estimation has been shown to provide faster results than traditional methods. However, researchers have discovered that ANNs could be easily fooled by adversarial examples. In this paper, we initiate a new study of adversarial false data injection attacks against ANN-based state estimation. By injecting a deliberate attack vector into measurements, the attacker can degrade the accuracy of ANN state estimation while remaining undetected. We propose two algorithms to generate the attack vectors, a population-based algorithm (differential evolution or DE) and a gradient-based algorithm (sequential least square quadratic programming or SLSQP). The performance of these algorithms is evaluated through simulations on IEEE 9-bus, 14-bus, and 30-bus systems under various attack scenarios. Simulation results show that DE is more effective than SLSQP on all simulation cases. The attack examples generated by the DE algorithm successfully degrade the ANN state estimation accuracy with high probability (more than 80% in all simulation cases), despite having a small number of compromised meters and low injection strength. We further discuss the potential defense strategy to mitigate such attacks, which provides insights for robustness improvement in future research.

1. Introduction

With the increase in residential and industrial power demand, nowadays a regional or nationwide power outage often leads to catastrophic consequences in the matter of public safety. After the US Northeast Blackout in 2003, the US and Canada reached a consensus to transition to a smart grid system, which would be cleaner and more efficient, reliable, resilient and responsive than a traditional grid. A smart grid is a complex system that integrates a traditional power grid and information technologies to enable inter-networking over power grids. While transferring from the traditional power grid to the smart grid provides many new attractive features such as remote and automatic grid monitoring, control, and pricing, it has also raised serious security challenges by opening up the traditional power system to many potential attacks in cyber space. For example, in the 2015 Ukraine power outage Lee et al. (2016); Liang et al. (2017), the hacker successfully compromised the information systems of three energy distribution companies and caused power disruption to over 225,000 customers lasting from 1 to 6 hours. Since then, cyber attacks on smart grids have caught public's attention and become a realistic and growing concern for governments, vendors, and customers.

☆ A preliminary version of this work has been presented in SecureComm'19, Orland, FL, Oct. 2019.

* Corresponding author.
E-mail addresses: tianliu@auburn.edu (T. Liu), tshu@auburn.edu (T. Shu).

https://doi.org/10.1016/j.cose.2021.102265

One of the key mechanisms in ensuring normal operation of a smart grid is state estimation, which provides the current status of the grid for the control center operators to take corrective action in order to prevent an accident from happening. State estimation aims to compute those system states (the complex voltages at all buses Wood and Wollenberg (1996)) that are not directly measurable, based on the grid's topology and the meter's power usage measurements collected from the *supervisory control and data acquisition* (SCADA) system. Conventionally, state estimation is formulated as a non-linear *weighted least square* (WLS) problem that minimizes the distance between actual measurements and computed measurements from the estimated state. Such methods have several limitations. First of all, solvers to the problem, such as Gauss-Newton, are computationally heavy, sensitive to initial values, and may encounter convergence issues. In addition, the state estimation has to be computed periodically for every set of meter measurements collected in each meter reading cycle (typically a 15-minute period) in order to obtain the current system status. Furthermore, a prior observability analysis is often required to ensure the system is over-determined. This state estimation scheme is further challenged by the growing grid scale and unprecedented system dynamics brought by the increasing deployment of new elements in the smart grid, such as renewable generators, electric vehicles, and dynamic pricing.

In light of the above issues in existing state estimation methods, *artificial neural networks* (ANNs) have received a lot of interest as a new approach to smart grid state estimation, due to mainly two reasons: (1) computation cost can be ignored once the model is trained. In particular, once the ANN state estimation model is trained offline based on historical data or simulated data, such a model can provide accurate estimation online at minimal computation cost, eliminating the need for carrying out observability analysis prior to running the state estimation. (2) ANNs naturally fit into the non-linear nature of the state estimation problem. So far several efforts have been made to adopt ANNs to state estimation. It has been established that the ANN-based state estimation provides results much faster, and the accuracy is comparable or higher than that of conventional state estimations.

While the state estimation plays an important role in ensuring the normal operation of the smart grid, it has been well known that the conventional state estimation methods are vulnerable to *false data injection* (FDI) attacks Liu et al. (2011), which is a data integrity cyber-attack and has been proven to be a real threat to the smart grid system. In particular, an adversary can corrupt the state variable by injecting carefully coordinated false data to meter measurements while evading the bad data detection. The injected false data may result in generation re-dispatch Liang et al. (2016) or trigger a branch outage sequence that involves multiple branches and finally leads to a system failure Che et al. (2019).

Although FDI attacks to conventional state estimation methods have been well understood in the literature, little is known about the FDI attacks against ANN-based state estimation. As the ANN-based state estimation is expected to receive more and more applications for the smart grid in the near future, and because the smart grid is a critical infrastructure of the society, it is necessary to garner a better understanding on the vulnerabilities of this new state estimation method of FDI attacks, so as to identify possible threats and propose countermeasures to eliminate such threats before this new method can be applied in practice on a larger scale. Hence, we can reduce the potential loss and increase the society's confidence in the security feature of the new method.

In contrast to existing FDI attacks that mainly rely on a linear DC power flow model, FDI attacks against an ANN-based state estimation must accommodate a nonlinear AC power flow model as the non-linearity is a fundamental feature of the ANN state estimation. As the ANN becomes a popular technique in the power system, there are several works demonstrating the effectiveness of adversarial attacks on power system applications Chen et al. (2018, 2019); Li et al. (2020). Unfortunately, there has been few work analyzing the vulnerabilities and robustness of the ANN-based state estimation model.

Meanwhile, in the area of image classification, researchers noticed that ANNs can be easily fooled by well-coordinated samples with small perturbations. This discovery has spurred a lot of efforts in exploring the insecurities of ANN by designing adversarial attacks.

In this paper, we are interested in examining whether the above vulnerability of ANN presenting in image classification problem also exists in the state estimation problem in the smart grid. We create an FDI attack customized for the ANN-based state estimation model. This attack can also be used to construct an upper bound on the robustness of the model. Furthermore, we attempt to develop algorithms that can systematically generate contaminated measurements that maximize the ANN-based state estimation error while eluding the detection by bad data detector. By doing so, we intend to establish new understanding on the security vulnerabilities of the latest high-accuracy ANN-based state estimator. To the best of our knowledge, our work is the first in the literature that studies the vulnerabilities and robustness of the ANN-based state estimator by FDI attacks.

Compared with its image classification counterpart, solving our problem faces new and significant challenges. In addition to the obvious difference in the application model, our problem presents the following three novel features in its structure. First of all, our problem has an optimization nature in the sense that we seek the optimal attack vector that maximizes the attack outcomes. In contrast, the goal of the image-classification counterpart is just to find a feasible attack vector. Secondly, the attack model in our problem considers the attacker's access and resource constraints, in which the attacker only has access to and can only manipulate a certain number of meters. The attacker's injection is also subject to physical constraints on the smart grid system. In contrast, the image-classification problem has no such constraints and the attacker is allowed to change any pixel of the image. Lastly, the output of state estimator is a vector of continuous values, whereas that of the image-classification is discrete and covers a limited number of pre-defined cases. Due to these fundamental structural differences, the existing results from image-classification ANN are not directly applicable to our problem, and therefore new solutions need to be developed.

In this paper, we study the robustness of ANN-based state estimators by constructing adversarial FDI attacks. We first

create ANN-based state estimators as our target models, followed by evaluating both model accuracy and bad data rate to ensure the target models are sufficiently strong. We then use the idea of adversarial example to formulate an optimization-based FDI attack. In this model, an attacker attempts to maximize the state estimation error without being reported by the bad data detector, subject to given resource and meter access constraints. Two algorithms are subsequently proposed to solve the above optimization to find the best false data injection vector: *differential evolution* (DE) and *sequential least square quadratic programming* (SLSQP). We extensively evaluate our proposed attacks based on simulations on IEEE 9-bus, 14-bus and 30-bus system models under various scenarios to verify their effectiveness.

The main contributions of our work include the following five-fold:

- In creating the target ANN state estimator for large-scale grid systems (e.g., 30-bus and above), a novel penalty term is proposed for the loss function, which significantly improves the accuracy of the ANN on modeling the voltage phase angle for large-scale grids.
- An optimization-based FDI attack formulation is proposed for the ANN-based AC state estimation model, which can accommodate various practical constraints on the attacker, including their resource and meter accessibility.
- We adapt two algorithms, DE and SLSQP, to solve the above optimization, targeting at two different attack scenarios: DE generates attack vectors for the scenario, in which the attacker can compromise any k meters, while both DE and SLSQP can accommodate for the scenario, in which the attacker has only access to specific k meters.
- The effectiveness of the proposed attack models is verified by extensive simulations on IEEE 9-bus, 14-bus, and 30-bus systems under various attack scenarios. Our results show that the DE attack is successful with high probability (more than 80% in all simulated cases), despite having a small number of compromised meters and low false injection level.
- We adopt an adversarial training to defend against the above attacks. It turns out the adversarial training could lower the attack success rate, but would slightly impair the model accuracy.

The proposed algorithms provide a practical way for systematically identifying key meters whose readings have a higher weight in the state estimation, thus may serve as a guide to the utility company to reach a more focused/concentrated protection against these key meters under resource and budget constraints. Furthermore, our defense strategy encourages building more robust ANN-based state estimation models in the future.

This remainder of the paper is organized as follows. In Section 2, we survey the ANN-based state estimation, false data injection attack, as well as adversarial example. We then provide a preliminary for state estimation and bad data detection in Section 3. We construct ANN-based state estimation models as our attack targets, and evaluate their performance in Section 4. Subsequently, we introduce our adversary model and attack formulation in Section 6. Our two attack algorithms, DE and SLSQP algorithms are presented in Section 6. The experimental analysis and the proposed defense are presented in Section 7 and Section 8, respectively.

2. Related work

2.1. ANN-Based state estimation

Various neural network architectures are explored for state estimation in the smart grid, such as the feed-forward neural network Abdel-Nasser et al. (2018), radial basis function neural network Singh et al. (2004), counter propagation network and functional link network Kumar et al. (1996). In Onwuachumba and Musavi (2014), Onwuachumba *et al.* proposed a reduced ANN-based state estimation model, which uses fewer measurements and no prior observability analysis is required. To adapt to the new features emerged in smart grid, such as renewable generators and dynamic pricing, the ANN-based state estimation for real-time and distributed power systems is studied in Mestav et al. (2018); Mosbah and El-Hawary (2015); Zamzam et al. (2019); Zamzam and Sidiropoulos (2020).

2.2. False data injection attack

Existing results on FDI attacks against conventional state estimations are inapplicable to the ANN-based state estimation due to the following two reasons. First, most prior works on FDI attacks are based on the *Direct current* (DC) power flow model Esmalifalak et al. (2011); Hug and Giampapa (2012); Liu et al. (2011); Sandberg et al. (2010), which is a linear approximation of the real-world *alternate current* (AC) power flow model, and is usually used as a simplified version of the AC power flow model. FDI attacks against AC models are more complicated, and hence require a more sophisticated attacker than DC models. The FDI attacks derived from DC models may be ill-suited for AC models Rahman and Mohsenian-Rad (2013). In addition, works on constructing FDI attacks against AC models are mainly targeting on WLS state estimators Hug and Giampapa (2012); Jia et al. (2012); Liang et al. (2014); Teixeira et al. (2011); Wang et al. (2015), thus cannot be directly applied to ANN-based state estimators.

A considerable number of works have been proposed to defend against FDI attacks. The authors in Bobba et al. (2010) approached the issue by identifying and protecting a set of critical meters in order to detect FDI attacks. The authors in Chakhchoukh et al. (2020); Li et al. (2017); Sedghi and Jonckheere (2013) approached the issue from a statistical method combined with physical laws of the power system. Data-driven and machine learning based approaches were proposed in Esmalifalak et al. (2017); Guo et al. (2019); He et al. (2017); Yu et al. (2018); Zhang et al. (2019). A Kalman filter based detector was developed in Manandhar et al. (2014). Liu *et al.* developed a detection by using the sparsity of the attacks Liu et al. (2014). The authors in Li et al. (2015) proposed a sequential detector and the authors in Huang et al. (2011) proposed an adaptive CUSUM algorithm, in order to accelerate the detection process.

Table 1 – Notations and definitions.

Notations	Definitions
n, m	Number of state variables/measurements
$\mathbf{x}, \mathbf{x_a}, \hat{\mathbf{x}}$	Natural/compromised/estimated state variables, including voltage magnitude $\|V_i\|$ and phase angle θ_i at all buses, $i = 1, ., n$
P_i,Q_i	Real and reactive power injection at bus i.
P_{ij}, Q_{ij}	Real and reactive power injection at branch connecting bus i to bus j
$\mathbf{z}$, $\mathbf{z_a}$	Natural/compromised measurements, including real and reactive power injection of buses P_i and Q_i and branches P_{ij} and Q_{ij}
$h(\cdot)$	A set of non-linear, deterministic functions that relate states to measurements $h : \mathbf{x} \rightarrow \mathbf{z}$
$f(\cdot)$	ANN-based state estimator that eliminates errors in measurements and output
$\mathbf{a}$	Attack vector that injects to a given measurement $\mathbf{z}$
$G_{ij} + jB_{ij}$	The ij-th element of the complex bus admittance matrix
$g_{ij} + jb_{ij}$	The admittance of the series branch connecting busses i and j
$g_{sj} + jb_{sj}$	The admittance of the shunt branch connected at bus i

2.3. *Adversarial examples*

Szegedy *et al.* were the first to propose the adversarial attack against deep neural networks Szegedy et al. (2014). After that, various attack algorithms are proposed, such as the Fast Gradient Sign Method (FGSM) Goodfellow et al. (2014), Fast Gradient Value (FGV) Rozsa et al. (2016) and DeepFool Moosavi-Dezfooli et al. (2016). Especially, in Su et al. (2019), the deep learning model can be fooled by adding one pixel perturbation to the image. Furthermore, the perturbations are shown to be transferable among ANN models, even if they are trained on different data sets, and preserve different architectures Kurakin et al. (2016); Liu et al. (2016); Tramèr et al. (2017); Xie et al. (2019).

Another branch of research studies the defense against adversarial examples. Papernot *et al.* used a distillation network to extract knowledge to improve the robustness Papernot et al. (2016). In the adversarial training, the adversarial examples are generated in every training step, then they are injected to the training data set Goodfellow et al. (2014); Huang et al. (2015); Madry et al. (2017). And in the classifier robustifying, the authors in Abbasi and Gagné (2017); Bradshaw et al. (2017) put emphasis on how to design a robust architecture of the ANN.

3. Preliminaries

In this section, we briefly introduce the state estimation and bad data detection. All notations used are defined in Table 1.

3.1. *State estimation*

In the AC power flow model, measurements are non-linearly dependent on state variables, as characterized by the following equation:

$$\mathbf{z} = h(\mathbf{x}) + \mathbf{e},$$

where $\mathbf{z}$ and $\mathbf{x}$ denote a N_m-dimension measurement vector and a N_n-dimension state vector, respectively, and $\mathbf{e}$ denotes a N_m-dimension vector of normally distributed measurement errors. $h(\mathbf{x})$ denotes a set of non-linear functions, by which the measurements are related to state variables, according to Kirchhoff's circuit law:

$$P_i = V_i \sum_{j=1}^{N} |V_j|(G_{ij}cos\theta_{ij} + B_{ij}sin\theta_{ij}), \tag{1}$$

$$Q_i = V_i \sum_{j=1}^{N} |V_j|(G_{ij}sin\theta_{ij} - B_{ij}cos\theta_{ij}), \tag{2}$$

$$P_{ij} = |V_i|^2\left(g_{si} + g_{ij}\right) - |V_iV_j|\left(g_{ij}\cos\theta_{ij} + b_{ij}\sin\theta_{ij}\right), \tag{3}$$

$$Q_{ij} = -|V_i|^2(b_{si} + b_{ij}) - |V_iV_j|(g_{ij}sin\theta_{ij} - b_{ij}cos\theta_{ij}). \tag{4}$$

In an over-determined case, where we have more measurements than state variables ($N_m > N_n$), the state variables are determined from the WLS optimization over a residual function $J(\mathbf{x})$ Wood and Wollenberg (1996):

$$\hat{\mathbf{x}} = \arg\min_{x} J(\mathbf{x}), \text{ where } J(\mathbf{x}) = (\mathbf{z} - h(\mathbf{x}))^T\mathbf{W}(\mathbf{z} - h(\mathbf{x})). \tag{5}$$

Here, the weight matrix $\mathbf{W}$ is defined as $diag\{\sigma_1^{-2}, \sigma_2^{-2}, \ldots, \sigma_{N_m}^{-2}\}$, and σ_i^2 is the variance of the i-th measurement ($i = 1, \ldots, N_m$). $\mathbf{W}$ is introduced to emphasize trusted measurements while de-emphasizing less trusted ones.

3.2. *Bad data detection*

Meter measurements may contain errors due to various reasons, such as transmission error, wiring failure or malicious attack. Therefore, for data quality control purpose, a bad data detection is usually introduced to identify measurements whose error exceeds a pre-defined threshold. The integration of the state estimation and the bad data detection can largely suppress the presence of bad data and ensure that the state estimation is based on only good data. Most bad data detection schemes rely on the residual $J(\hat{x})$ as their decision variable. In particular, given the assumption that $\mathbf{e}$ is normally distributed, it is shown that $J(x)$ follows a $\chi^2(K)$ distribution, where $K = N_m - N_n$ is the degree of freedom. Any measurements with a residual greater than the pre-determined threshold τ is recognized as bad data:

z is identified as bad data, if

$$J(\hat{\mathbf{x}}) = (\mathbf{z} - h(\hat{\mathbf{x}}))^T\mathbf{W}(\mathbf{z} - h(\hat{\mathbf{x}})) > \tau. \tag{6}$$

The threshold τ can be determined by a significant level α in hypothesis testing, indicating the false alarms would occur with probability α.

4. ANN-Based AC state estimation

The main difficulty in utilizing Eq. (5) directly to estimate the AC state is that it requires solving a nonlinear optimization problem. Instead of making any particular assumption on the structure of $h(\cdot)$, we adopt an empirical methodology to characterize the non-linear state estimation function. In particular, based on a sufficient number of empirical state-measurement readings, we attempt to train an ANN model that can accurately represent the states as a nonlinear function of the measurements. In the operational phase, this ANN is expected to directly output a state estimation $\hat{\mathbf{x}}$ for each input of measurements $\mathbf{z}$, without the need of solving the nonlinear optimization in Eq. (5). In the following, we present our procedure in generating the training data, defining the loss function, training the ANNs, and testing the accuracy of the trained ANN state estimators.

4.1. *Model training*

Although it would be more convincing by using actual data from a real power grid, power companies use their own proprietary data format, in which most of them are not accessible. Therefore, lacking actual state-measurement data from a real power grid, we follow the convention to present our results based on computer simulations, as in previous studies (e.g. Che et al. (2019); Chen et al. (2012); Liu et al. (2011)). Simulation-based evaluation would give valid results, because the simulation data is generated according to realistic grid typologies and well-established physical laws/mechanics that govern the operation of the grids. In addition, simulation data provides a wider range of the operational condition coverage. In particular, real meter data can only cover a limited set of operational conditions of the grids under which these actual data are recorded, while the simulation data has a much wider coverage on the grids' operation conditions as these data can be generated on demand, for any operation condition of interest.

The training and testing cases in our study are generated by simulations over the IEEE test systems (9-bus, 14-bus, 30-bus). A Matlab package, MATPOWER Zimmerman et al. (2011), is used for data generation and power flow analysis. Note that the use of simulation data in training does not affect the validity of the proposed ANN model. One can simply replace the simulation data by actual data once they become available, and then re-train the ANN by same procedure.

Our state-measurement data are generated in the following way. The state variable, consisting of the magnitude $|V_i|$ and phase angle θ_i of the bus voltages, is a function of the load of the power system, and changes within a small range. To account for this dynamic behavior, we consider a series of loads of the power grid ranging from 80% to 120%. For each instance of loads, the state is calculated by power flow analysis. According to the American National Standard for Code for Electricity Metering ANSI (2008), class 2 accuracy applies for power grid measurements, which tolerates a ±2% error in a measurement reading. In line with this specification, we add an independent Gaussian noise ϵ to each measurement reading ψ, so that the simulated measurement reading becomes $(1+\epsilon)\psi$, where $\epsilon \sim N(0, 0.67\%^2)$. For each of the test systems, 10,000 and 1000 state-measurement pairs are generated for training and testing, respectively. Note that all constant values are excluded from measurements and state variables.

An ANN-based state estimation model is trained for each of the test systems. Following Abdel-Nasser et al. (2018); Jain et al. (2008); Menke et al. (2019); Mosbah and El-Hawary (2015), each ANN state estimation model possesses a *multi-layered perceptron* (MLP) architecture, consisting of one input layer, one or more hidden layers, and one output layer. We use the mean WLS error as the loss function:

$$loss(\mathbf{z}, \mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{z} - h(\mathbf{x}))^T\mathbf{W}(\mathbf{z} - h(\mathbf{x})), \tag{7}$$

where N is the number of training samples.

Our experiments show that the accuracy on both voltage magnitude and phase angle are satisfactory, yet the phase angle accuracy is lower. There are several reasons behind this phenomenon. First, the loss function only narrows the difference between the actual and estimated measurements. Being different from conventional machine learning problems, the state estimation requires the error to be minimized from both measurement and state sides. Second, the voltage magnitudes are strictly confined in a small range in order to provide a stable and consistent power supply.

These trained models serve as the targets for our proposed attacks. The inaccuracy in the state estimation, i.e., the deviation of the estimated state from the actual state, overlays the goal of the FDI attack. So any estimation inaccuracy would be counted as an attack success in the attack evaluation. To eliminate such effect, we revise the loss function in order to achieve high accuracies on both voltage magnitude and phase angle. A new penalty term of the *mean square error* (MSE) between the actual state and the estimated state is added in Eq. (7), leading to a new loss function in Eq. (8) specially designed for large-scale systems. In this new loss function, a small constant c is added to balance both error terms so that the gradient descent works on both terms simultaneously:

$$loss(\mathbf{z}, \mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{z} - h(\mathbf{x}))^T\mathbf{W}(\mathbf{z} - h(\mathbf{x})) + c\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x} - \hat{\mathbf{x}})^2. \tag{8}$$

Empirically, we investigate the value of c spaced uniformly (on a log scale) from $c = 1 \times 10^1$ to $c = 1 \times 10^5$, and choose a c that brings the best estimation accuracy. Our experiments show that by adding this new penalty term, the voltage phase angle estimation accuracy increases to an equivalent level as that of the voltage magnitude. The proposed ANNs are implemented in Python, using *TensorFlow* package with *Keras* as back-end. The model architectures and parameters are given in Table 2.

4.2. *Model evaluation*

After the models are trained, we use testing data to evaluate their performance. A good state estimation model should have the following two properties: first, it should be able to provide

Table 2 – ANN-based state estimator architectures and parameters.

	9-bus	14-bus	30-bus
Architecture			
Input Size	42	103	204
Fully Connected + ReLU	64	128	256
Output Size	14	22	53
Parameter			
Learning Rate	0.001	0.001	0.001
Decay Rate	1×10^{-5}	1×10^{-5}	1×10^{-5}
Batch Size	64	64	64
Epochs	300	500	1000

Table 3 – Model evaluation on voltage magnitude.

Test System	MAE (p.u.)	MARE	Bad Data(%)	Accuracy (%)
9-bus	2.2×10^{-5}	2.4×10^{-5}	0	100
14-bus	5.8×10^{-5}	5.6×10^{-3}	3	100
30-bus	6.3×10^{-5}	6.5×10^{-5}	5	100

Table 4 – Model evaluation on voltage angle.

Test System	MAE (rad)	MARE	Accuracy(%)
9-bus	1.0×10^{-4}	1.6×10^{-2}	96
14-bus	6.1×10^{-3}	2.6×10^{-2}	99
30-bus	1.2×10^{-4}	1.3×10^{-2}	98

accurate state estimation irrespective of the noise in the measurements; second, regular measurement noises should not trigger bad data alarms (i.e., low false alarm rate). Accordingly, we evaluate the estimation accuracy of the ANNs by *maximum absolute error* (MAE) and the *maximum absolute relative error* (MARE) between the true and the estimated values, where MARE is simply MAE normalized w.r.t. the magnitude of the true value. An estimation is considered accurate if the MARE of the voltage magnitude and the voltage phase angle do not exceed 1% and 5%, respectively. To evaluate the false alarm rate, we use a bad data significant level $\alpha = 0.01$. Table 3 and Table 4 summarize the performance evaluation for the trained ANN models. It is clear from these tables that the proposed ANN models are able to estimate states accurately, and have low false alarm rate under regular measurement noises.

5. Adversarial model and attack formulation

In this section, we present a detailed adversarial model against the ANN-based state estimator, following Yuan et al. (2019). This model characterizes the adversary by their goal, knowledge of the data and the target system, and resource and meter accessibility constraints. Based on this model, we formulate an optimization problem that the attacker can use to decide their best attack strategy.

5.1. Adversarial model

It is realistic and practical for an attacker to have the capability to compromise meters, given the fact that the meters are physically distributed and lack protection. The goal of the attacker is to launch an FDI attack, in which the attacker aims to inject a manipulated measurement vector, whose ultimate goal is to maximize the state estimation error while remaining undetected. The false data is injected to the compromised meters, then collected by the SCADA system, and eventually sent to the state estimation application.

The attacker is assumed to have full knowledge of the topology and configuration of the power grid, such as the nodal admittance matrix. Such information could be accessed or estimated from public database or historical records. In addition, the attacker is also assumed to know everything about the ANN-based state estimation model, including the architecture and parameters. These information could be obtained by an attacker either through breaking into the information system of the power grid (similar to the 2015 Ukraine case) or through training a shadow ANN that mimics the real ANN-based state estimator on a substitute data set. We assume that the attacker also knows the threshold of the bad data detector.

Although these assumptions render a strong attacker that may not always represent the practical cases, it enables us to evaluate the robustness and vulnerabilities of the ANN-based state estimators under the worst-case scenario, which provides an upper bound on the impact of FDI attacks against the ANN-based state estimation.

In addition to the bad data detection threshold, the adversary also faces other constraints, including the set of meters they have access to, the maximum number of meters they can compromise, and the maximum amount of errors they can inject into the actual measurements without being detected.

Note that in this paper we only consider the FDI attacks that happen during the operational phase of the ANN-based state estimator. In other words, the adversary is only able to tamper the measurement inputs after the ANN model is trained. It is not allowed to perturb either the training data or the trained model. The investigation of training data or model poisoning is out of the scope of this paper and will be studied in our future work.

5.2. Attack formulation

Let $\mathbf{z}_a$ be the measurement vector in the presence of FDI attack, then $\mathbf{z}_a$ can be described as following:

$$\mathbf{z_a} = \mathbf{z} + \mathbf{a} = h(\mathbf{x}) + \mathbf{a}, \tag{9}$$

where $\mathbf{a}$ is a N_m-dimension non-zero attack vector. Given the input of a manipulated measurement $\mathbf{z_a}$, the state estimation output by ANN-based state estimator f is as follows:

$$\hat{\mathbf{x}}_\mathbf{a} = f(\mathbf{z_a}) = f(\mathbf{z} + \mathbf{a}). \tag{10}$$

According to Eq. (6), an adversary intending to elude bad data detection must satisfy the following condition:

$$J(\hat{\mathbf{x}}_\mathbf{a}) = (\mathbf{z_a} - h(\hat{\mathbf{x}}_\mathbf{a}))^T \mathbf{W} (\mathbf{z_a} - h(\hat{\mathbf{x}}_\mathbf{a})) \leq \tau. \tag{11}$$

The error injected to the state estimation hence can be calculated by:

$$\hat{\mathbf{x}}_\mathbf{a} - \hat{\mathbf{x}} = f(\mathbf{z}_\mathbf{a}) - f(\mathbf{z}). \tag{12}$$

With the above notations, the problem of finding the best adversarial injection $\mathbf{a}$ for a given measurement $\mathbf{z}$ can be formulated as a constrained optimization:

$$\begin{aligned} \underset{\mathbf{a}}{\text{maximize}} \quad & \|\hat{\mathbf{x}}_\mathbf{a} - \hat{\mathbf{x}}\|_p \\ \text{subject to} \quad & (\mathbf{z}_\mathbf{a} - h(\hat{\mathbf{x}}_\mathbf{a}))^T \mathbf{W} (\mathbf{z}_\mathbf{a} - h(\hat{\mathbf{x}}_\mathbf{a})) < \tau, \\ & \|\mathbf{a}\|_0 \le L, \\ & a_i^l \le a_i \le a_i^u, i = 1, \dots, N_m, \\ & z_i^{min} \le z_{a_i} \le z_i^{max}, i = 1, \dots, N_m, \end{aligned} \tag{13}$$

where L is the maximum number of meters that the attacker can compromise (so that they can tamper the meter reported measurement), and $[a_i^l, a_i^u]$ provides the lower and upper limits of modification to the measurement of each compromised meter, and $[z_i^{\min}, z_i^{\max}]$ denotes the valid range for each measurement, ensuring that the manipulated measurement is still within the permitted range on that particular unit. The strength of the measurement modification/manipulation depends on the attacker's resource and meter accessibility constraints, which have not been considered in previous works. In our work, by limiting the measurement manipulation to a subset of meters, the attacker can avoid injecting excessive errors, which can be easily detected by a univariate analysis. In addition, if the adversary knows where the high precision meters are located, they can avoid injecting too much errors into those meters and instead allocate the resource to other meters to improve the overall attack outcome.

The objective function in the optimization Eq. (13) requires some distance metric $\|\cdot\|_p$ to quantify the attack impact. In this work, we evaluate the ANN-based state estimation by examining if the state estimation is misled by an injection vector whose values are limited to a noise level. The injection is tiny itself, and its impact will be further cracked by the nonlinearity of the AC power model. Therefore, this distance metric must be carefully chosen. In reality, the voltage magnitude is always limited in a tight range in order to ensure the stable electricity supply, whereas the voltage phase angle varies in a relatively large range. Hence, an erroneous estimation of the latter may seriously affect the consistent operation of the power grid, but cannot be easily detected. Therefore, instead of targeting on the total difference contributed by both voltage magnitudes and the voltage phase angles, we define the adversary's objective function as the maximum change to the voltage phase angles θ:

$$\|\hat{\mathbf{x}}_\mathbf{a} - \hat{\mathbf{x}}\|_\infty = \max(|\hat{\theta}_{a_1} - \hat{\theta}_1|, \dots, |\hat{\theta}_{a_n} - \hat{\theta}_n|). \tag{14}$$

6. Attack methodology

In this section, we present two algorithms, DE and SLSQP, to solve the proposed optimization Eq. (13).

6.1. Solving the proposed attack with DE

As a population based stochastic optimization algorithm, DE algorithm was first proposed in 1996 by Rainer *et al*. Storn and Price (1997). The population is randomly initialized within the variable bounds. The main optimization process consists of three operations: mutation, crossover, and selection. In each generation, a mutant vector is produced by adding a target vector (father) with a weighted difference of other two randomly chosen vectors. Then a crossover parameter mixes the father and the mutant vector to form a candidate solution (child). A pair-wise comparison is drawn between fathers and children, whichever is better will enter the next generation.

We follow Su et al. (2019) to encode our measurement attack vector into an array, which contains a fixed number of perturbations, and each perturbation holds two values: the compromised meter index and the amount to inject to that meter.

The use of DE and the encoding have the following three advantages for generating attack vectors:

- **Higher probability of finding global optimum** - In every generation, the diversity introduced by the mutation and crossover operations ensures the solution not to be stuck in a local optimum, thus leads to a higher probability of finding the global optimum Storn and Price (1997); Su et al. (2019).
- **Adaptability for multiple attack scenarios** - DE can adapt to different attack scenarios by our encoding method. On one hand, by specifying the number of meters to compromise, DE can search for both meter indices and injection amount. On the other hand, by fixing the meter indices, DE can only search for injection amount to these specified meters.
- **Parallelizibility to shorten attack time** - The function evaluation of an ANN is computationally demanding. As the smart grid scale increases, generating one attack vector may take seconds to minutes. An attacker must finish the attack vector generation and injection before the next state estimation takes place. DE algorithm is parallelization friendly, as it is based on a vector population. DE operations can be mounted to a computer cluster, so as to significantly expedite the computation for the attack vector.

Next, we present how we adapt the DE algorithm to our proposed attack:

- **Deal with duplicate meter indices** - In our work, instead of outputting the exact meter value, we select to output the injection vector to narrow down search space. We use two approaches to ensure the uniqueness of meter indices in the solution. First, we generate meter indices without replacement in the population initialization. Second, we add a filter in the crossover operation. This filter keeps the meter indices unchanged if the newly selected meter index is repetitive.
- **Ensure the measurement after injection is within range** - A valid measurement reading must satisfy $z_i^{\min} \le z_i + a_i \le z_i^{\max}$, where $z_i^{\min}$ and $z_i^{\max}$ are the lower and upper limit power permitted on z_i. We use an intuitive approach

by replacing $\mathbf{z_a} = \mathbf{z} + \mathbf{a}$ with $\mathbf{z_a} = \min(max(\mathbf{z_a}, \mathbf{z}^{\min}), \mathbf{z}^{\max})$, where the *min* and *max* are element-wise operations.

- **Deal with the overall constraint** - In addressing the constraints, adding a penalty term into the original objective function has been one of the popular approaches. However, they do not always yield satisfactory solutions since the appropriate multiplier for the penalty term is difficult to choose and the objective function may be distorted by the penalty term. Therefore, we use a heuristic constraint handling method proposed in Deb (2000). A pair-wise comparison is drawn between fathers and children in order to differentiate feasible solutions from infeasible ones. The three criteria of the pair-wise comparison are as the following:
  1. If both vectors are feasible, the one with the best objective function value is preferred.
  2. If one vector is feasible and the other one is not, the feasible one is preferred.
  3. If both two vectors are infeasible, the one with the smaller constraint violation is preferred.

  Essentially, the above comparison handles constraints in two steps: first, the comparison among feasible and infeasible solutions provides a search direction towards the feasible region; then, the crossover and mutation operations keep the search near the global optimum, while maintaining the diversity among feasible solutions. The pseudo code for the proposed attack using DE is presented in Algorithm 1.

Algorithm 1 DE attack.

Input: measurement $\mathbf{z}$, GEN_{MAX} {maximum number of generations}, N {population size}, f {objective function}, g {constraint function}, CR {crossover rate}

Output: injection vector $\mathbf{a}$

$g = 0$
Population initialization $\mathbf{a}_{i,0}$ for $i = 1, \ldots, N$. Meter indices are randomly select without replacement and injection amounts are randomly select within the univariate bound.
Evaluate the $f(\mathbf{a}_{i,g})$ and constraint violation $CV(\mathbf{a}_{i,g}) = \max(g(\mathbf{a}_{i,g}), 0)$, for $i = 1, \ldots, N$
for $g = 1 : MAX_{GEN}$ **do**
  for $i = 1 : N$ **do**
    Randomly select r_1 and r_2
    $j_{rand} = randint(1, N_m)$
    for $j = 1 : D$ **do**
      if $(rand_j[0, 1) < CR$ **or** $j = j_{rand})$ **and** the meter index not repetitive with previous meter indices **then**
        $u^j_{i,g+1} = x^j_{best,G} + F(x^j_{r_1,g} - x^j_{r_2,g})$
      else
        $u^j_{i,g+1} = x^j_{i,G}$
      end if
    end for
    Evaluate $f(\mathbf{u}_{i,g+1})$ and $CV(\mathbf{u}_{i,g+1})$
    Update the population if the child $\mathbf{u}_{i,g+1}$ is better than the father $\mathbf{x}_{i,g}$ by above three criteria
  end for
end for

6.2. Solving the proposed attack with SLSQP

In some gradient-based attack algorithms in image classification(Carlini and Wagner (2017); Szegedy et al. (2014)), a logistic function is added to the objective function as a penalty term and the multiplier for the penalty term is chosen by a line search. These algorithms aim to find a feasible solution, not the optimal one. Therefore, we use a conventional optimization algorithm (SLSQP) Kraft (1988). SLSQP is a variation on the SQP algorithm for non-linearly constrained gradient-based optimization. In our SLSQP attack, we encode the solution to a N_m-dimension vector, in which the i-th element denotes the injection amount to the i-th meter. This encoding allows the attacker to generate attack vectors for a set of specified meters by placing upper and lower bounds to the corresponding elements in the attack vector. To solve the proposed optimization problem, we first construct the Lagrangian function:

$$\mathcal{L}(\mathbf{a}, \lambda) = f(\mathbf{a}) + \lambda \cdot g(\mathbf{a}), \tag{15}$$

where

$$\begin{cases} f(\mathbf{a}) = \|\hat{\mathbf{x}_a} - \hat{\mathbf{x}}\|_\infty \\ g(\mathbf{a}) = (\mathbf{z} - h(\hat{\mathbf{x}_a}))^T \mathbf{W} (\mathbf{z_a} - h(\hat{\mathbf{x}_a})) < \tau. \end{cases} \tag{16}$$

In each iteration k, the above problem can be solved by transferring to a linear least square sub-problem in the following form:

$$\begin{aligned} \max_{\mathbf{d}} \quad & \|(\mathbf{D^k})^{1/2} (\mathbf{L^k})^T \mathbf{d} + ((\mathbf{D^k})^{-1/2} (\mathbf{L^k})^{-1} \nabla(\mathbf{a}^k)\| \\ \text{subject to} \quad & \nabla g(\mathbf{a}^k)\mathbf{d} + g(\mathbf{a}^k) \geq 0, \end{aligned} \tag{17}$$

where $L^k D^k (L^k)^T$ is a stable factorization of the chosen search direction $\nabla^2_{zz}\mathbf{L}(\mathbf{z}, \lambda)$ and is updated by BFGS method.

By solving the QP sub-problem for each iteration, we can get the value of $\mathbf{d}^k$, i.e., the update direction for $\mathbf{z}^k$:

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \alpha \mathbf{d}^k, \tag{18}$$

where α is the step size, which is determined by solving an additional optimization. The step size $\psi(\alpha) := \phi(\mathbf{a}^k + \alpha d^k)$ with $\mathbf{x}^k$ and d^k are fixed, can be obtained by a minimization:

$$\phi(\mathbf{a}^k; r) := f(\mathbf{a^k}) + max(r \cdot g(\mathbf{a}), 0), \tag{19}$$

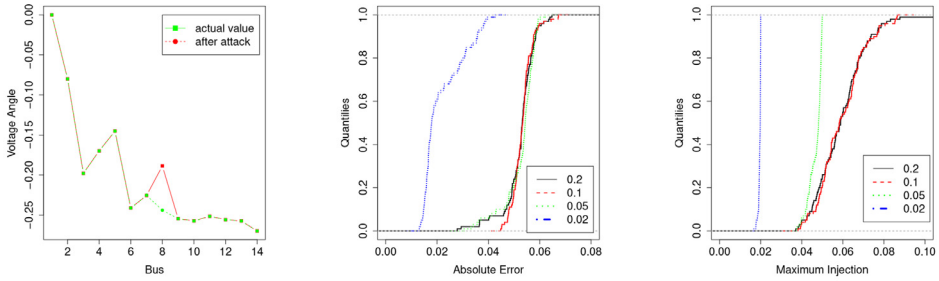
with r being updated by:

$$r^{k+1} := max(\frac{1}{2}(r^k + |\lambda|, |\lambda|)). \tag{20}$$

The limit on the injection amount is achieved by setting a bound to the optimizing variable. The physical constraint for branch limit is ensured by performing an element-wise min-max operation as it is in the DE attack.

7. Attack evaluation

In this section, we evaluate both FDI attacks on three IEEE test systems: 9-bus, 14-bus, and 30-bus systems. The implementation of our attacks is done in Python, using package *TensorFlow*



(a) Voltage phase angle before and after the attack.

(b) CDF of absolute error in voltage phase angle.

(c) CDF of maximum injection.

Fig. 1 – An example of a 5-meter attack on the 14-bus system.

and *SciPy*. We run the experiments on a computer equipped with a 3.5 GHz CPU and a 16 GB memory.

Attack Scenarios: Depending on the attacker's capabilities and practical constraints, the attacker can launch an attack under different scenarios. Inspired by Liu et al. (2011), we consider the following two attack scenarios to facilitate the evaluation:

- **Any k-meter attack** - The attacker can access all meters, but the number of compromised meters is limited by k. In this scenario, the attacker may want to wisely allocate the resource, by selecting meters and injection amount that maximize the attack impact.
- **Specific k-meter attack** - The attacker has the access to k specific meters. For example, the attacker may only access meters in a confined region. In this case, the attacker needs to determine the injection amount to each meter to maximize the attack impact.

We perform the experiments as follows. To fairly compare the attack performance on different test systems, we choose the percentage of compromised meters, R, to be 5%, 10% and 20%. For each R, we explore the attack performance under different injection levels: 2%, 5% and 10%. The injection level is defined as the maximum injection strength in terms of the proportion to the measurement. Each experiment runs on 1000 measurement instances, and is repeated for 10 times to reduce randomness.

We consider the following four metrics throughout evaluating the effectiveness of the attacks. We measure the MAE and MARE that are injected to the voltage phase angle. We also report the success rate, where the success is defined as an attack producing more than 5% MARE to the voltage phase angle. Moreover, since the smart grid is assumed to be a quasi-static system and the state changes slowly over time, we want to investigate if the time between two state estimations allows an adversary to mount the FDI attack on the smart grid.

7.1. *Any k meter attack*

Under this scenario, the attacker can access all meters and has the freedom to choose any k meters to compromise. The way we encode the attack vector in DE enables the search for better target meters in every generation. In contrast, SLSQP only allows us to put constraints on specific meter indices. Therefore, only DE can be used to find the attack vector in any k-meter attack. *DE/x/y/z* denotes a DE variant, in which x specifies the vector to be mutated is chosen by "random" or "best", and y denotes the number of difference vectors, and z denotes the crossover scheme. We implement three DE variants in our experiments: *DE/best/1/bin, DE/current to best/1/bin* and *DE/current to rand/1/bin*, where *bin* denotes binomial. These DE variants differ in the way of how the father vector is selected and how the differential variation is formed. We find that there is no significant difference among them. Hence, *DE/best/1/bin* is used through all experiments:

$$u_{i,G+1} = x_{best,G} + F(x_{r_1,G} - x_{r_2,G}),$$

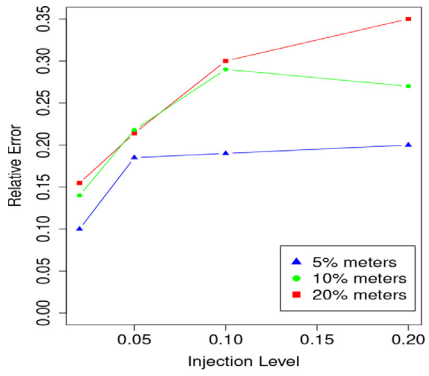
where $x_{r_1,G}, x_{r_2,G}$ are integers drawn from the current population, and $x_{best,G}$ denotes the best individual in terms of objective function value in the current population. F is a real and constant factor $\in [0.5, 1]$, which controls the mutant intensity.

Fig. 1 shows an example of a 5%-meter attack on the 14-bus system. Our DE attack injects error to one of voltage phase angles while others keeping unchanged. In Fig. 1 (b) and (c), for injection levels 10% and 20%, the maximum injections are condensed at 5% and seldom go beyond 10%, due to the overall constraint of bad data detection.

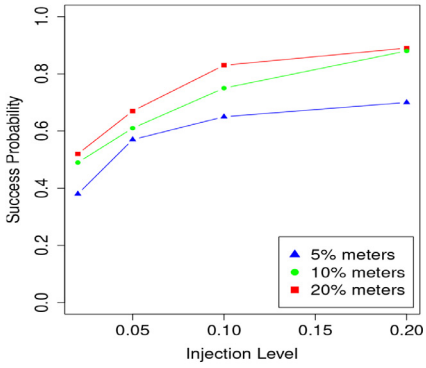
Fig. 2 shows the attack impact with the change of R and injection level. In general, the success probability and attack impact increase as the attacker controls more resource. The attacker achieves a high success rate (80% of simulation instances) by compromising 10% of meters with injection level 10%. Especially for the 14-bus system, the attack achieves 100% success for any combination of R and injection level.

Interestingly, for the 30-bus system, the impact of 10% compromised meters surpasses that of 20% compromised meters. Moreover, the performance of 20% of compromised meters drops drastically as the injection level increases. A possible explanation for this is that, with the expansion of search dimension and space, DE requires more generations to find a satisfactory solution.

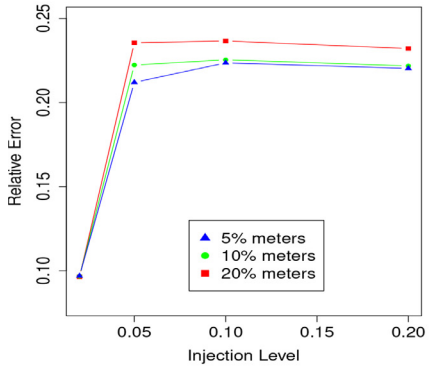
We compare our proposed attack with a random attack, where the injection vectors are generated from a uniform distribution. The success probability is reported on the same set of instances with 1,000 attempts on each instance. The suc-



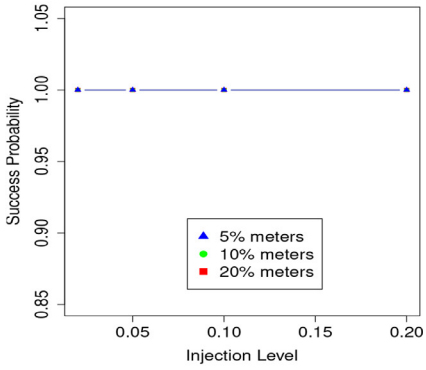
(a) 9-bus relative error.



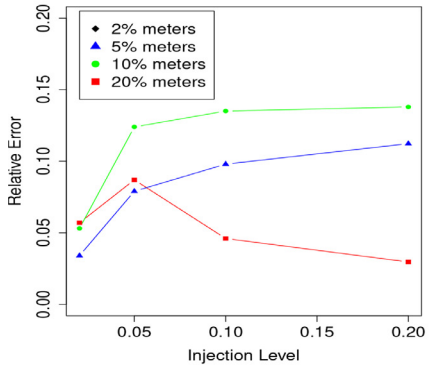
(b) 9-bus success rate.



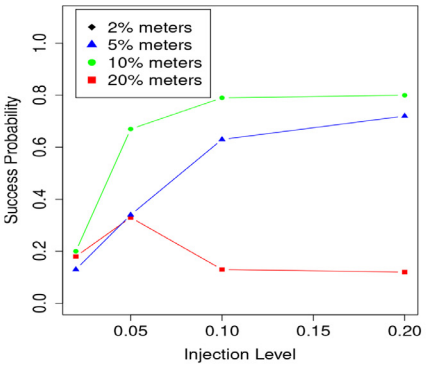
(c) 14-bus relative error.



(d) 14-bus success rate.



(e) 30-bus relative error.

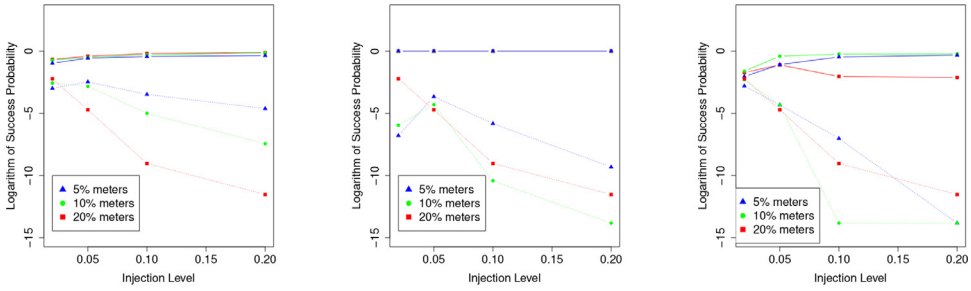


(f) 30-bus success rate.

Fig. 2 – Relative error and success rate of the any *k*-meter attack on 3 test systems with $N = 400$ and $G_{MAX} = 400$.

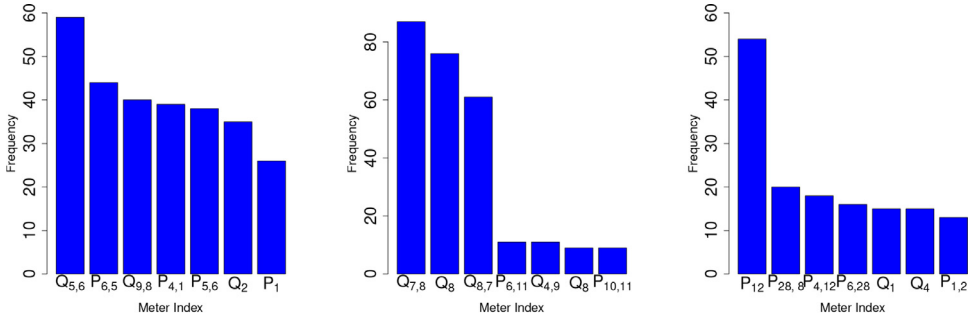
cess rate is compared with that of our DE attack on a log scale (Fig. 3). There is no significant difference between the impact of the DE attack and that of the random attack when the injection level is low, in which the attack impact is limited. However, if the attacker wants to achieve greater impact, our DE attack outperforms the random attack by order of magnitude.

Fig. 4 shows the frequency of the meter indices presenting in the attack vectors. Because most of the meter frequencies are small, only the 7 meters with largest frequencies are presented. Injection to meters with high frequency can introduce large error to the state variable. Our DE attacks also help to identify vulnerable meters, on which people can strengthen



(a) 9-bus. (b) 14-bus. (c) 30-bus.

Fig. 3 – Success rate of the DE attack and the random attack on a log scale. Solid lines refer to the DE attack and dashed lines refer to the random attack.



(a) 9-bus. (b) 14-bus. (c) 30-bus.

Fig. 4 – Frequency of meters selected in the attack vectors.

Table 5 – Average NFEs and execution time (in second) of the any k-meter attack on 3 test systems.

Test System	NFEs	Time (s)
9-bus	500–1500	0.25-0.45
14-bus	500–3500	0.5-1.73
30-bus	800–5600	1.5-2.7

Table 6 – Convergence time (in second) comparison of the specific k-meter attack on 3 test systems.

Test System	DE (s)	SLSQP (s)
9-bus	0.12-0.4	0.036-0.6
14-bus	0.06-0.6	0.14-1.0
30-bus	0.3-3.0	0.26-2.2

the physical protection, e.g., replace them with higher precision meters or lock them in boxes.

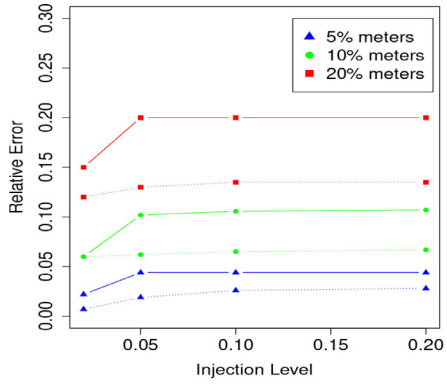
7.2. *Specific k meter attack*

To explore the effect of the population size and iteration number, we evaluate the average *number of function evaluations* (NFEs) before delivering a successful attack or when no significant change in the solution is observed. In the DE case, NFE is equal to the population size multiplied by the number of generations. The NFEs and the corresponding running time are shown in Table 5. For all combinations of systems and attack settings, the attacker can find a successful attack vector in 3 seconds or conclude the attack is infeasible.

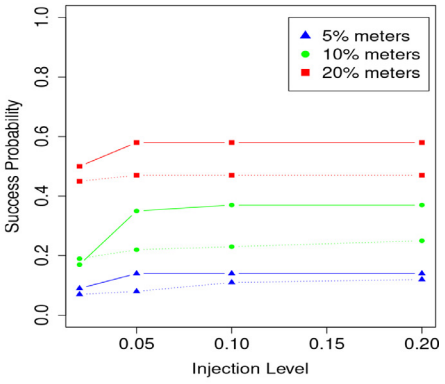
In this scenario, the attacker is able to compromise specific k meters due to the physical location restriction. DE and SLSQP are implemented and compared under this attack scenario. To search the injection amount to specific k meters, DE specifies the indices of the k meters in population initialization and disables the meter index mutation operation, while SLSQP only allows modifications to the k meters in the attack vector. We randomly select 3 sets of meters such that R is 5%, 10% and 20%, respectively. We perform the same set of experiments using both DE and SLSQP algorithms and compare their performance by the same metrics.

In general, the DE algorithm outperforms the SLSQP algorithm in effectiveness (Fig. 5). This is not surprising, as the DE brings in more diversity in every generation whereas SLSQP only explores the neighbors in each iteration.

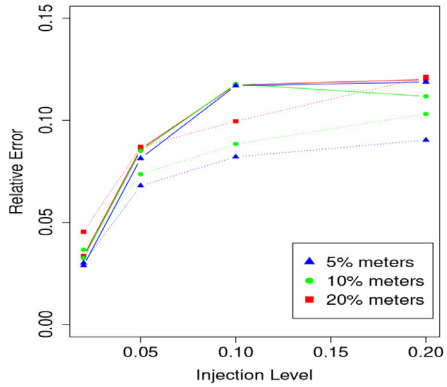
Table 6 shows the convergence time of the DE attack with 10,000 NFEs and the SLSQP attack with 100 iterations. Both attacks converge quickly within 3 seconds, which is feasible for



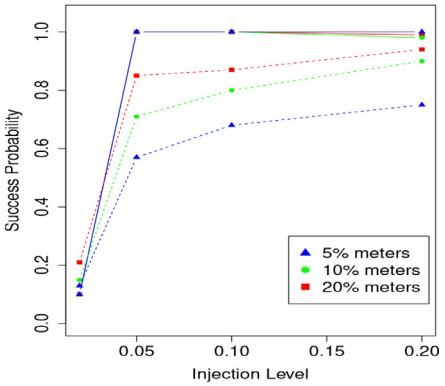
(a) 9-bus relative error.



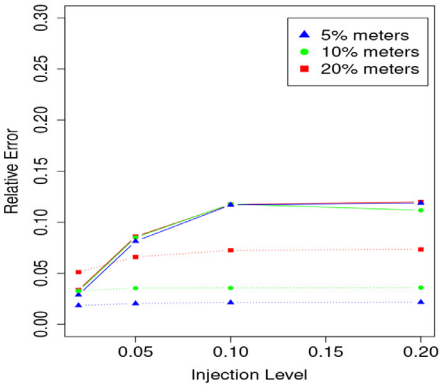
(b) 9-bus success rate.



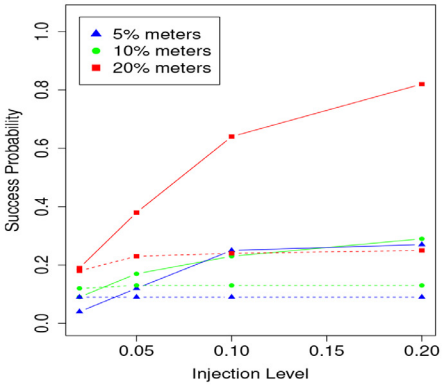
(c) 14-bus relative error.



(d) 14-bus success rate.



(e) 30-bus relative error.



(f) 30-bus success rate.

Fig. 5 – Relative error and success rate of the specific *k*-meter attack on 3 test systems.

an attacker to finish before the next state estimation takes place. A simple comparison of running time between them can be misleading, since the specific k meters engaged in our test are blindly chosen. The convergence time highly relies on the meters chosen to perform the attack. The participation of vulnerable meters would greatly shorten the attack time. In addition, the execution time can be further shortened by applying an early-stop criteria or parallel processing to the DE attack, or adjusting the max iterations for the SLSQP. Therefore, taking no account of the running time, our experiments exhibit clear pattern that the DE attack is more effective than the SLSQP attack.

8. Potential defenses

In this section, we are interested in how the proposed attacks behave when a defense mechanism is specifically customized/optimized to these attacks. Note, such a specialized defense mechanism is in sharp contrast to the general defense mechanisms considered in previous works, which do not assume/exploit any knowledge or feature of the proposed attacks. Putting the proposed attacks into the context of a strong and specialized defense mechanism allows us to garner insights on the limit of both the attacker and the defender in a more realistic "sharpest-sword vs. strongest-shield" setting, as in practice "maximum-effort" is commonly executed not only by attackers but also by defenders, especially when it comes to a mission-critical infrastructure such as the power grid. In the following, we first review existing state-of-the-art defense proposals against adversarial examples in image classification, and explain why some of them are not applicable to our problems. Then, we propose an adversarial training based defense mechanism to counter our proposed attacks. Several techniques are also developed to optimize the proposed defense. The performance of the proposed mechanism is evaluated by simulations in Section 7.

Despite the significant number of works on the detection against the FDI attack, most of the existing detection mechanisms are mainly built on the DC state estimations or traditional WLS state estimators. These detection methods achieve high detection accuracy with low false alarm rate, but they are not applicable to the ANN-based state estimator. The defense strategy against the FDI attack on the AC ANN-based state estimation has not been intensively studied.

In the image classification area, proactive countermeasures against adversarial examples aim to make the ANN model more robust before the attacker gets a chance to generate adversarial examples. Mainstreaming proactive countermeasures fall into three categories Yuan et al. (2019): the defensive distillation, adversarial training and classifier robustifying.

However, our problem has a different goal compared to the image classification. Methods based on the probability of the target class, such as the defensive distillation and classifier robustifying, are not applicable. To propose the defense, we need to address two challenges: (1) in contrast to an image classification problem, our goal is to minimize the error in the state space while keeping the residual in the measurement space below a pre-defined threshold; (2) measurements contaminated by a small injection level are well-hidden as they are nearly from the same distribution as clean measurements. The defense should not be sensitive to adversarial injections, yet measurements with regular noises should not trigger bad data detection alarms.

As stated in Jagielski et al. (2018), there are two mainstream methods to strengthen a regression model: the noise-resilient regression and adversarial training. The idea behind the noise-resilient regression is to enhance the model's tolerance to noises, and identify and remove the outliers, while not triggering bad data alarm nor losing accuracy. In the target model training process in Section 4, we adopt the idea of noise resilience by adding noises sampled from a certain distribution to the training data, so the model learns the distribution and is able to eliminate the effect of such noises. In addition, we minimize both the errors in state space and the measurement space to improve the ANN-based state estimation accuracy and narrow down the space left for attacks. While these methods do provide robustness improvement against noises and outliers, results in Section 7 show a noise-resilient model is not resistant to our attacks. It is suggested that an adversary can still generate noise-like injections to mislead the state estimation. It turns out that introducing noises to measurements and minimizing the training error in both spaces do not make the model more robust to adversarial injections.

Among many proposed defenses against adversarial examples, the adversarial training Goodfellow et al. (2014); Szegedy et al. (2014) has been one of the most effective methods Kurakin et al. (2016); Madry et al. (2017). The adversarial training attempts to minimize the impact of the injection in the model training phase, rather than trying to identify and mitigate them in the operational phase of the trained model. This is achieved by a min-max formulation:

$$\theta = \arg\min_{\theta} E_{(x,y)\sim D}\left[\max_{\delta\in S} L(\theta, x+\delta, y)\right], \tag{21}$$

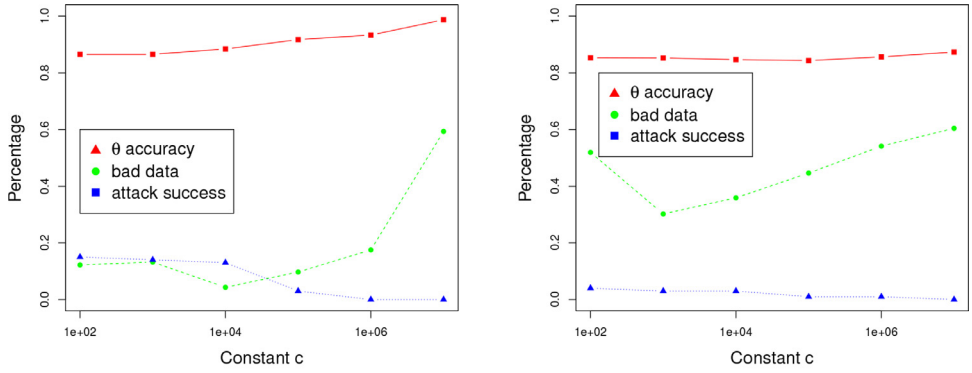
where D is the set of training data, L is the loss function, θ is the parameter of the network, and S is a norm-constrained ball centered at 0. In contrast to the regular training, the adversarial training uses a min-max optimization, where the *inner maximization* produces injection data based on the current model and injects them into the training data set, while the *outer minimization* minimizes the state estimation deviation on the enlarged training data set, in which the injection data is included.

Inspired by Madry et al. (2017) and considering the uniqueness of our problem, we propose a defense through an optimization perspective with the goal of improving the robustness while keeping the accuracy of the ANN-based state estimation model:

$$\theta = \arg\min_{\theta} c \cdot E_{(\mathbf{x},\mathbf{z})\sim D}\left[\max_{\delta} \|\tilde{\mathbf{x}} - \mathbf{x}\|\right] + loss(\mathbf{z}, \mathbf{x}), \tag{22}$$

where $c > 0$ is a suitably chosen constant, controlling the optimization strength on each term. Compared to Eq. (21), a training loss term is added to the optimization such that the model accuracy is taken in to account.

In the process of choosing a suitable c, since the value of the first term is very small, a large c would make the optimization emphasize on minimizing the risk of the FDI attack, whereas a small c would cause a high false alarm rate. Empirically, we found the best way to choose c is to balance the model accuracy, bad data rate, and model robustness. We verify this by running the adversarial training model for values of c spaced uniformly (on a log scale) from $c = 1 \times 10^2$ to $c = 1 \times 10^7$, on the 9-bus system customized for the 10%-meter specific DE and SLSQP attacks respectively. The model accuracy and bad data rate are evaluated on the test data set, while the effectiveness of the adversarial training is evaluated by the DE and SLSQP attacks. We plot the voltage angle accuracy, bad data rate and attack success rate as a function of c



(a) SLSQP. (b) DE.

Fig. 6 – Sensitivity on the constant c.

in Fig. 6. We found both attacks show similar patterns. As c increases, the attacks become rarely successful at the cost of the state estimation model being more conservative. The conservativenss is mainly reflected by the model recognizing a growing number of measurements with regular noises as bad data. In practical state estimation applications, bad measurements are usually discarded and will not be used to estimate the current system status. Therefore, a high false alarm rate would increase the risk of system unobservability. Although the adversarial trained state estimation model could identify more data as bad data, this is a minor model degradation, which can be handily resolved by, for example, increasing the sampling rate.

As claimed in Madry et al. (2017), solving the optimization alone is not a sufficient condition for model accuracy and robustness. What's more, it requires both solving the optimization and the value of the objective function to be small. This is because in general, a smaller objective value implies a better model. However, in our problem, that is not always true. Due to the presence of noise, a smaller objective value does not always indicate a better model. Furthermore, obsessively pursuing a small objective value may lead to overfitting. Therefore, we stop the training process when we observe the loss is consistently smaller than the threshold.

We then use the *Adam optimizer* to adversarially train state estimation models on the 9-bus, 14-bus and 30-bus systems with the same attack settings and meter indices as in Section 7.2. According to our experiment results, the three systems present similar patterns. To evaluate the effectiveness of the adversarial training across all test systems, we present the experiment results of the adversarial training for the 10%-meter specific attack with the injection level of 10% in Table 7, in terms of voltage angle accuracy, bad data rate and attack success rate. While the adversarial training significantly reduces the attack success rate, it achieves this benefit at the cost of an elevated bad data rate and a slight degradation (several percent) in model accuracy, for defenses against both the DE and SLSQP attacks.

The reason for the slightly degraded accuracy is that the adversarial training is done on an enlarged training data set, in which the adversarial data is generated and added to the data set as the training process goes on. At the individual level, one adversarial example may hide well in the actual data distribution. But if looked at the whole population, the adversarial data distribution may differ slightly from the actual data distribution. So the model learned from the adversarial data may shift accordingly, causing a slightly decreased accuracy.

It is also noted that the adversarial training with examples generated by DE has a higher bad data rate than the training with examples generated by SLSQP. One possible explanation is the high skewness in the residual distribution. In the process of generating adversarial examples, while the SLSQP finds adversarial examples in the neighbors, DE, being a stochastic method, always probes more possibilities to make use of the resource. Taking a closer look at the residuals of the adversarial data, we can notice that the residual distribution is highly left skewed and it is highly condensed at the value

Table 7 – Performance of the adversarial training against the specific 10%-meter attack with the injection level of 10%.

Without Adversarial Training				
	θ Accuracy(%)	Bad Data(%)	SLSQP(%)	DE(%)
9-bus	96	0	22	35
14-bus	99	3	71	100
30-bus	98	5	13	17
SLSQP	**With Adversarial Training**			
	θ Accuracy(%)	**Bad Data(%)**	**Attack Success(%)**	
9-bus	91.7	9.7	3	
14-bus	93.3	13.5	5	
30-bus	92.4	14.9	2	
DE	**With Adversarial Training**			
	θ Accuracy(%)	**Bad Data(%)**	**Attack Success(%)**	
9-bus	85.2	30.2	3	
14-bus	86.8	34.5	7	
30-bus	80.3	40.2	1	

of the bad data detection threshold. Due to the skewness, it takes the adversarial training more iterations to converge, yet to a value just below the threshold. Such an unsteady convergence is susceptible to distribution difference, therefore, data from the true distribution are very likely to violate the bad data threshold, resulting in an elevated bad data rate. Note that such a drawback is not critical to the state estimation, as it can be easily overcome by proportionally raising up the sampling rate to compensate for those good data lost due to the false alarm.

In summary, our proposed adversarial training works well in significantly reducing the attack success rate, but only at the cost of a higher bad data rate and a slight degradation of the model accuracy.

9. Conclusions

In this paper, we performed the first study of the adversarial FDI attack against the ANN-based AC state estimation. We first created target models that are sufficiently strong. Then we formulated the adversarial FDI attack into an optimization problem, followed by extensive evaluations under two attack scenarios on IEEE 9-bus, 14-bus and 30-bus test systems, based on the adaption of DE and SLSQP algorithms aiming to find attack vectors. In the any k-meter attack, our results showed that the DE attack achieves high success rate (more than 80% in all simulated cases), despite having a small number of compromised meters and low false injection strength. The DE outperforms SLSQP in the specific k-meter attack. Our findings also showed the potential of the adversarial training in defending against these attacks, and such approach can be further explored to improve the ANN-based AC state estimation model robustness.

Credit Author Statement

Tao Shu and Tian Liu conceived the presented idea. Tian Liu performed the experiments and data analysis. Tao Shu and Tian Liu contributed to the interpretation of the results. Tian Liu wrote the paper with support from Tao Shu.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work is supported in part by NSF under grants CNS-2006998, CNS-1837034, and CNS-1745254. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of NSF.

REFERENCES

Abbasi M, Gagné C. Robustness to adversarial examples through an ensemble of specialists. arXiv preprint arXiv:170206856 2017.

Abdel-Nasser M, Mahmoud K, Kashef H. A novel smart grid state estimation method based on neural networks. IJIMAI 2018;5(1):92–100.

ANSI. ANSI C12.1-2008: American National Standard for Electric Meters: Code for Electricity Metering2008;.

Bobba RB, Rogers KM, Wang Q, Khurana H, Nahrstedt K, Overbye TJ. Detecting false data injection attacks on dc state estimation, volume 2010; 2010.

Bradshaw J, de G MAG, Ghahramani Z. Adversarial examples, uncertainty, and transfer testing robustness in gaussian process hybrid deep networks. arXiv preprint arXiv:170702476 2017.

Carlini N, Wagner D. Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP); 2017. p. 39–57. doi:10.1109/SP.2017.49.

Chakhchoukh Y, Lei H, Johnson BK. Diagnosis of outliers and cyber attacks in dynamic PMU-based power state estimation. IEEE Trans. Power Syst. 2020;35(2):1188–97. doi:10.1109/TPWRS.2019.2939192.

Che L, Liu X, Li Z, Wen Y. False data injection attacks induced sequential outages in power systems. IEEE Trans. Power Syst. 2019;34(2):1513–23. doi:10.1109/TPWRS.2018.2871345.

Chen P, Cheng S, Chen K. Smart attacks in smart grid communication networks. IEEE Commun. Mag. 2012;50(8):24–9. doi:10.1109/MCOM.2012.6257523.

Chen Y, Tan Y, Deka D. Is machine learning in power systems vulnerable?. In: 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm); 2018. p. 1–6. doi:10.1109/SmartGridComm.2018.8587547.

Chen Y, Tan Y, Zhang B. Exploiting vulnerabilities of load forecasting through adversarial attacks. In: Proceedings of the Tenth ACM International Conference on Future Energy Systems; 2019. p. 1–11. doi:10.1145/3307772.3328314.

Deb K. An efficient constraint handling method for genetic algorithms. Comput. Methods Appl. Mech. Eng. 2000;186(2):311–38. doi:10.1016/S0045-7825(99)00389-8.

Esmalifalak M, Liu L, Nguyen N, Zheng R, Han Z. Detecting stealthy false data injection using machine learning in smart grid. IEEE Syst. J. 2017;11(3):1644–52. doi:10.1109/JSYST.2014.2341597.

Esmalifalak M, Nguyen H, Zheng R, Han Z. Stealth false data injection using independent component analysis in smart grid. In: 2011 IEEE International Conference on Smart Grid Communications (SmartGridComm); 2011. p. 244–8. doi:10.1109/SmartGridComm.2011.6102326.

Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. arXiv preprint arXiv:14126572 2014.

Guo Z, Shi D, Quevedo DE, Shi L. Secure state estimation against integrity attacks: agaussian mixture model approach. IEEE Trans. Signal Process. 2019;67(1):194–207. doi:10.1109/TSP.2018.2879037.

He Y, Mendis GJ, Wei J. Real-time detection of false data injection attacks in smart grid: a deep learning-based intelligent mechanism. IEEE Trans. Smart Grid 2017;8(5):2505–16. doi:10.1109/TSG.2017.2703842.

Huang R, Xu B, Schuurmans D, Szepesvári C. Learning with a strong adversary. arXiv preprint arXiv:151103034 2015.

Huang Y, Li H, Campbell KA, Han Z. Defending false data injection attack on smart grid network using adaptive CUSUM test. In: 2011 45th Annual Conference on Information Sciences and Systems. IEEE; 2011. p. 1–6.

Hug G, Giampapa JA. Vulnerability assessment of AC state estimation with respect to false data injection cyber-attacks. IEEE Trans. Smart Grid 2012;3(3):1362–70. doi:10.1109/TSG.2012.2195338.

Jagielski M, Oprea A, Biggio B, Liu C, Nita-Rotaru C, Li B. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In: 2018 IEEE Symposium on Security and Privacy (SP); 2018. p. 19–35. doi:10.1109/SP.2018.00057.

Jain A, Balasubramanian R, Tripathy SC. Topological observability: Artificial neural network application based solution for a practical power system. In: 2008 40th North American Power Symposium; 2008. p. 1–6. doi:10.1109/NAPS.2008.5307305.

Jia L, Thomas RJ, Tong L. On the nonlinearity effects on malicious data attack on power system. In: 2012 IEEE Power and Energy Society General Meeting; 2012. p. 1–8. doi:10.1109/PESGM.2012.6345685.

Kraft D. A software package for sequential quadratic programming. Forschungsbericht Deutsche Forschungs und Versuchsanstalt für Luft und Raumfahrt 1988;88:33.

Kumar DMV, Srivastava SC, Shah S, Mathur S. Topology processing and static state estimation using artificial neural networks. IEE Proceedings-Generation, Transmission and Distribution 1996;143(1):99–105. doi:10.1049/ip-gtd:19960050.

Kurakin A, Goodfellow I, Bengio S. Adversarial machine learning at scale. arXiv preprint arXiv:161101236 2016.

Lee R., Assante M., Conway T.. Analysis of the cyber attack on the ukrainian power grid2016;https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf.

Li B, Lu R, Wang W, Choo KKR. Distributed host-based collaborative detection for false data injection attacks in smart grid cyber-physical system. J Parallel Distrib. Comput. 2017;103:32–41. doi:10.1016/j.jpdc.2016.12.012.

Li J, Lee JY, Yang Y, Sun JS, Tomsovic K. ConAML: constrained adversarial machine learning for cyber-physical systems. arXiv preprint arXiv:200305631 2020.

Li S, Ylmaz Y, Wang X. Quickest detection of false data injection attack in wide-area smart grids. IEEE Trans. Smart Grid 2015;6(6):2725–35. doi:10.1109/TSG.2014.2374577.

Liang G, Weller SR, Zhao J, Luo F, Dong ZY. The 2015 ukraine blackout: implications for false data injection attacks. IEEE Trans. Power Syst. 2017;32(4):3317–18. doi:10.1109/TPWRS.2016.2631891.

Liang J, Kosut O, Sankar L. Cyber attacks on AC state estimation: Unobservability and physical consequences. In: 2014 IEEE PES General Meeting | Conference Exposition; 2014. p. 1–5. doi:10.1109/PESGM.2014.6939486.

Liang J, Sankar L, Kosut O. Vulnerability analysis and consequences of false data injection attack on power system state estimation. IEEE Trans. Power Syst. 2016;31(5):3864–72. doi:10.1109/TPWRS.2015.2504950.

Liu L, Esmalifalak M, Ding Q, Emesih VA, Han Z. Detecting false data injection attacks on power grid by sparse optimization. IEEE Trans. Smart Grid 2014;5(2):612–21. doi:10.1109/TSG.2013.2284438.

Liu Y, Chen X, Liu C, Song D. Delving into transferable adversarial examples and black-box attacks. arXiv preprint arXiv:161102770 2016.

Liu Y, Ning P, Reiter MK. False data injection attacks against state estimation in electric power grids. ACM Transactions on Information and System Security (TISSEC) 2011;14(1):13. doi:10.1145/1653662.1653666.

Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:170606083 2017.

Manandhar K, Cao X, Hu F, Liu Y. Detection of faults and attacks including false data injection attack in smart grid using kalman filter. IEEE Trans. Control Network Syst. 2014;1(4):370–9. doi:10.1109/TCNS.2014.2357531.

Menke JH, Bornhorst N, Braun M. Distribution system monitoring for smart power grids with distributed generation using artificial neural networks. International Journal of Electrical Power & Energy Systems 2019;113:472–80. doi:10.1016/j.ijepes.2019.05.057.

Mestav KR, Luengo-Rozas J, Tong L. State estimation for unobservable distribution systems via deep neural networks. In: 2018 IEEE Power Energy Society General Meeting (PESGM); 2018. p. 1–5. doi:10.1109/PESGM.2018.8586649.

Moosavi-Dezfooli SM, Fawzi A, Frossard P. Deepfool: A simple and accurate method to fool deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016. p. 2574–82.

Mosbah H, El-Hawary M. Multilayer artificial neural networks for real time power system state estimation. In: 2015 IEEE Electrical Power and Energy Conference (EPEC); 2015. p. 344–51. doi:10.1109/EPEC.2015.7379974.

Onwuachumba A, Musavi M. New reduced model approach for power system state estimation using artificial neural networks and principal component analysis. In: 2014 IEEE Electrical Power and Energy Conference; 2014. p. 15–20. doi:10.1109/EPEC.2014.40.

Papernot N, McDaniel P, Wu X, Jha S, Swami A. Distillation as a defense to adversarial perturbations against deep neural networks. In: 2016 IEEE Symposium on Security and Privacy (SP). IEEE; 2016. p. 582–97.

Rahman MA, Mohsenian-Rad H. False data injection attacks against nonlinear state estimation in smart power grids. In: 2013 IEEE Power Energy Society General Meeting; 2013. p. 1–5. doi:10.1109/PESMG.2013.6672638.

Rozsa A, Rudd EM, Boult TE. Adversarial diversity and hard positive generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops; 2016. p. 25–32.

Sandberg H, Teixeira A, Johansson KH. In: Preprints of the First Workshop on Secure Control Systems, CPSWEEK 2010, Stockholm, Sweden. On security indices for state estimators in power networks; 2010 http://www.truststc.org/conferences/10/CPSWeek. QC 20120206

Sedghi H, Jonckheere E. Statistical structure learning of smart grid for detection of false data injection. In: 2013 IEEE Power Energy Society General Meeting; 2013. p. 1–5. doi:10.1109/PESMG.2013.6672176.

Singh D, Pandey JP, Chauhan DS. Radial basis neural network state estimation of electric power networks, volume 1; 2004. p. 90–5. doi:10.1109/DRPT.2004.1338474.

Storn R, Price K. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. 1997;11(4):341–59. doi:10.1023/A:1008202821328.

Su J, Vargas DV, Sakurai K. One pixel attack for fooling deep neural networks. IEEE Trans. Evol. Comput. 2019;23(5):828–41. doi:10.1109/TEVC.2019.2890858.

Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, Fergus R. Intriguing properties of neural networks. International Conference on Learning Representations, 2014. http://arxiv.org/abs/1312.6199

Teixeira A, Dn G, Sandberg H, Johansson KH. A cyber security study of a SCADA energy management system: stealthy deception attacks on the state estimator*. IFAC Proceedings Volumes 2011;44(1):11271–7. doi:10.3182/20110828-6-IT-1002.02210.

Tramèr F, Papernot N, Goodfellow I, Boneh D, McDaniel P. The space of transferable adversarial examples. arXiv preprint arXiv:170403453 2017.

Wang J, Hui LCK, Yiu SM. System-state-free false data injection attack for nonlinear state estimation in smart grid. International Journal of Smart Grid and Clean Energy 2015;4(3). doi:10.12720/sgce.4.3.169-176. http://www.ijsgce.com/index.php?m=content&c=index&a=show&catid=51&id=234

Wood AJ, Wollenberg BF. Power generation, operation, and control second edition. John Wiley & Sons, Inc.; 1996.

Xie C, Zhang Z, Zhou Y, Bai S, Wang J, Ren Z, Yuille AL. Improving transferability of adversarial examples with input diversity. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

Yu JJQ, Hou Y, Li VOK. Online false data injection attack detection with wavelet transform and deep neural networks. IEEE Trans. Ind. Inf. 2018;14(7):3271–80. doi:10.1109/TII.2018.2825243.

Yuan X, He P, Zhu Q, Li X. Adversarial examples: attacks and defenses for deep learning. IEEE Trans Neural Netw Learn Syst 2019;30(9):2805–24. doi:10.1109/TNNLS.2018.2886017.

Zamzam AS, Fu X, Sidiropoulos ND. Data-driven learning-based optimization for distribution system state estimation. IEEE Trans. Power Syst. 2019;34(6):4796–805. doi:10.1109/TPWRS.2019.2909150.

Zamzam AS, Sidiropoulos ND. Physics-aware neural networks for distribution system state estimation. IEEE Trans. Power Syst. 2020;35(6):4347–56. doi:10.1109/TPWRS.2020.2988352.

Zhang F, Kodituwakku HADE, Hines JW, Coble J. Multilayer data-driven cyber-attack detection system for industrial control systems based on network, system, and process data. IEEE Trans. Ind. Inf. 2019;15(7):4362–9. doi:10.1109/TII.2019.2891261.

Zimmerman RD, Murillo-Snchez CE, Thomas RJ. Matpower: steady-state operations, planning, and analysis tools for power systems research and education. IEEE Trans. Power Syst. 2011;26(1):12–19. doi:10.1109/TPWRS.2010.2051168.

Tian Liu received her B.S. degree in Mathematics and Applied Mathematics from Sichuan University, China in 2011. She is currently pursuing the Ph.D. degree in the Department of Computer Science and Software Engineering at Auburn University. Her research interests focus on security and privacy issues in machine learning algorithms on IoT and CPS.

Tao Shu received the B.S. and M.S. degrees in electronic engineering from the South China University of Technology, Guangzhou, China, in 1996 and 1999, respectively, the Ph.D. degree in communication and information systems from Tsinghua University, Beijing, China, in 2003, and the Ph.D. degree in electrical and computer engineering from The University of Arizona,Tucson, AZ, USA, in 2010. He is currently an Associate Professor in the Department of Computer Science and Software Engineering at Auburn University, Auburn, AL. His research aims at addressing the security, privacy, and performance issues in wireless networking systems, with strong emphasis on system architecture, protocol design, and performance modeling and optimization.