Detecting False Data Injection Attacks on DC State Estimation*

Rakesh B. Bobba, Katherine M. Rogers, Qiyan Wang, Himanshu Khurana, Klara Nahrstedt and Thomas J. Overbye
University of Illinois, Urbana-Champaign
Email: {rbobba, krogers6, qwang26, hkhurana, klara, overbye}@illinois.edu

***Abstract*—State estimation is an important power system application that is used to estimate the state of the power transmission networks using (usually) a redundant set of sensor measurements and network topology information. Many power system applications such as contingency analysis rely on the output of the state estimator. Until recently it was assumed that the techniques used to detect and identify bad sensor measurements in state estimation can also thwart malicious sensor measurement modification. However, recent work by Liu *et al.* [1] demonstrated that an adversary, armed with the knowledge of network configuration, can inject false data into state estimation that uses DC power flow models without being detected. In this work, we explore the detection of false data injection attacks of [1] by protecting a strategically selected set of sensor measurements and by having a way to independently verify or measure the values of a strategically selected set of state variables. Specifically, we show that it is necessary and sufficient to protect a set of *basic measurements* to detect such attacks.**

I. Introduction

The power grid is a complex system of interconnected networks each of which consists of electric power generators and power consumers (loads) connected by transmission and distribution lines. To ensure safe and reliable operation of the power grid, each of the interconnected networks is continuously monitored and controlled by a control center[1] using an industrial control system known as Supervisory Control and Data Acquisition (SCADA) system. SCADA system collects measurements from sensors in the network, every 2 to 4 seconds. These sensor measurements are fed into a State Estimator which, as the name indicates, estimates the state of the power network based on the sensor measurements. Local grid operators use this estimate of the current state to take corrective control actions if necessary and to plan for any contingencies (*e.g.*, loss of a transmission line or generator). Thus state estimation plays an important role in the reliable operation of the power grid.

The power grid, being critical infrastructure, is an attractive attack target. Adversaries may attempt to manipulate sensor measurements, insert fake control commands, delay measurements and/or control commands, and resort to other malicious actions. Therefore, it is crucial to protect power system applications against such malicious activity to ensure safe and reliable operation of the power grid. Until recently, it was generally assumed that the techniques used to detect, identify and correct [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16] bad sensor measurements in state estimation are sufficient to detect and recover from sensor measurement manipulation. However, recent work by Liu *et al.* [1] demonstrated that an adversary, armed with the knowledge of the network configurations, can inject false data into state estimation that uses DC power flow models without being detected.

One of the key reasons behind such attack demonstrations is that current bad data detection techniques were designed to deal with errors and not coordinated malicious activity. Therefore, there is a need to develop advanced defense strategies for protecting state estimation and other power system applications. The aim of this work is to take the first step in this direction and develop defense strategies for protecting DC state estimation against the false data injection attacks proposed in [1]. While the work of Liu *et al.* [1] presented the false data injection attacks from an adversary's point of view and showed what it takes for an adversary to launch a successful attack, we look at the problem from the power grid operator's point of view and ask what it takes to defend against such attacks. Intuitively, there are two approaches to protecting control applications such as state estimation. The first is to design robust control algorithms that can detect or tolerate malicious data modification. The second is to protect the sensor measurements and other data from being manipulated. These two approaches are not necessarily mutually exclusive but can complement each other.

The first approach of handling malicious data injection at the application layer might mean reduced application efficiency with higher development costs. Furthermore, changing algorithms that the grid operators are used to and have gained significant experience with is not lightly done. More often than not, new algorithms are first introduced as research and development prototypes and are not commissioned for production use until the operators gain some experience and get comfortable with using the new algorithm. Thus, the second approach of fundamentally thwarting sensor data manipulation at the lower layer is the only alternative until the new algorithm is accepted for production use. This second approach can mean

*Presented at First Workshop on Secure Control Systems (SCS 2010), CPSWEEK2010, April 12th 2010, Stockholm Switzerland.

[1] In order to ensure reliability of the interconnected networks as a whole, designated entities known as reliability coordinators monitor the network over a wide region and provide oversight and reliability coordination between control centers.

simpler power applications and higher performance. However, it may not be feasible to protect all sensor measurements, either due to budgetary constraints or the legacy nature of the measurement device and its communications. In this work, we explore bringing application awareness to the second approach in order to reduce the burden of protecting all sensor measurements.

Specifically, we investigate whether it is possible to significantly reduce the risk of undetectable false data injection attacks of [1] against DC state estimation by 1) protecting a carefully chosen subset of the sensor measurements, and 2) having ways to independently verify or measure the values of a carefully chosen set of state variables, both for a given network topology. The intuition behind this approach is that for a given topology some sensor measurements influence more state variables than others and hence might provide better cost to benefit ratio when protected. Similarly, some state variables are dependent on more sensor measurements than others and hence independently verifying their estimate might limit the attackers ability in manipulating sensor measurements without being detected.

Such an analysis is very useful for state estimation, and any deployed control algorithm in general, as it allows grid operators to make informed decisions regarding how to invest their protection budget. Even when a grid operator is willing and has the resources to protect all sensor measurements and upgrade their state estimation algorithm, this change will not be effected overnight. Thus, our investigation is useful in prioritizing which sensor measurements to protect first from a security point of view. Besides, this approach is general in nature in the sense that it can be combined with solutions that handle malicious data at the upper layers.

Our results show that our defense strategy is very effective in thwarting undetectable false data injection attacks of [1] on DC state estimation. Our main contribution in this context is showing that protecting a set of *basic measurements* is a necessary and sufficient condition for detecting false data injection attacks of [1] on DC state estimation. A set of *basic measurements* is composed of the minimum number of measurements needed to ensure *observability* of the power network, *i.e.*, to ensure that the state variables can be estimated using the measurements. For DC state estimation, the size of a set of basic measurements is equal to the number of state variables, n, that need to be estimated, while the number of measurements, m, is often larger than that, *i.e.* $m > n$. For example, for the IEEE 300-bus test system, the number of measurements is 1122 while the number of state variables to be estimated is 299[2]. The additional measurements provide redundancy and are useful for traditional bad sensor measurement detection and identification methods mentioned above.

The rest of this paper is organized as follows. In Section II, we provide a brief background on state estimation and associated bad data detection mechanisms and the false data injection attacks proposed in [1]. We motivate and present our general approach in Section III. We discuss approaches to identifying a set of sensors and state variables to protect and then present our results in Section IV. We discuss practical issues, limitations and future directions in V.

TABLE I: Notations

m	The number of measurements
n	The number of state variables
$\mathbf{H}$	$m \times n$ Jacobian matrix representing the topology
$\mathbf{x}$	$n \times 1$ vector of state variables
$\mathbf{z}$	$m \times 1$ vector of measurements
$\mathbf{e}$	$m \times 1$ vector of measurement errors, s.t., $\mathbf{z} = \mathbf{Hx} + \mathbf{e}$
$\hat{\mathbf{x}}$	$n \times 1$ vector of estimated state variables
$\mathbf{W}$	$m \times m$ diagonal matrix, s.t., $w_{i,i} = \sigma_i^{-2}$, where σ_i^2 is the variance of the i-th measurement $(1 \leq i \leq m)$
τ	Threshold for the L_2-norm based detection of bad measurements
$\mathbf{z_a}$	$m \times 1$ measurement vector with bad measurements
$\mathbf{a}$	$m \times 1$ attack vector, s.t., $\mathbf{z_a} = \mathbf{z} + \mathbf{a}$
$\mathbf{c}$	$n \times 1$ vector of estimation errors introduced due to $\mathbf{a}$
$\mathcal{M}$	The set of sensor or measurement indices
$\mathcal{V}$	The set of state variable indices
$\mathcal{I}_{\bar{m}}$	The set of indices of protected sensor measurements
$\mathcal{I}_{\bar{v}}$	The set of indices of independently verified state variables
$\mathcal{I}_m$	The set of indices of potentially manipulated measurements
$\mathcal{I}_v$	The set of indices of potentially manipulated state variables
p	The number of protected measurements, i.e., $\lvert\mathcal{I}_{\bar{m}}\rvert$
q	The number of independently verified state variables, i.e., $\lvert\mathcal{I}_{\bar{v}}\rvert$

II. Background

In this section, we briefly discuss DC state estimation including bad data detection and the false data injection attacks proposed in [1]. Table I shows the notation used.

A. DC State Estimation [17]

Here, we present a common formulation of the state estimation problem when using a DC power flow model.

$$\mathbf{z} = \mathbf{Hx} + \mathbf{e} \tag{1}$$

In (1), $\mathbf{x} = (x_i, x_2, \ldots, x_n)^T$ represents the true states of the system that are to be estimated, $\mathbf{z} = (z_i, z_2, \ldots, z_m)^T$ represents the sensor measurements, $\mathbf{H}$ is an $m \times n$ Jacobian matrix where $\mathbf{Hx}$ is a vector of m linear functions linking measurement to states, and $\mathbf{e} = (e_i, e_2, \ldots, e_m)^T$ represents random errors in measurement.

The power network is considered *observable* if there are enough measurements to make state estimation possible. There are many sensor placement algorithms that can identify the set of sensor measurements that ensure observability of a power network [17]. Typically, there are more sensors in the power network than those needed for observability, *i.e.* $m > n$. The minimum set of measurements needed to estimate the n state variables is commonly referred to as a set of *basic measurements* or *essential measurements*. The remaining set of measurements are referred to as *redundant measurements*. The redundant measurements are useful in identifying bad sensor measurements [17]. Note that for DC state estimation, any set of n measurements whose corresponding rows in $\mathbf{H}$ are linearly independent are sufficient to solve for the n state

[2] Here we are excluding the slack bus angle

variables and hence constitute a set of *basic measurements*. In other words, n independent linear equations are sufficient to solve for n variables. When m is greater than n, as is the typical case, state estimation involves solving an over-determined system of linear equations. It can be solved as a weighted least squares problem to arrive at the following estimator:

$$\hat{\mathbf{x}} = (\mathbf{H^T W H})^{-\mathbf{1}} \mathbf{H^T W z} \tag{2}$$

where $\mathbf{W}$ is a diagonal matrix whose elements are the measurement weights. It is common to base $\mathbf{W}$ on the reciprocals of the variance of measurement error. As pointed out in [1], as long as the sensor measurement error is assumed to be normally distributed with zero mean, other commonly used estimation criteria, namely, maximum likelihood criterion and minimum variance criterion also lead to the estimator in (2).

1) Bad Measurement Detection: Sensor measurements used for state estimation might be inaccurate because of device misconfiguration, device failures, malicious actions or other errors and can adversely affect the estimate of state variables. Thus, it is extremely valuable for power system operators to detect the presence of bad measurements and identify them. Many schemes for detecting, identifying and correcting bad measurements have been proposed [18], [17].

A common approach [18], [17] for detecting the presence of bad data is by looking at L_2-norm of measurement residual which is defined as follows:

$$||\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}|| \tag{3}$$

In, equation (3), $\hat{\mathbf{x}}$ is the state estimate and $\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}$ is the measurement residual, which is the difference between the vector of observed measurements and estimated measurements. Intuitively, when observed measurements, $\mathbf{z}$, contain bad data, the $L_2 - norm$ of the measurement residual will be high. Thus if the value of expression in (3) is greater than a certain threshold τ it is assumed that bad data is present. Assuming that all state variables are mutually independent and that the sensor errors follow a normal distribution, it can be shown that $(\|\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}\|)^2$ follows a *chi-squared distribution* with $\nu = m - n$ degrees of freedom [18]. Threshold value τ can then be determined through a hypothesis test with a significance level α.

B. False Data Injection Attacks [1]

False data injection attacks on state estimation [1] are those in which an attacker[3] manipulates the sensor measurements to induce an arbitrary change in the estimated value of state variables without being detected by the bad measurement detection algorithm of the state estimator. In [1], Liu *et al.* present false data injection attacks that can bypass the bad measurement detection algorithm described in Section II-A1. Here, we summarize the basic attack principle, attack scenarios and goals from [1].

1) Attack Principle: Let $\mathbf{a} = (a_1, a_2, \ldots, a_m)^T$ be an attack vector representing the malicious data added to the original measurement vector $\mathbf{z} = (z_1, z_2, \ldots, z_m)^T$. Let $\mathbf{z_a} = \mathbf{z} + \mathbf{a}$ represent the resulting modified measurement vector. Let $\hat{\mathbf{x}}_{\mathbf{bad}}$ and $\hat{\mathbf{x}}$ represent the estimates of $\mathbf{x}$ when using the manipulated measurements $\mathbf{z_a}$ and original measurements $\mathbf{z}$ respectively. Then $\hat{\mathbf{x}}_{\mathbf{bad}}$ can be represented as $\hat{\mathbf{x}} + \mathbf{c}$, where $\mathbf{c}$ is the estimation error introduced by the attacker.

Theorem 1 in [1] shows that if the adversary chooses the attack vector, $\mathbf{a}$, to be equal to $\mathbf{Hc}$, then resulting manipulated measurement $\mathbf{z_a} = \mathbf{z} + \mathbf{a}$ can pass the bad measurement detection algorithm described in Section II-A1 as long as the original measurement $\mathbf{z}$ can pass it. To see this, consider the L_2-norm of the measurement residual with manipulated data

$$\begin{aligned}
\|\mathbf{z_a} - \mathbf{H}\hat{\mathbf{x}}_{\mathbf{bad}}\| \qquad &= \|\mathbf{z} + \mathbf{a} - \mathbf{H}(\hat{\mathbf{x}} + \mathbf{c})\| \\
&= \|\mathbf{z} - \mathbf{H}\hat{\mathbf{x}} + (\mathbf{a} - \mathbf{Hc})\| \qquad (4) \\
&= \|\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}\| \qquad (5) \\
&\text{when} \quad \mathbf{a} = \mathbf{Hc} \qquad (6)
\end{aligned}$$

2) Attack Scenarios and Goals or Adversary Model: It is assumed that the adversary has access to $\mathbf{H}$ which is determined by the power network topology and line impedances. It is also assumed that the adversary has the capability to manipulate sensors measurements, either by compromising the sensor or the communication between the sensor and the control center. However, this capability of the attacker is constrained as follows:

- **Scenario I:** The attacker is restricted to accessing only specific sensors. This takes into account the possibility that some sensors may be protected or beyond the reach of the attacker for other reasons.
- **Scenario II:** The attacker has limited resources to compromise sensors. That is, the attacker can compromise any sensor but is restricted to compromising only a limited number, say k, out of all sensors.

For both of the above scenarios, two attack goals are considered, namely, *random false data injection* and *targeted false data injection*. In *random false data injection*, the adversary aims to find any attack vector that injects arbitrary errors into the estimates of state variables. In *targeted false data injection*, the adversary aims to find an attack vector that injects specific errors into the estimates of specific state variables chosen by him. For *targeted false data injection*, two cases are considered: *constrained* and *unconstrained*. In the *constrained* case, the adversary aims to find an attack vector that injects specific errors into the estimates of specific state variables but does not pollute the estimates of other state variables. This case represents situations where the control center may have independent ways to verify the estimates of certain state variables, and to avoid detection, the adversary does not want to pollute them. In the *unconstrained* case, the adversary has no such concerns regarding polluting other state variables.

[3] We use the terms attacker and adversary interchangeably throughout this paper

Methods to identify attack vectors for both of the above described attack goals and in each of the above described attack scenarios, as well as the effectiveness of those methods on the IEEE 9-bus, 14-bus, 30-bus, 118-bus and 300-bus test systems, are presented in [1]. We refer the readers to [1] for details.

III. Motivation and Approach

While the work of Liu *et al.* [1] presented the false data injection attacks from an adversary's point of view and showed what it takes for an adversary to launch a successful attack, we look at the problem from the power grid operator's point of view and ask what it takes to defend such attacks. An obvious approach is to protect all sensor measurements from being manipulated. However, this is not always feasible, and in this work we explore the feasibility of detecting false data injection attacks without having to protect measurements from all sensors. Specifically, for a given $\mathbf{H}$, we aim to identify a set of sensors and a set of state variables such that, when the measurements from the sensors in the chosen set are protected and when the values of state variables from the chosen set can be verified independently, then an adversary cannot find attack vectors that can inject false data without being detected. Furthermore, we would like to identify the smallest of such sets.

The existence of a set of sensors such that, when the measurements from those sensors are protected, an adversary cannot inject false data without being detected is evident from the results in [1]. For a given integer k, Figure 2 in [1] shows the estimated success probability of an attacker in injecting false data without being detected when he picks k measurements at random to manipulate. This success probability was estimated using multiple trials of picking k measurements at random to manipulate. If the success probability of an attacker is less than 1 for a given k, it implies that there exist sets of $m - k$ measurements such that when they are protected an attacker cannot inject false data without being detected. For example, an adversary has to compromise close to 80% and 50% of the total measurements, for the IEEE 9-bus and 300-bus systems respectively, before his success probability approaches 1. That means there exist sets consisting of more than 20% and 50% of the sensors for the IEEE 9-bus and 300-bus systems respectively, such that when the measurements from those sensors are protected against compromise an adversary cannot inject false data without being detected.

However, it is useful to identify the smallest set of sensors that need to be protected for detecting false data injection attacks. According to Theorem 2 in [1], if an attacker can compromise k sensor measurements, where $k \geq m - n + 1$, there always exist attack vectors that can inject false data without being detected even when the attacker has no control over which k sensors he can compromise. This provides a lower bound on the number of sensors that need to be protected. That is, a necessary condition for detecting false data injection is protecting at least n sensors. However, it does not seem to be a sufficient condition. Results in [1] show that protecting any n out of m sensors doesn't guarantee detection of false data injection, and that sometimes more than n sensors need to be protected. For example, consider the IEEE 300-bus test system where there are $m = 1122$ measurements and $n = 299$ state variables[4]. For this system, according to Theorem 2 of [1], if the adversary compromises any $m - n + 1 = 824$ measurements, then he can always find an attack vector for *random false data injection* (without being detected). But, as mentioned above, experimental results presented in Figure 2 of [1] show that an attacker is able to find an attack vector with probability 1, *i.e.*, found an attack vector in all the trials, when manipulating about 50% of measurements, *i.e.* 561 measurements, picked at random. Thus, it seems the grid operator is forced to protect more than 561 measurements to detect false data injection, and even then, if the set of protected measurements is not carefully chosen, the attacker may still succeed in injecting false data without being detected.

While selectively protecting a little more than 50% of the total measurements is more cost-effective than having to protect all sensor measurements, we explore the possibility of further reducing this burden by leveraging the operators ability to independently verify the values of a few chosen state variables. Intuitively, the ability to independently verify the value of a state variable provides some measure of indirect protection for the sensor measurements that most influence the value of the state variable. One way to independently verify the value of state variables is through the deployment of Phasor Measurement Units (PMUs). PMUs can directly measure both the magnitude and phase angles of currents and voltage at a bus and the measurements are GPS timestamped. There are already about 200 networked PMUs deployed in North America and another 800+ are slated to be deployed with support from Department of Energy (DOE) Smart Grid Investment Grants. It might be better to use the measurements from these PMU devices as an independent way to verify the value of a state variable and potentially save on the cost of protecting measurements from multiple legacy sensors.

A. Adversary Model

We assume that the adversary has access to the topology matrix $\mathbf{H}$ which is determined by the power network topology and line impedances. We also assume that the adversary has the capability to manipulate sensors measurements, either by compromising the sensors or the communication between the sensors and the control center. However, the attacker is restricted to compromising the measurements from only specific sensors denoted by the set $\mathcal{I}_m$. This takes into account the fact that the remaining measurements are protected by the grid operator. Furthermore, as discussed above, we assume that the grid operator can independently verify the values of a few chosen state variables, denoted by the set $\mathcal{I}_{\bar{v}}$, and that

[4] Based on the topology matrix $\mathbf{H}$ of the IEEE 300-bus test system obtained from MATPOWER [19], a MATLAB package for solving power flow equations. All the topology matrices used in this work are obtained from MATPOWER package.

the adversary, in order to avoid detection, is constrained not to inject false data into those variables.

B. Detecting False Data Injection

Let $\mathcal{M}$ denote the set of measurement indices. Let $\mathcal{I}_{\bar{m}} = \mathcal{M} \setminus \mathcal{I}_m$ denote the set of indices of measurements that are protected by the grid operator. Let $\mathcal{V}$ denote the set of state variables indices. Let $\mathcal{I}_v = \mathcal{V} \setminus \mathcal{I}_{\bar{v}}$ denote the set of indices of state variables that the attacker may inject false data into.

Since the measurements of sensors in $\mathcal{I}_{\bar{m}}$ cannot be manipulated by the attacker, the corresponding elements a_i for $i \in \mathcal{I}_{\bar{m}}$ in the attack vector $\mathbf{a} = (a_1, a_2, \ldots, a_m)^T$ are zero. Similarly, if $\mathbf{c} = (c_1, c_2, \ldots, a_n)^T$ represents the estimation error introduced by the attack vector $\mathbf{a}$, then c_j for $j \in \mathcal{I}_{\bar{v}}$ are also zero. Thus, to launch a false data injection attack without being detected, the attacker needs to find an attack vector $\mathbf{a} = (a_1, a_2, \ldots, a_m)^T$ such that it satisfies the following three conditions:

$$\mathbf{a} = \mathbf{Hc} \tag{7}$$

$$a_i = 0 \qquad \text{for} \quad i \in \mathcal{I}_{\bar{m}} \tag{8}$$

$$c_j = 0 \qquad \text{for} \quad j \in \mathcal{I}_{\bar{v}} \tag{9}$$

On the other hand, from the grid operator's perspective, in order to ensure that false data injection attacks are always detected, the grid operator needs to identify a set of sensors, $\mathcal{I}_{\bar{m}}$, and a set of state variables, $\mathcal{I}_{\bar{v}}$, such that an adversary *cannot* find an attack vector that satisfies the above three conditions. Ideally, the operator should find the smallest such sets. How to select the set of sensors, $\mathcal{I}_{\bar{m}}$, and the set of state variables, $\mathcal{I}_{\bar{v}}$, is described in the following section.

IV. Identifying Optimal $\mathcal{I}_{\bar{m}}$ and $\mathcal{I}_{\bar{v}}$

A. Approach I: Brute-Force Search

In our first attempt at identifying optimal $\mathcal{I}_{\bar{m}}$ and $\mathcal{I}_{\bar{v}}$, we tried a straight forward brute-force approach. Let $p = |\mathcal{I}_{\bar{m}}|$ and $q = |\mathcal{I}_{\bar{v}}|$. The grid operator can pick at random a fixed q out of n state variables to populate $\mathcal{I}_{\bar{v}}$ and a fixed p out of m sensors to populate $\mathcal{I}_{\bar{m}}$, and check if any attack vectors that satisfy the above three conditions exist for this choice, as follows.

Let $\mathbf{H} = (\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n)$, where $\mathbf{h}_i$ denotes the ith column vectors of $\mathbf{H}$. Set $\mathbf{H}_s = (\mathbf{h}_{j_1}, \mathbf{h}_{j_2}, \ldots, \mathbf{h}_{j_{n-q}})$ and $c_s = (c_{j_1}, c_{j_2}, \ldots, c_{j_{n-q}})^T$ where $j_i \notin \mathcal{I}_{\bar{v}}$ for $1 \le i \le n-q$. Let $\mathbf{P}_s = \mathbf{H}_s(\mathbf{H}_s^T\mathbf{H}_s)^{-1}\mathbf{H}_s^T$ and $\mathbf{B}_s = \mathbf{P}_s - \mathbf{I}$. Then,

$$\begin{aligned}
\mathbf{a} = \mathbf{Hc} \Leftrightarrow \mathbf{a} &= \sum_{i \in \mathcal{I}_v} \mathbf{h}_i c_i + \sum_{j \in \mathcal{I}_{\bar{v}}} \mathbf{h}_j c_j = \mathbf{H}_s \mathbf{c}_s \\
&\text{since} \quad c_j = 0 \text{ for } j \in \mathcal{I}_{\bar{v}} \\
&\Leftrightarrow \mathbf{P}_s\mathbf{a} = \mathbf{P}_s\mathbf{H}_s\mathbf{c}_s \\
&\Leftrightarrow \mathbf{P}_s\mathbf{a} = \mathbf{H}_s\mathbf{c}_s = \mathbf{a} \\
&\Leftrightarrow \mathbf{P}_s\mathbf{a} - \mathbf{a} = 0 \Leftrightarrow (\mathbf{P}_s - I)\mathbf{a} = 0 \\
&\Leftrightarrow \mathbf{B}_s\mathbf{a} = 0
\end{aligned} \tag{10}$$

This means an attack vector $\mathbf{a}$ satisfies (10) if and only if it satisfies conditions (7) and (9). Now to take into account condition (8), let $\mathbf{B}_s = (\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m)$, where $\mathbf{b}_i$ $(1 \le i \le m)$ denote the column vectors of $\mathbf{B}_s$. Set $\mathbf{B'}_s = (\mathbf{b}_{i_1}, \mathbf{b}_{i_2}, \ldots, \mathbf{b}_{i_{m-p}})$ and $\mathbf{a'} = (a_{i_1}, a_{i_2}, \ldots, a_{i_{m-p}})^T$, where $i_r \notin \mathcal{I}_{\bar{v}}$ for $1 \le r \le m-p$. Then,

$$\begin{aligned}
\mathbf{B}_s\mathbf{a} = 0 &\Leftrightarrow \sum_{i \in \mathcal{I}_m} \mathbf{b}_i a_i + \sum_{j \in \mathcal{I}_{\bar{m}}} \mathbf{b}_j a_j = 0 \\
&\Leftrightarrow \mathbf{B'}_s\mathbf{a'} = 0 \\
&\text{since} \quad a_j = 0 \text{ for } j \in \mathcal{I}_{\bar{m}}
\end{aligned} \tag{11}$$

Thus, for a given topology matrix $\mathbf{H}$ and sets $\mathcal{I}_{\bar{m}}$ and $\mathcal{I}_{\bar{v}}$, to find an attack vector that can inject false data without being detected, an attacker needs to (1) compute $\mathbf{a'}$ that satisfies (11), and (2) set $\mathbf{a} = (0, \ldots, 0, a_{i_1}, 0, \ldots, 0, a_{i_2}, 0, \ldots, 0, a_{i_{m-p}}, 0, \ldots, 0,)^T$, where a_{i_r} occupy the appropriate places denoted by i_r for $1 \le r \le m-p$. Note that $\mathbf{B'}_s$ is a $m \times (m-p)$ matrix. If the rank of $\mathbf{B'}_s$ is $m-p$ then equation (11) has no non-zero solutions and thus no error can be injected into state estimation without being detected, but if rank of $\mathbf{B'}_s$ is less than $m-p$ then an infinite number of solutions exist. The operator thus needs to find the smallest possible sets of $\mathcal{I}_{\bar{m}}$ and $\mathcal{I}_{\bar{v}}$ such that the rank of $\mathbf{B'}_s$ is $m-p$ in order to be able to detect false data injection attacks of [1].

Such a brute-force approach to identifying $\mathcal{I}_{\bar{m}}$ and $\mathcal{I}_{\bar{v}}$ needs to search through $\binom{m}{p} * \binom{n}{q}$ combinations for a given choice of p and q, where $0 \le p \le m$ and $0 \le q \le n$. Thus the potential search space for finding the smallest possible sets is quite large. However, in practice an operator may not have an independent way to verify the estimated value of a state variable for more than 10% of the total state variables, *i.e.*, $q \le \frac{n}{10}$. Similarly, with a way to verify the estimated value of some state variables, it is most likely that one could detect false data injection attacks by protecting less than half the sensors, *i.e.*, $p \le \frac{n}{2}$. Furthermore, the lower bound of n on the number of sensor measurements that need to be protected (when there are no verifiable state variables), as indicated by Theorem 2 of [1], provides a good starting point around which to search for a solution.

We implemented this approach using Matlab and analyzed the IEEE 9-bus system. The results are summarized in Table II. For the IEEE 9-bus system, there are 8 state variables and 27 measurements, *i.e.* $m = 27$ and $n = 8$. Thus, according to the Theorem 2 of [1], when an adversary is allowed to compromise more than or equal to $m - n + 1 = 20$ sensors he can always find successful attack vectors that inject false data without being detected. As seen in Table II, when only 7 sensors are protected, *i.e.* 20 sensors are allowed to be compromised, and no state variables are verifiable, there are no defensible configurations. That is, when $\mathcal{I}_{\bar{v}}$ is null, there exists no set, $\mathcal{I}_{\bar{m}}$, of size 7 that can prevent an adversary from finding a successful undetectable attack vector. However, when 8 sensors are protected, there are 329245 or 14% of

TABLE II: Number of protected sensors and verifiable state variables needed to detect false data injection attacks for IEEE 9-bus system

Number of Protected Sensors	Number of Verifiable State Variables	Number of Defensible Configurations *i.e.* those that can detect attacks	Percentage of Defensible Configurations
7	0	0	0
8	0	329245	14%
9	0	1991771	35%
6	1	0	0
7	1	18954135	75%
6	2	12288444	62%

the total combinations of 8 sensors, that provide defensible configurations. That is, an adversary cannot inject false data without being detected when one of the 329245 possible sets of 8 sensors is selected as $\mathcal{I}_{\bar{m}}$ even when $\mathcal{I}_{\bar{v}}$ is null.

When one state variable is verifiable, then defensible configurations can be found even when only 7 sensors are protected, and it turns out that 75% of all the possible combinations (*i.e.* $\binom{27}{7} * \binom{8}{1}$) are defensible configurations. However, protecting any less than 7 sensors when there is only one verifiable state variable yields no defensible configurations. Thus, for the IEEE 9-bus system we do not seem to be gaining much in terms of reduction in the number of sensors to be protected by having a way to verify state variables. However, the number of defensible configurations increases considerably compared with the case where there are no verifiable state variables. This provides a lot of flexibility to the operator in terms of the set of sensors he can choose to protect.

While this approach was tractable for IEEE 9-bus system, the search space got very large for the IEEE 14-bus system even with small p and q. When we ran a parallelized version, using Matlab Parallel Computing Toolbox, of our algorithm with $p = 12$ and $q = 1$ on an Intel Xeon dual processor quad-core 64-bit machine, the analysis did not complete even after two days. It is worth noting that, given a $m \times m$ matrix $\mathbf{B}_s$ as in equation (10), and p, the number of sensors to be protected, the problem of identifying the set of protected sensors of size less than or equal to p such that an adversary cannot inject false data without being detected is NP-hard. To see this, let $\mathcal{U} = \{u_i\}$ denote the set of matrices such that (1) each u_i is a sub-matrix of $\mathbf{B}_s$ and contains no more than $m - p$ columns of $\mathbf{B}_s$, and (2) if u_i contains x columns, then $rank(u_i) \leq x - 1$. To find the set of measurements to be protected, we need to find a sub-matrix h of $\mathbf{B}_s$ such that, (1) it has no more than p columns, and (2) for each $u_i \in \mathcal{U}$, $h \cap u_i \neq \emptyset$, where $\cap$ between two sub-matrices returns the common columns in them. Clearly, this problem is reducible to the *hitting set problem* which is NP-complete.

B. Approach II: Protecting Basic Measurements

While existing approximate algorithms for the *hitting set problem* could have been leveraged to analyze larger IEEE test systems using the approach in the preceding section, we wanted to find a more intuitive solution. The alternate approach described below provides such a solution and leverages the concept of *basic measurements*.

For a given $\mathcal{I}_{\bar{m}}$ set of protected sensors, let $\mathbf{H}' = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_n)^T$, where $\mathbf{r}_i$ $(1 \leq i \leq m)$ are the row vectors of $\mathbf{H}'$, denote a Jacobian matrix obtained by re-arranging the rows of $\mathbf{H}$ such that the rows corresponding to the sensors in $\mathcal{I}_{\bar{m}}$ appear as the first p rows of $\mathbf{H}'$. Thus, $\mathcal{I}'_{\bar{m}}$ corresponding to $\mathbf{H}'$ is simply $\{1, 2, \ldots, p\}$. Then, equation (7) can be written as

$$\begin{bmatrix} \mathbf{a}'_p \\ \mathbf{a}'_k \end{bmatrix} = \begin{bmatrix} \mathbf{H}'_{pn} \\ \mathbf{H}'_{kn} \end{bmatrix} \begin{bmatrix} \mathbf{c}' \end{bmatrix} \tag{12}$$

In equation (12), $\mathbf{a}'$ and $\mathbf{c}'$ are appropriately re-arranged versions of $\mathbf{a}$ and $\mathbf{c}$ with $\mathbf{a}'_p$, and $\mathbf{a}'_p$ being column vectors of length p and $k = m - p$ respectively; $\mathbf{H}'_{pn}$ is a $p \times n$ matrix, and $\mathbf{H}'_{kn}$ is a $k \times n$ matrix. Taking equation (8) into account ($\mathbf{a}'_p$ becomes a zero vector) and splitting equation (12) into two matrix equations we arrive at the following:

$$\mathbf{0} = \mathbf{H}'_{pn}\mathbf{c}' \tag{13}$$

$$\mathbf{a}'_k = \mathbf{H}'_{kn}\mathbf{c}' \tag{14}$$

Let us for now assume that there are no verifiable state variables, *i.e.* $\mathcal{I}_{\bar{v}} = \emptyset$. Then, for an undetected false data injection attack to be possible, there must be a $\mathbf{c}'$ in the null space of $\mathbf{H}'_{pn}$ such that $\mathbf{a}'_k = \mathbf{H}'_{\mathbf{kn}}\mathbf{c}'$ is satisfied. Conversely, no attacks are possible if $\mathbf{H}'_{pn}$ has full column rank, *i.e.*, $rank(\mathbf{H}'_{pn}) = n$. Since the rank of a $m \times n$ matrix is always less than or equal to $min(m, n)$, $rank(\mathbf{H}'_{pn})$ can be equal to n only if $p \geq n$. However, $p \geq n$ does not guarantee the detection of attacks since the $rank(\mathbf{H}'_{pn})$ may still be less than n. This result is captured in the following corollary.

Corollary 4.1: It is necessary but not sufficient to protect at least n measurements in order to be able to detect false data injection attacks.

The above result is in line with Theorem 2 and experimental observations of [1]. In order for the rank of $\mathbf{H}'_{pn}$ to be equal to n, at least n rows of $\mathbf{H}'_{pn}$ should be linearly independent vectors. That is, $\mathbf{H}'_{pn}$, should contain rows corresponding to at least one set of what are referred to as *basic measurements* (refer to Section II-A). A set of *basic measurements* in state estimation is a minimum set of measurements which is sufficient to ensure *observability* (refer to Section II-A). The following theorem states this result.

Theorem 4.1: When there are no verifiable state variables, it is necessary and sufficient to protect a set of **basic measurements** *in order to be able to detect false data injection attacks.*

For DC state estimation, the size of the set of basic measurements is equal to the number of state variables which is n. The remaining $m - n$ measurements provide redundancy and help with bad measurement identification. Note that the choice of a set of basic measurements is not unique. The existence of multiple sets of basic measurements is obvious since if there are two independent measurements of a state variable, it does not matter which is taken to be the basic measurement and which is taken to be redundant. Thus, the optimal number of sensor measurements to protect in order to detect false data injection attacks is n.

Theorem 4.1 seems to contradict the findings in Figure 2 of [1] and as such needs some clarification. As discussed in Section III, Figure 2 in [1] shows the estimated success probability of an attacker in injecting false data without being detected when he picks k measurements to manipulate at random. Figure 2 of [1] shows that, the probability of success of an adversary is 1, *i.e.*, always able to find an attack vector, even when about 561 (> 299) and 171 (> 117) measurements are protected in IEEE 300-bus and 118-bus test systems respectively. This apparent contradiction is due to the fact that the success probability shown was an estimated value, estimated using multiple trials of picking k measurements at random to manipulate. We observe that the discrepancy is very stark only for IEEE 300-bus and 118-bus systems and not for the other systems, namely IEEE 9-bus, 14-bus and 30-bus, that were also analyzed in [1]. We also note that only 100 trials were used in estimating success probabilities for the IEEE 300-bus and 118-bus systems, in order to reduce simulation time, as opposed to using 1000 trials as was done for the other smaller bus systems. Given the large search space for the IEEE 300-bus and 118-bus systems and the small number of trials used, it is very likely that the set of protected measurements picked by the simulations in the small number of trials did not contain a set of basic measurements. In fact, for IEEE 9-bus, 14-bus, 30-bus, 118-bus and 300-bus test systems, we picked a set of basic measurements (using the method outlined in Section IV-B1) and verified that there are no attack vectors, *i.e.*, the rank of $\mathbf{B'}_s$ in equation (11) is 0. Thus, the probability of success of an attacker cannot be 1 when the number of protected measurements is greater than n.

Now suppose we also have verifiable state variables. Without loss of generality, let us say we have only one verifiable state variable and it is the jth state variable. Taking equation (9) into account, equations (13) and (14) can be written as

$$\mathbf{0} = \mathbf{H''}_{pn'}\mathbf{c''} \tag{15}$$

$$\mathbf{a'}_k = \mathbf{H''}_{kn'}\mathbf{c''} \tag{16}$$

where $\mathbf{H''}_{pn'}$ and $\mathbf{H''}_{kn'}$ are derived from $\mathbf{H'}_{pn}$ and $\mathbf{H'}_{kn}$ respectively by removing the jth column, $n' = n - 1$ and $\mathbf{c''}$ is a column vector of length n' derived from $\mathbf{c'}$ by removing the jth element. If $\mathbf{H'}_{pn}$ was a full column rank matrix, *i.e.*, rank n matrix, then $\mathbf{H''}_{pn'}$ will also be a full column rank matrix, *i.e.*, rank $n - 1$ matrix, and thus no undetectable false data injection attacks are possible. Since $p \geq n$, it is possible to remove a few measurements from the protected measurements without compromising attack detectability as long as (1) p', the size of the resulting set of protected measurements, is equal to $n-1$ and (2) the resulting set of measurements are sufficient to ensure observability of the $n-1$ state variables (*i.e.*, excluding the jth state variable).

Corollary 4.2: If there are q verifiable state variables it is necessary and sufficient to protect a set of basic measurements *corresponding to the remaining $n - q$ state variables in order to be able to detect false data injection attacks.*

Thus, while a protected basic measurement may be replaced by a verifiable state variable, it is clear that the minimum required number of protected or verifiable quantities is equal to n, *i.e.*, the number of state variables.

1) Determining the Protected Set: The importance of protecting a set of basic measurements has been made clear. We now discuss how a defender of the system can identify such a set of measurements. Much work has been done to determine sensor placement for observability of a power network, *i.e.*, to determine a set of basic measurements, including [20], [21], [22], [23], [17], [24]. Another straight forward but brute-force approach is to pick a set of n measurements out of m at random and see if the rows corresponding to them in $\mathbf{H}$ are linearly independent. In this work however, we leverage a more computationally efficient approach described and justified in [16] and [25] respectively. In this approach, the measurements in the system are mapped to a new equivalent state space where identification of basic and redundant measurements is easily accomplished. This approach is briefly described below.

To obtain the equivalent states, an LU decomposition is performed on $\mathbf{H}$,

$$\widetilde{\mathbf{H}} = \mathbf{P} \cdot \mathbf{H} = \mathbf{L_{AA}} \cdot \mathbf{U_b} \tag{17}$$

where $\mathbf{P}$ is a row permutation matrix which maps the original rows of $\mathbf{H}$ to the new rows of $\widetilde{\mathbf{H}}$. The new basis is given by

$$\mathbf{L'_{AA}} = \begin{bmatrix} \mathbf{I_n} \\ \mathbf{R} \end{bmatrix} \tag{18}$$

where $\mathbf{I_n}$ is the $n \times n$ identity matrix. Rows of $\mathbf{I_n}$ correspond to the n basic measurements, and rows of $\mathbf{R}$ correspond to redundant measurements. Columns correspond to equivalent states. We compute the LU decomposition of $\mathbf{H}$ and use $\mathbf{P}$ to map the first n measurements in the new basis back to the original measurements. This gives us one set of basic measurements.

Other basic measurement sets may be derived after the first one is found. Furthermore, the matrix $\mathbf{L'_{AA}}$ can tell us which measurements we may switch out. As an example, consider this $\mathbf{L'_{AA}}$ from [16]:

$$\mathbf{L}'_{\mathbf{AA}} = \begin{array}{c} p_2 \\ p_3 \\ p_{24} \\ p_{12} \\ p_{34} \\ p_{23} \end{array} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0.5 & 0.5 & 0 \\ 0 & -0.5 & 0.5 & 0 \end{pmatrix}$$

The basic measurements are (p_2, p_3, p_{24}, p_{12}), but p_3 and p_{24} could be replaced by either p_{34} or p_{23}. Thus, another basic measurement set is (p_2, p_{34}, p_{23}, p_{12}). The key is that the rows switched out of the redundant measurement set must be linearly independent, otherwise they could not both be made into basic measurements. Following this line of reasoning, in an incremental manner, we can switch a basic measurement with one of its redundant measurements in $\mathbf{H}$ and recompute $\mathbf{L}'_{\mathbf{AA}}$ to obtain a new set of basic measurements. Note that the LU decomposition can be computed quickly, even for large systems. This is beneficial, especially when compared to a brute-force search of the entire space for meters and state variables to protect. We implemented this approach of identifying a basic set of measurements using Matlab, and identified multiple sets of basic measurements for the the IEEE 9-bus, 14-bus, 30-bus, 118-bus and 300-bus test systems. We verified that no attack vectors exist when these sets of measurements are protected using the approach described in Section IV-A.

In summary, there are many choices of sets of minimal size n which may be protected. Ultimately, the choice comes down to the interests of the defenders or owners of the system. The owners may have particular measurements that they would like to include in a basic measurement set if possible, or they may already know which particular state variables will be made verifiable. Their selection process may proceed as follows: (1) Determine a satisfactory set of basic measurements for the system using the approach described above. This is the candidate set of protected items. (2) Decide which state variables will be made verifiable. Add these state variables to the candidate set, and optionally remove an equal number of protected measurements.

V. Discussion and Future Directions

Protecting Sensor Measurements: So far, we have focused on identifying a set of sensors whose measurements need to be protected in order to detect the false data injection attacks of [1]. Here, we discuss what it means to protect sensor measurements in this context. Clearly, we want the measurements from the sensors be authentic. That means manipulation of the sensor measurements either by 1) physically tampering with the device or 2) by tampering with the communication between the sensor and control center needs to be prevented. That is, sensors should be protected from unauthorized access (both physical and remote access), and measurements from the sensor should be authenticated and integrity protected. However, this may not be sufficient if one would like to only protect measurements in the smallest required set. This is because the operator may not be able to detect false data injection attacks when a measurement from one of the protected sensors is unavailable. Thus, it is also essential that the measurements from the protected sensors be available at all times. This latter requirement may be relaxed a bit by protecting a few more strategically selected sensors than the smallest set necessary. Identification of this larger set of sensors and studying the associated trade-offs are left to be addressed in a future work.

Considering Topology Changes: In this work, we focus on identifying measurements and state variables to protect, for a given $\mathbf{H}$, but do not consider how topology changes such as line outages would affect these decisions. In reality, the defender of the system needs to deploy a protected set so that in the event of any expected topology change, false data injection attacks are still detectable.

If any line l is opened, and that line has a protected measurement, the measurement is no longer valid. Thus, the corresponding protected measurement row in $\mathbf{H}'_{pn}$ needs to be replaced. Otherwise, the resulting $\mathbf{H}'_{pn}$ is not full rank, and attacks are possible. The measurement m that replaces the lost measurement must have a row in $\mathbf{H}$ which is linearly independent with the remaining protected measurement rows in $\mathbf{H}'_{pn}$, so that when it is added, the resulting $\mathbf{H}'_{pn}$ is again full rank. In this case, the set of measurements required for the system to be protected before and after a line outage of l is of size $(n+1)$. Each considered line outage will thus increase the size of the required protected set by at least one. Suppose that there is a basic measurement set $\{1, 2, 3, 4, 5\}$ and another basic measurement set which is $\{2, 3, 4, 5, 6\}$. If measurement 1 is lost, it can be replaced by measurement 6, and the system will be protected. For the system to be protected both with and without line 1, measurements $\{1, 2, 3, 4, 5, 6\}$ must all be protected. As before, one basic measurement may be substituted for one verifiable state variable. There are different ways [24], [26] to identify a set of measurements such that full observability is maintained with most common/frequent topology changes.

Generic False Data Injection Attacks: So far in this work, we have focused on strategies to detect false data injection attacks proposed in [1]. The basic principle behind such attacks, as discussed in Section II-B1, is that when the adversary sets his attack vector $\mathbf{a}$ to be equal to $\mathbf{Hc}$, then the bad measurement detection algorithm described in Section II-A1 fails to detect attacks. However, this is not the only way to inject false data without being detected. To see this, consider the equation (4). It is clear from equation (4) that, even if $\mathbf{a} \neq \mathbf{Hc}$, as long as the adversary chooses his attack vector, $\mathbf{a}$, such that the following equation (19) is true then the attacker could still inject false data without being detected.

$$\|\mathbf{z} - \mathbf{H}\hat{\mathbf{x}} + (\mathbf{a} - \mathbf{Hc})\| \leq \tau \qquad (19)$$

However, in this case, apart from knowing $\mathbf{H}$, the adversary has to know $\mathbf{z}$, *i.e.* values of all sensor measurements, and $\hat{\mathbf{x}}$. The adversary can compute $\hat{\mathbf{x}}$ using equation (2) but then needs to know $\mathbf{W}$. Thus, this form of false data injection attack imposes higher burden on the adversary than the one described in [1]. Furthermore, it may be possible to protect against these attacks by protecting the confidentiality of sensor measurements, *i.e.* preventing the adversary from knowing $\mathbf{z}$. Thus, by incorporating the requirement of confidentiality into our definition of "sensor measurement protection" discussed above, we might be able to detect generic false data injection attacks. A detailed analysis of such attacks and defense strategies will be the subject of future work.

Acknowledgements

The authors would like to thank Yao Liu for sharing with us the Matlab code used in the analysis of [1]. The authors would also like to thank William H. Sanders, Tim Yardley and the anonymous reviewers for their helpful comments. This material is based upon work supported by the National Science Foundation under Grant No. CNS-0524695. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

[1] Y. Liu, M. K. Reiter, and P. Ning, "False data injection attacks against state estimation in electric power grids," in *CCS '09: Proceedings of the 16th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2009, pp. 21–32.

[2] H. Merrill and F. Schweppe, "Bad data suppression in power system static state estimation," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-90, no. 6, pp. 2718–2725, Nov. 1971.

[3] E. Handschin, F. Schweppe, J. Kohlas, and A. Fiechter, "Bad data analysis for power system state estimation," *Power Apparatus and Systems, IEEE Transactions on*, vol. 94, no. 2, pp. 329–337, Mar 1975.

[4] D. Falcao, P. Cooke, and A. Brameller, "Power system tracking state estimation and bad data processing," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-101, no. 2, pp. 325–333, Feb. 1982.

[5] W. Kotiuga and M. Vidyasagar, "Bad data rejection properties of weughted least absolute value techniques applied to static state estimation," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-101, no. 4, pp. 844–853, April 1982.

[6] X. Nian-de, W. Shi-ying, and Y. Er-keng, "A new approach for detection and identification of multiple bad data in power system state estimation," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-101, no. 2, pp. 454–462, Feb. 1982.

[7] A. Monticelli and A. Garcia, "Reliable bad data processing for real-time state estimation," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-102, no. 5, pp. 1126–1139, May 1983.

[8] T. Van Cutsem, M. Ribbens-Pavella, and L. Mili, "Hypothesis testing identification: A new method for bad data analysis in power system state estimation," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-103, no. 11, pp. 3239–3252, Nov. 1984.

[9] X. N. de, W. Shi-ying, and Y. Ers-keng, "An application of estimation-identification approach of multiple bad data in power system state estimation," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-103, no. 2, pp. 225–233, Feb. 1984.

[10] W. Peterson and A. Girgis, "Multiple bad data detection in power system state estimation using linear programming," in *System Theory, 1988., Proceedings of the Twentieth Southeastern Symposium on*, Mar 1988, pp. 405–409.

[11] F. Wu, W.-H. Liu, and S.-M. Lun, "Observability analysis and bad data processing for state estimation with equality constraints," *Power Systems, IEEE Transactions on*, vol. 3, no. 2, pp. 541–548, May 1988.

[12] I. Slutsker, "Bad data identification in power system state estimation based on measurement compensation and linear residual calculation," *Power Systems, IEEE Transactions on*, vol. 4, no. 1, pp. 53–60, Feb 1989.

[13] A. Abur, "A bad data identification method for linear programming state estimation," *Power Systems, IEEE Transactions on*, vol. 5, no. 3, pp. 894–901, Aug 1990.

[14] B. Zhang, S. Wang, and N. Xiang, "A linear recursive bad data identification method with real-time application to power system state estimation," *Power Systems, IEEE Transactions on*, vol. 7, no. 3, pp. 1378–1385, Aug 1992.

[15] E. Asada, A. Garcia, and R. Romero, "Identifying multiple interacting bad data in power system state estimation," in *Power Engineering Society General Meeting, 2005. IEEE*, June 2005, pp. 571–577 Vol. 1.

[16] J. Chen and A. Abur, "Placement of pmus to enable bad data detection in state estimation," *Power Systems, IEEE Transactions on*, vol. 21, no. 4, pp. 1608–1615, Nov. 2006.

[17] A. Monticelli, *State estimation in electric power systems: a generalized approach*. Kluwer Academic Publishers, 1999.

[18] A. Wood and B. Wollenberg, *Power Generation, Operation, and Control*, 2nd ed. John Wiley and Sons, 1996.

[19] R. Zimmerman, C. Murillo-Sanchez, and R. Thomas, "Matpower's extensible optimal power flow architecture," july 2009, pp. 1 –7.

[20] V. Quintana, A. Simoes-Costa, and A. Mandel, "Power system topological observability using a direct graph-theoretic approach," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-101, no. 3, pp. 617 –626, march 1982.

[21] E. Fetzer and P. Anderson, "Observability in the state estimation of power systems," *Power Apparatus and Systems, IEEE Transactions on*, vol. 94, no. 6, pp. 1981 – 1988, nov. 1975.

[22] G. Krumpholz, K. Clements, and P. Davis, "Power system observability: A practical algorithm using network topology," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-99, no. 4, pp. 1534 –1542, july 1980.

[23] A. Monticelli and F. Wu, "Network observability: Identification of observable islands and measurement placement," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-104, no. 5, pp. 1035 –1041, may 1985.

[24] A. Abur and F. H. Magnago, "Optimal meter placement for maintaining observability during single branch outages," *IEEE Transactions on Power Systems*, vol. 14, no. 4, pp. 1273 – 1278, 1999, least absolute value estimations;. [Online]. Available: http://dx.doi.org/10.1109/59.801884

[25] J. London, L. Alberto, and N. Bretas, "Analysis of measurement-set qualitative characteristics for state-estimation purposes," *Generation, Transmission Distribution, IET*, vol. 1, no. 1, pp. 39 –45, January 2007.

[26] A. Abur and F. Magnago, "Optimal meter placement against contingencies," vol. 1, no. SUMMER, Vancouver, BC, Canada, 2001, pp. 424 – 428, network observability;Optimal meter placement;. [Online]. Available: http://dx.doi.org/10.1109/PESS.2001.970061