5-2018

Low Latency Intrusion Detection in Smart Grids

Israel Zairi Akingeneye
University of Arkansas, Fayetteville

Low Latency Intrusion Detection in Smart Grids

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Engineering with a concentration in Electrical Engineering

by

Israel Akingeneye
University of Arkansas
Bachelor of Science in Electrical Engineering, 2013

May 2018
University of Arkansas

This dissertation is approved for recommendation to the Graduate Council.

Jingxian Wu, Ph.D.
Dissertation Director

Roy A. McCann, Ph.D.
Committee Member

Qinghua Li, Ph.D.
Committee Member

Yue Zhao, Ph.D.
Committee Member

Abstract

The transformation of traditional power grids into smart grids has seen more new technologies such as communication networks and smart meters (sensors) being integrated into the physical infrastructure of the power grids. However, these technologies pose new vulnerabilities to the cybersecurity of power grids as malicious attacks can be launched by adversaries to attack the smart meters and modify the measurement data collected by these meters. If not timely detected and removed, these attacks may lead to inaccurate system state estimation, which is critical to the system operators for control decisions such as economic dispatch and other related functions.

This dissertation studies the challenges associated with cyberattacks in power grids and develops solutions to effectively and timely detect these attacks to ensure an accurate state estimation. One of the common approaches to improving the state estimation accuracy is to incorporate phasor measurement unit (PMU) devices into the system to provide extra and more secure measurements. In this research, we design algorithms that place PMUs at strategic locations to enhance the system's state estimation accuracy and its capability to detect cyberattacks. This approach of installing PMU devices in power grids, nonetheless, does not guarantee a timely attack detection that is critical for a timely deployment of countermeasures to prevent catastrophic impacts from the attacks. Thus, the low latency intrusion detection problem is studied to reduce attack detection delays. The state estimation and intrusion detection problem is further extended to a dynamic power system, where there are sudden changes in system loads.

Acknowledgements

This was a long project with lots of complex problems. But with help from lots of people, it seemed far simpler than it turned out to be. I would like to gratefully and sincerely express my gratitude to all the people who helped to complete this dissertation.

First and foremost, I would like to thank God for giving me the strength, courage, and knowledge to pursue this research study. This journey initially seemed long and arduous. But He made it short and enjoyable. Without Him this would have been impossible.

Second, I am forever grateful to my family for their love and support that kept me afloat when I thought I was drowning. They have always been the motivation and force that kept me believing in the end of this journey every step of the way.

Third, I would like to give my deepest appreciation to my supervisor, Dr. Jingxian Wu whose invaluable advice and recommendations ignited my desire to pursue the truth in the area of communication technologies. I will forever be indebted to him for his intelligent guidance, patience, and persistent insights that helped bring this dissertation to fruition.

Last but not least, I am extremely grateful to my dissertation committee members, Dr. Roy A. McCann, Dr. Qinghua Li, and Dr. Yue Zhao. Their careful reviews and valuable comments have helped me significantly revise and improve the quality of the dissertation.

Contents

List of Figures

List of Papers

- **Chapter 2**, Israel Akingeneye, Jingxian Wu, and Jing Yang, "Optimum PMU Placement for Power System State Estimation," published in *Proceedings IEEE Power & Energy Society General Meeting* , pp. 1-5, July, 2017.

- **Chapter 3**, Israel Akingeneye, Jingxian Wu, and Jing Yang, "PMU-Assisted Bad Data Detection in Power Systems," accepted in *IEEE Power & Energy Society Conference & Exposition on Transmission & Distribution* , April, 2018.

- **Chapter 4**, Israel Akingeneye and Jingxian Wu "Low Latency Detection of Sparse False Data Injections in Smart Grids," submitted to *IEEE Transaction on Smart Grid*, 2017.

Chapter 1

Introduction

1.1 Background and Motivation

The increasing energy demand from businesses, schools, hospitals, and residences requires a reliable energy supply by the utilities around the globe. The power grid, a critical infrastructure for energy supply needs constant monitoring by the utility operators to ensure that the dynamic energy demand is met without exceeding the power generation limits. The power grid operators use power grid state estimation to constantly track the state, bus voltage magnitudes and phases, of the system and use the estimated state variables to determine the power flow and power injections of the system. Moreover, modern power grids are integrated with communication networks and smart meters (sensors), which can be susceptible to malicious cyberattacks by an adversary. These malicious attacks cannot merely modify the measurements acquired by the smart meters but can also cause erroneous state estimations, which could result into costly decisions by the SCADA. The control system, SCADA system, makes decisions to increase or decrease the amount of power injected at certain buses and/or the power flows at certain transmission lines based on the current state of the system obtained through state estimations. The wrong decisions could, for instance, cause to decrease the power generation below the demand levels which could cause power outage to certain clients. It could also cause to increase the power generation at certain buses and power flows at some transmission lines and result in power overflows at these lines.

Considering this, there is a necessity for consistent upgrade of the state estimation and malicious attack detection methods in the power grid as the new technologies get integrated in the systems and bring new challenges. This dissertation studies various approaches of detecting the attacks and enhancing the accuracy of the state estimation. First, we develop techniques to strategically incorporate new devices, phasor measurement unit (PMU) devices, which provide extra and more secure measurements into the system. The PMU measurements are then used along with the traditional measurements to maximize the attack detection probability and minimize the state estimation error. Second, we design low latency attack detection algorithms that can minimize the detection delay to ensure the timely deployment of countermeasures to prevent catastrophic impacts from the attack.

Since the early 1980s, PMUs have been installed in many power systems around the globe [15] to enhance the robustness of power grid state estimations and malicious attack detection. A PMU device installed at a certain bus can provide accurate measurements of the voltage phasor at the bus and the current phasors of the branches incident to that bus [18]. The state estimator benefits from the PMU data by using these more accurate data along with the conventional measurements to improve the state estimation performance. However, due to the financial limitations of the utilities and the cost associated with the installation of these devices in the power grid, there are usually far less PMUs than buses. Therefore, one of the critical problems faced by power system design is PMU placement, that is, identifying the buses on which the PMUs should be installed.

Many existing approaches seek to solve the PMU placement problem by converting the system's critical measurements into redundant ones, thus to render it fully observable. The critical measurements, as opposed to the redundant measurements, are those measurements

whose removal in the system results in the system being unobservable [2] and [7]. In other words, an error in a redundant measurement will have very little effect on the state estimation while an error in critical measurement can degrade the entire state estimation and is very difficult to detect with the common statistical tests based on measurement residual [1].

The critical measurement based PMU placement approach has been adopted in the development of numerous existing algorithms, such as [15], [18], [7], and [1]. These works thoroughly explained how the critical measurements can be converted into redundant measurements to improve system observability. While increasing the system observability can enhance the state estimation and malicious attack or error detection, these works did not study the extent at which the observability obtained from a PMU addition into the system can improve the state estimation and malicious attack detection of the system.

In our research, we develop PMU placement algorithms that aim at improving the accuracy of power state estimation and algorithms that increase the capabilities of the system in detecting malicious attacks. For the designed state estimation algorithms, the design metric is the MSE, which can give insight on the estimation accuracy gained with each PMU installation. For the designed malicious attack detection algorithms, the probability of detection is used as a measure of how much gain obtained with each PMU installation. Our algorithms are quite different from the previous approaches that are based on the critical measurements in that they can almost guarantee a certain gain in terms of state estimation accuracy and/or malicious attack detection accuracy for each installed PMU.

The designed PMU placement approaches focus on maximizing the attack detection probability and the state estimation accuracy. However, a high attack detection probability alone may not guarantee a short detection delay without which there is no timely deployment

of countermeasures to prevent catastrophic impacts from the attack. To tackle this challenge, this dissertation studies the low latency detection problem to minimize the attack detection delay, the time difference between the occurrence and detection of the attack.

There are limited works on low latency detection of malicious attacks in smart grids. A generalized cumulative sum (CUSUM) detector is proposed in [13] for false data detection, where the generalized likelihood ratio test (GLRT) is utilized to estimate the unknown parameters. The complexity of the generalized CUSUM detector grows exponentially with the number of meters. The complexity mainly arises from the need to identify the meters under attack. A low complexity approximation of the generalized CUSUM is developed in [13], where each meter tracks the false data injection separately. In [8, 9], an adaptive multi-thread CUSUM algorithm is proposed for false data detection in power grids. It is pointed out in [9] that the complexity of GLRT might be too high for practical implementation, thus the Rao test is used for unknown parameter estimation. The elements in the attack vector are assumed to be positive in [8], and such assumption is not always true in practical attacks.

For a large power grid with a large number of buses and meters, it is extremely difficult, if not impossible, for an attacker to attack all meters at once. In almost all cases the attacker can modify the measurements from a small number of meters, that is, the attack is sparse among meters [11]. In recognition of the sparse nature of false data injections, we design a new orthogonal matching pursuit (OMP) CUSUM algorithm, which utilizes sparse recovery to identify the meters under attack. In the OMP-CUSUM algorithm, the attack vector is modeled as a sparse vector with dimension equal to the number of power measurements in the grid. The indices of the non-zero elements of the attack vector correspond to meters under attack, and the number of non-zero elements is called the sparsity level. A naive

way to locate the meters under attack will be to perform exhaustive search of all possible combinations of attack patterns with GLRT, the complexity of which grows exponentially with the number of buses. To reduce the complexity, we resort to the OMP algorithm [17, 14, 4, 5], which is a well known algorithm for sparse signal recovery. Given the fact that neither the sparsity nor the support of the attack vector is known, we develop a new stopping condition for the OMP algorithm by analyzing the statistical properties of the measurements in the grid. The stopping condition can accurately terminate the iterative OMP procedure once all meters under attack are successfully identified, without the prior knowledge of the sparsity level. The results of the OMP are then used in the CUSUM algorithm to minimize the detection delay of false data injection, subject to constraints on the detection accuracy and probability of false alarm. The OMP algorithm and CUSUM is combined in an iterative and sequential manner, that is, for each new group of measurements, OMP is used to estimate the support of the attack vector, and the results are then used for the sequential CUSUM test. Theoretical analysis and simulation results show that the newly proposed OMP-CUSUM algorithm can efficiently and promptly detect false data injections with low complexity, low detection delays, and good detection accuracy.

All of these state estimation and false data detection methods described in the previous paragraphs of this section assume a static system model, where the system is in a steady state and its measurements are quasi-static over time. In reality, though, the state of a power system changes with time due to the dynamic nature of system loads [3]. Therefore, state estimation and false data detection methods need a dynamic model to track the time evolution of the system states, which can be used to detect and replace corrupt measurements in the system. A dynamic state estimator can capture the system transients due to sudden

system changes faster and more accurately than its static counterpart does. A dynamic state estimator owes these properties to its capability to use past state estimations to predict the future state of the system one step ahead. The predicted states can be used to initialize the state estimation algorithm during the next step and detect measurements that deviate from these predictions. A mismatch between newly collected measurements and their predicted values indicates that there has been sudden changes in the system such as a loss of a large load and changes in network configurations, or malicious attacks that have modified some system measurements. It is necessary to detect and identify these malicious attacks in order to replace the corrupt measurements before they are processed by the state estimator.

The problem of dynamic state estimation has been studied before in [3, 6, 16, 12, 10]. These works use different versions of an extended Kalman filter (EKF) to perform dynamic state estimation by filtering the predicted state variables. All these algorithms utilize an amplitude test on the innovation vector, difference vector between the newly collected measurements and their predictions, to test the presence of false data and sudden changes in the system. Once the magnitude of the innovation vector exceeds a certain threshold, a flag is raised indicating that there is a sudden change in the system's operating point or false data injection attacks on the system. The false data are discriminated from sudden system changes by analyzing correlated measurements in the region near the abnormality and if the correlated measurements simultaneously fail the detection test, a sudden change is characterized. Otherwise, the suspected measurements contain false data and they are replaced with their predictions. This method of discriminating attacks from sudden change in the system operating point, however, may not be effective if the attacks are simultaneously injected in the correlated measurements. This may lead to a mischaracterization of the attacks as

sudden changes, and therefore, fail to remove and prevent the corrupt measurements from entering the state estimation stage.

In this research study, we propose a new detection algorithm that can accurately detect the bad data and discriminate them from sudden changes in the system. Based on the statistical distribution of the innovation vector, a hypothesis test is developed to study the system behavior with and without false data injections. From the hypothesis test, a chi-square test is then designed to detect the attacks. Once the false data are detected, corrupt measurements are identified and replaced with their predictions and then forwarded to the state estimator. Theoretical analysis and simulation results show that the newly proposed detection algorithm can effectively detect and replace false data injections including those injected in correlated measurements.

1.2 Objectives

The goal of this dissertation is to design low latency attack (intrusion) detection algorithms that maximize the detection probability and state estimation accuracy. These algorithms are developed in four folds.

First, PMU placement algorithms are designed with an objective of finding the best PMU locations that minimize the state estimation MSE or equivalently maximize the state estimation accuracy. The state estimation MSE is expressed as an explicit function of the locations of the PMUs and the problem is then formulated as a combinatorial optimization problem. The best PMU locations are obtained by solving the combinatorial optimization problem. The simulation results compare the MSE performance of these developed algorithms with

some of the most common PMU placement algorithms that are based on critical measurements. Our developed algorithms exhibit a better MSE performance than the common PMU placement algorithms.

Second, PMU placement algorithms are designed with an objective of finding the best PMU locations that maximize the probability of malicious attack detection. Similar to the first objective, the probability of malicious attack detection is expressed as an explicit function of the locations of the PMUs and the problem is then formulated as a combinatorial optimization problem. The best PMU locations are obtained by solving the combinatorial optimization problem. Once again, the simulation results show that the developed PMU placement algorithms for attack detection outperform that of the common PMU placement algorithms in terms of attack detection.

Third, low latency algorithms are studied with an objective of minimizing the attack detection delay. The attack is modeled as a sparse attack vector, with each non-zero element corresponding to one meter under attack. Since neither the support nor the values of the sparse attack vector is known, a new orthogonal matching pursuit (OMP) algorithm with a high computational efficiency is designed to identify the meters under attack. The OMP algorithm is then combined with a cumulative sum (CUSUM) test to sequentially test the statistical properties of each new measurement and declare an attack if the expected statistical properties are not met.

Fourth, the problems of state estimation and intrusion detection are further extended to a dynamic power system, where the system states and measurements vary with time owing to the dynamic nature of the system loads. A detection algorithm is designed that is capable of distinguishing intrusions from sudden changes in the system's operating point.

Moreover, the detected false data are removed from the corrupt measurements before these measurements enter the state estimator.

1.3 Dissertation Outline

The outline of the rest of the dissertation is given as follows.

Chapter 2: In this chapter, the optimum PMU placement for power system state estimation is studied. The MSE is expressed as a function of the PMU locations and the problem is solved by finding the PMU locations that minimize the MSE function. Different algorithms are developed to tackle the tradeoff between the complexity and performance. The algorithms are simulated on different IEEE bus systems to analyse their performances and compare them to other commonly used algorithms.

Chapter 3: This chapter builds on the system model of *Chapter 2* to study the optimum PMU placement for intrusion detection in power systems. The impacts of PMU placement in detecting the malicious attacks on the power grid is investigated. The probability of detecting the malicious attacks is expressed as a function of the PMU locations and the problem is solved by finding the PMU locations that maximize this probability. Furthermore, a least detectable attack is designed that is used to test the performances of the attack detection algorithms on various IEEE bus systems.

Chapter 4: In this chapter, low latency detection algorithms are designed to minimize the attack detection delay. We develop an orthogonal matching pursuit (OMP) CUSUM algorithm, which utilizes sparse recovery to identify the meters under attack and sequentially apply a statistical CUSUM test to each new group of measurements for attack detection. The simulation results show that the OMP-CUSUM algorithm can efficiently detect the attacks

with low detection delay and good detection accuracy.

Chapter 5: In this chapter, we study the state estimation and attack detection problems in dynamic systems, where sudden changes in the system loads generate transients that may be confused with attacks on the system if they are not correctly identified. We design a detection algorithm that detects and discriminate the attacks from sudden changes in the system. The designed detector alleviates the impact of attacks on the system by removing these attacks from the affected measurements before these measurements are processed by the state estimator.

Chapter 6: Conclusion remarks are drawn in this chapter. The major contributions of this dissertation are summarized and some future research topics are listed.

1.4 References

[1] J. Chen and A. Abur. Placement of pmus to enable bad data detection in state estimation. *IEEE Trans. on Power Systems*, 21(4):1608–1615, Nov 2006.

[2] A Simoes Costa, TS Piazza, and A Mandel. Qualitative methods to solve qualitative problems in power system state estimation. *IEEE Trans. on Power Systems*, 5(3):941–949, 1990.

[3] AM Leite Da Silva, MB Do Coutto Filho, and JF De Queiroz. State forecasting in electric power systems. In *IEE Proceedings C (Generation, Transmission and Distribution)*, volume 130, pages 237–244. IET, 1983.

[4] Geoff Davis, Stephane Mallat, and Marco Avellaneda. Adaptive greedy approximations. *Constructive approximation*, 13(1):57–98, 1997.

[5] Ronald A DeVore and Vladimir N Temlyakov. Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187, 1996.

[6] Milton Brown Do Coutto Filho and Julio Cesar Stacchini de Souza. Forecasting-aided state estimationpart i: Panorama. *IEEE Transactions on Power Systems*, 24(4):1667–1677, 2009.

[7] AG Expsito and Ali Abur. Generalized observability analysis and measurement classification. In *20th International Conference on Power Industry Computer Applications*, pages 97–103. IEEE, 1997.

[8] Yi Huang, Husheng Li, Kristy A Campbell, and Zhu Han. Defending false data injection attack on smart grid network using adaptive cusum test. In *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, pages 1–6. IEEE, 2011.

[9] Yi Huang, Jin Tang, Yu Cheng, Husheng Li, Kristy A Campbell, and Zhu Han. Real-time detection of false data injection in smart grid networks: an adaptive cusum method and analysis. *IEEE Systems Journal*, 10(2):532–543, 2016.

[10] Amit Jain and NR Shivakumar. Power system tracking and dynamic state estimation. In *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*, pages 1–8. IEEE, 2009.

[11] O. Kosut, L. Jia, R. J. Thomas, and L. Tong. Malicious data attacks on the smart grid. *IEEE Trans. on Smart Grid*, 2(4):645–658, Dec 2011.

[12] Hong Li and Weiguo Li. Estimation and forecasting of dynamic state estimation in power systems. In *Sustainable Power Generation and Supply, 2009. SUPERGEN'09. International Conference on*, pages 1–6. IEEE, 2009.

[13] Shang Li, Yasin Yilmaz, and Xiaodong Wang. Quickest detection of false data injection attack in wide-area smart grids. *IEEE Transactions on Smart Grid*, 6(6):2725–2735, 2015.

[14] Yagyensh Chandra Pati, Ramin Rezaiifar, and PS Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.

[15] A. G. Phadke, J. S. Thorp, R. F. Nuqui, and M. Zhou. Recent developments in state estimation with phasor measurements. In *IEEE Power and Energy Society Power Systems Conf. Expo.*, pages 1–7, March 2009.

[16] Kuang-Rong Shih and Shyh-Jier Huang. Application of a robust algorithm for dynamic state estimation of a power system. *IEEE Transactions on Power Systems*, 17(1):141–147, 2002.

[17] Joel A Tropp and Anna C Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Information Theory*, 53(12):4655–4666, 2007.

[18] B. Xu and A. Abur. Observability analysis and measurement placement for systems with pmus. In *IEEE Power and Energy Society Power Systems Conf. Expo.*, pages 943–946 vol.2, Oct 2004.

Chapter 2

Optimum PMU Placement for Power System State Estimation

2.1 Abstract

The integration of phasor measurement units (PMUs) in power grids can greatly enhance the robustness of power grid state estimations. Due to the cost of components and installations, the number of PMUs is usually much less than that of buses in a power system. Therefore one of the critical problems faced by power system design is PMU placement, that is, identifying the buses on which the PMU should be installed. The objective of this chapter is to develop PMU placement algorithms to improve the power grid state estimation. Unlike many existing PMU placement algorithms developed based on the concept of critical measurements, we use the estimation mean squared error (MSE) as the design metric. By applying a linear minimum MSE (MMSE) algorithm, the MSE is expressed as an explicit function of the locations of the PMUs. The problem is formulated as a combinatorial optimization problem that is known to be NP-hard. To balance the tradeoff between complexity and performance, we propose two low complexity algorithms, a greedy algorithm that sequentially searches for the best PMU location, and a heuristic ordered MSE algorithm that places PMUs at buses with highest MSE. Simulation results show that the proposed low complexity algorithms can almost achieve the globally optimum performance, and they significantly outperform existing PMU placement algorithms.

keywords

Power system state estimation, phasor measurement units, PMU placement, MSE

2.2 Introduction

Ever since their introduction in the power grid in the early 1980s, phasor measurement units (PMUs) have been installed in many power systems around the globe at an increasing rate [7]. A PMU device installed at a certain bus accurately measures the positive sequence voltage phasors at the bus and the current phasors of the branches incident to that bus [8]. The state estimator benefits from the PMU data by using these data along with the conventional measurements to improve the state estimation performance. However, due to the financial limitations of the utilities and the cost associated with the installation of these devices in the power grid, there are usually far less PMUs than buses. Therefore, one of the critical problems faced by power system design is PMU placement, that is, identifying the buses on which the PMUs should be installed.

Many existing approaches seek to solve the PMU placement problem by converting the system's critical measurements into redundant ones, thus to render it fully observable. The critical measurements, as opposed to the redundant measurements, are those measurements whose removal in the system results in the system being unobservable [2] and [3]. In other words, an erroneous critical measurement cannot be detected by the statistical tests based on measurement residual unless it is converted into a redundant measurement [1].

The critical measurement based PMU placement approach has been adopted in the development of numerous existing algorithms, such as [7], [8], [3], and [1]. These works thoroughly

explained how the critical measurements can be converted into redundant measurements to improve system observability and error detection. While making the system observable can increase the probability of error detection, it does not necessarily improve the accuracy of state estimation. The power system state is carried in the phase angles of all the buses, and it provides critical information regarding the health condition and security of a power system. Therefore, it is critical to develop PMU placement algorithms that can improve the accuracy of power state estimation.

In this chapter, we propose to develop PMU placement algorithms that aim at improving the accuracy of power state estimation. The design metric is the mean squared error (MSE) of the state estimation. This is quite different from the previous approaches that based on the critical measurements. In addition, the MSE provides a good metric in measuring the gain obtained by installing PMUs in the power system. With a linear minimum MSE (MMSE) algorithm, we express the state estimation MSE as an explicit function of the PMU locations. The optimum PMU placement problem is then formulated as a combinatorial optimization problem, the optimum solution of which can be achieved by means of exhaustive search. It is well known that the combinatorial optimization problem is NP-hard. To balance the tradeoff between complexity and performance, we propose two low complexity algorithms, a greedy algorithm that sequentially finds the best PMU location one at a time without considering future PMU placements, and a heuristic algorithm that place the PMUs at buses with highest estimation MSE. Although, the greedy algorithms adopted in [5] and [6] seek to improve the accuracy of the state estimation, they have different objective functions as the one in this chapter.

Simulation results show that the performance of the low complexity algorithms are very

close to that of the exhaustive search algorithm, but with a much lower complexity. In addition, the proposed algorithms achieve significant performance gains in terms of MSE over existing critical measurement based algorithms.

The remainder of this chapter is organized as follows. The system model is introduced in Section 5.3. In Section 2.4, the linear MMSE state estimator is developed, and the MSE is expressed as a function of the PMU locations. The optimum and sub-optimum PMU placement algorithms are proposed in Section 2.5. Simulation results are given in Section 5.6, and Section 5.7 concludes this chapter.

2.3 System Model

We consider a power system with $n+1$ buses. Each bus is equipped with a meter measuring the power flow and power injections. Without loss of generality, we will only consider a system model of active power flows and power injections. Define the set of buses connected to bus i as $\mathcal{X}_i$ with cardinality $c_i = |\mathcal{X}_i|$. Denote the power injection into bus i as P_i, and the power flow from bus i to bus j as P_{ij}, $\forall\, j \in \mathcal{X}_i$. The SCADA (Supervisory Control and Data Acquisition) system provides a total of $m = m_1 + m_2$ measurements, where $m_1 = n+1$ is the number of power injections and $m_2 = \frac{1}{2}\sum_{i=1}^{n+1} |\mathcal{X}_i|$ is the number of power flows. Define the power measurement vector as $\mathbf{z} = [\mathbf{z}_1^T, \mathbf{z}_2^T]^T \in \mathcal{R}^{m\times 1}$, where $(\cdot)^T$ is the matrix transpose operator, $\mathbf{z}_1 \in \mathcal{R}^{m_1\times 1}$ and $\mathbf{z}_2 \in \mathcal{R}^{m_2\times 1}$ are the power injection measurement vector and power flow measurement vector, respectively, with $\mathcal{R}$ being the set of real numbers.

The objective of state estimation is to estimate the phase angles of the buses based on the measurement vector $\mathbf{z}$. In phase estimation, one of the $n+1$ buses will serve as a reference, and we only need to estimate the phases of the remaining n buses relative to that of the

reference bus. Without loss of generality, assume that the $(n+1)$-th bus is the reference, and define the phase vector of the remaining n buses as $\mathbf{x} = [\theta_1, \theta_2, \ldots, \theta_n]^T$, where θ_i is the phase of the i-th bus.

The relationship between the observation vector $\mathbf{z}$ and the state vector $\mathbf{x}$ can be expressed as

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{e}, \tag{2.1}$$

where $\mathbf{e} = [e_1, e_2, \ldots, e_m]^T \in \mathcal{R}^{m\times 1}$ is the measurement error vector, and $\mathbf{h}(\mathbf{x}) = [h_1(\theta_1, \theta_2, \ldots, \theta_n), \ldots, h_m(\theta_1, \theta_2, \ldots, \theta_n)]^T$ is a function of bus phase angles.

In this chapter we use the standard DC power flow model [4], which results in a linear approximation of the model in (5.1) as

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{e}, \tag{2.2}$$

where $\mathbf{H} \in \mathcal{R}^{m\times n}$ is the measurement Jacobian matrix for the real power flow and power injection measurements. As in [4], we assume that the measurement noise $\mathbf{e}$ is zero-mean Gaussian with covariance matrix $\mathbf{\Sigma}_e$, that is, $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma_e^2\mathbf{I}_m)$, where $\mathbf{I}_m$ is a size-m identity matrix.

Adding PMUs to a power system provide additional measurements to the SCADA system. A PMU installed on a bus can measure both the voltage phasor of the bus and the current phasors on incident branches of the bus. The measurement results are sent directly to the state estimator of the SCADA system through communication networks. Similar to [1], we

assume that all branch impedances and bus voltages are $j1.0$ p.u. and 1.0 p.u, respectively. According to [1], for a PMU installed on the i-th bus, it can measure the voltage phase angle θ_i, and the real part of the current phasor from bus i to bus j, for all $j \in \chi_i$, as

$$I_{ij} = \theta_i - \theta_j, \tag{2.3}$$

where χ_i is the set of all the buses connected to bus i.

Based on the voltage and current measurements, define a PMU measurement matrix for the i-th bus as

$$\mathbf{H}_i = \begin{array}{c} \\ \theta_i \\ \vdots \\ I_{i,j} \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccccc} & \theta_i & & \theta_j & \end{array} \\ \begin{pmatrix} \cdots & 1 & \cdots & 0 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & 1 & \cdots & -1 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix} \end{array}, \tag{2.4}$$

The matrix $\mathbf{H}_i$ is of size $(c_i + 1) \times n$. The first row corresponds to the voltage phasor measurement, and the remaining rows correspond to the current phasor measurements on the incident branches of the i-th bus. The i-th column of $\mathbf{H}_i$ is an all-one vector. If $j \in \mathcal{X}_i$, then there will be exactly one -1 on the j-th column, with all other elements being 0.

If a PMU is placed on the i-th bus, define the corresponding PMU measurement vector as $\mathbf{v}_i = [\theta_i, I_{ij_{i,1}}, \cdots, I_{i,j_{i,c_i}}]^T \in \mathcal{R}^{(c_i+1)\times 1}$, where $j_{i,u} \in \mathcal{X}_i$ is the index of the u-th bus connected to the i-th bus, for $u = 1, \cdots, c_i$. With the above notation, the measurement from the PMU

on the i-th bus can be written as

$$\mathbf{y}_i = \mathbf{H}_i \mathbf{x} + \mathbf{e}_i \tag{2.5}$$

where $\mathbf{e}_i \in \mathcal{N}(0, \sigma_e^2 \mathbf{I}_{c_i})$ is the measurement noise.

Denote the indices of the buses with the k PMUs as $d_1 < d_2 < d_k$. Then the measurement vector provided by the PMUs is $\mathbf{z}_{\text{PMU}} = [\mathbf{y}_{d_1}^T, \cdots, \mathbf{y}_{d_k}^T]^T \in \mathcal{R}^{m_3 \times 1}$, where $m_3 = \sum_{i=1}^{k} c_{d_i} + k$ is the total number of measurements provided by the PMUs. Similarly, define $\mathbf{H}_{\text{PMU}} = [\mathbf{H}_{d_1}^T, \cdots, \mathbf{H}_{d_k}^T]^T \in \mathcal{R}^{m_3 \times n}$, and $\mathbf{e}_{\text{PMU}} = [\mathbf{e}_{d_1}^T, \cdots, \mathbf{e}_{d_k}^T]^T \in \mathcal{R}^{m_3 \times 1}$.

Then we can represent the measurement vector of the system with k PMUs as

$$\begin{bmatrix} \mathbf{z} \\ \mathbf{z}_{\text{PMU}} \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H}_{\text{PMU}} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{e} \\ \mathbf{e}_{\text{PMU}} \end{bmatrix} \tag{2.6}$$

or in a more compact form

$$\bar{\mathbf{z}} = \bar{\mathbf{H}} \mathbf{x} + \bar{\mathbf{e}}, \tag{2.7}$$

where $\bar{\mathbf{z}} = [\mathbf{z}^T, \mathbf{z}_{\text{PMU}}^T]^T \in \mathcal{R}^{\bar{m} \times 1}$, $\bar{m} = m_1 + m_2 + m_3$ is the total number of measurements, $\bar{\mathbf{e}} = [\mathbf{e}^T, \mathbf{e}_{\text{PMU}}^T]^T \sim \mathcal{N}(0, \sigma_e^2 \mathbf{I}_{\bar{m}})$, and $\bar{\mathbf{H}} = [\mathbf{H}^T, \mathbf{H}_{\text{PMU}}^T]^T \in \mathcal{R}^{\bar{m} \times n}$.

The PMUs provide additional measurements that can improve the accuracy of state estimation. Denote $\hat{\mathbf{x}}$ as the state vector estimated by using $\bar{\mathbf{z}}$. The objective is to find the optimum PMU placement vector, $\mathbf{d} = [d_1, d_2, \cdots, d_k]^T$, that can minimize the mean squared

error (MSE)

$$\sigma_0^2 = \mathbb{E}\left[\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2\right] \tag{2.8}$$

where $\|\mathbf{a}\|_2 = \sqrt{\mathbf{a}^T\mathbf{a}}$ is the l_2-norm of the vector $\mathbf{a}$.

To solve the above problem, we will first assume that the PMU location vector $\mathbf{d}$ is given, and find the minimum MSE σ_0^2 as a function of the PMU locations. The results will then be used to identify the optimum PMU locations.

2.4 MMSE State Estimation

In this section, we study the minimum mean squared error (MMSE) estimation of the state variables for a given PMU placement vector $\mathbf{d}$. For a given $\mathbf{d}$, the objective of the MMSE estimation is to find an estimate of the state variable by using the observation vector such that the MSE σ_0^2 is minimized. The results will be used to identify the optimum PMU placement.

Based on the assumption that $\mathbf{x}$ and $\bar{\mathbf{z}}$ are jointly Gaussian distributed, the MMSE estimator is a linear function of $\bar{\mathbf{z}}$ as $\hat{\mathbf{x}} = \mathbf{W}\bar{\mathbf{z}}$, where $\mathbf{W} \in \mathcal{R}^{n\times\bar{m}}$ is the MMSE matrix. Based on the orthogonality principal, $\mathbb{E}\left[(\hat{\mathbf{x}} - \mathbf{x})\mathbf{z}^T\right] = 0$, we have

$$\mathbf{W} = \mathbf{\Sigma}_x\bar{\mathbf{H}}^T\left(\bar{\mathbf{H}}\mathbf{\Sigma}_x\bar{\mathbf{H}}^T + \mathbf{\Sigma}_{\bar{e}}\right)^{-1}. \tag{2.9}$$

where $\mathbf{\Sigma}_x$ and $\mathbf{\Sigma}_{\bar{e}}$ are the covariance matrices of $\mathbf{x}$ and $\bar{\mathbf{e}}$, respectively.

Based on the orthogonality principal, we have

$$\begin{aligned}\sigma_0^2 &= \text{trace}\left\{\mathbb{E}\left[(\hat{\mathbf{x}}-\mathbf{x})(\hat{\mathbf{x}}-\mathbf{x})^T\right]\right\} \\ &= \text{trace}\left(\boldsymbol{\Sigma}_x - \mathbf{W}\bar{\mathbf{H}}\boldsymbol{\Sigma}_x\right)\end{aligned} \tag{2.10}$$

Substituting (2.9) into the above equation yields

$$\sigma_0^2 = \text{trace}\left[\boldsymbol{\Sigma}_x - \boldsymbol{\Sigma}_x\bar{\mathbf{H}}^T\left(\bar{\mathbf{H}}\boldsymbol{\Sigma}_x\bar{\mathbf{H}}^T + \boldsymbol{\Sigma}_{\bar{e}}\right)^{-1}\bar{\mathbf{H}}\boldsymbol{\Sigma}_x\right]. \tag{2.11}$$

Based on the Woodbury matrix identity $\mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{C}\left(\mathbf{B}^{-1} + \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^T\mathbf{A}^{-1} = \left(\mathbf{A} + \mathbf{CBC}^T\right)^{-1}$, the MSE in (2.11) becomes

$$\sigma_0^2 = \text{trace}\left[\boldsymbol{\Sigma}_x^{-1} + \bar{\mathbf{H}}^T\boldsymbol{\Sigma}_{\bar{e}}^{-1}\bar{\mathbf{H}}\right]^{-1}. \tag{2.12}$$

Assume there are $k \leq n$ PMUs available. Define a binary PMU indicator vector as $\mathbf{b} = [b_1, b_2, \cdots, b_n] \in \mathcal{B}^{n\times 1}$, where $\mathcal{B} = \{0, 1\}$. The indicator variable $b_i = 1$ if a PMU is placed on bus i, and $b_i = 0$ otherwise.

From the definition of $\bar{\mathbf{H}}$ and $\boldsymbol{\Sigma}_{\bar{e}}$, we have

$$\sigma_0^2 = \text{trace}\left[\boldsymbol{\Sigma}_x^{-1} + \sigma_e^{-2}\mathbf{H}^T\mathbf{H} + \sigma_e^{-2}\sum_{i=1}^{n} b_i{\mathbf{H}_i}^T\mathbf{H}_i\right]^{-1}. \tag{2.13}$$

where $b_i \in \mathcal{B} = \{0, 1\}$ is an indicator vector with $b_i = 1$ if a PMU is placed at bus i and $b_i = 0$ otherwise.

When there is no PMU, we have $b_i = 0\ \forall i$, and the MSE for system with no PMU can

thus be written as

$$\epsilon_0^2 = \text{trace}\left[\mathbf{\Sigma}_x^{-1} + \sigma^{-2}\mathbf{H}^T\mathbf{H}\right]^{-1}. \tag{2.14}$$

In (2.13), the MSE is expressed as an explicit function of the PMU location indication vector $\mathbf{b} = [b_1, \cdots, b_n]^T$. Comparing (2.13) and (2.14) reveals the impact of PMU on the estimation performance. The choice of the PMU placement vector will affect the MSE σ_0^2. The objective is to find the optimum $\mathbf{b}$ that can minimize the MSE given in (2.13).

2.5 PMU Placement Algorithms

In this section, we study the placement of PMU into a power grid to minimize the MSE of the estimated state variables. This is different from most existing PMU placement techniques, such as [1], which aim to find the best PMU placement that renders the power system observable. That is, the critical power measurements in the system are converted into redundant measurements by adding PMUs at certain bus locations [2] and [3].

From (2.13), the PMU placement problem can be formulated as

$$\begin{aligned} \text{min.} \quad & \text{trace}\left[\mathbf{\Sigma}_x^{-1} + \sigma_e^{-2}\mathbf{H}^T\mathbf{H} + \sigma_e^{-2}\sum_{i=1}^{n} b_i {\mathbf{H}_i}^T\mathbf{H}_i\right]^{-1} \\ \text{s.t.} \quad & \sum_{i=1}^{n} b_i = k \\ & b_i \in \mathcal{B}, \text{ for } i = 1, \cdots, n \end{aligned} \tag{2.15}$$

2.5.1 Optimum PMU Placement Algorithm

The optimization problem in (2.15) is in general non-convex. Given n buses and $k < n$ PMUs, it is a combinatorial problem with a complexity $\binom{n}{k}$. The optimum solution to the problem can be solved by means of exhaustive search as described in Algorithm 1.

Algorithm 1 Exhausitve Search Algorithm

1: Define the index set for all buses $\mathcal{I} = \{1, 2, \cdots, n\}$
2: Formulate the set of $\binom{n}{k}$ possible location indicator vectors $\mathcal{D} = \{\mathbf{b} | \mathbf{b} \in \mathcal{B}^n, \|\mathbf{b}\|_1 = k\}$
3: **for** $s = 1$ to $\binom{n}{k}$ **do**
4: pick $\mathbf{b}_s \in \mathcal{D}$
5: Calculate the MSE $\sigma_0^2(s)$ with (2.13)
6: **end for**
7: $\hat{s} = \text{argmin}_s \, \sigma_0^2(s)$
8: Output: $\hat{\mathbf{b}} = \mathbf{b}_{\hat{s}}$

In the exhaustive search algorithm, we try all the $\binom{n}{k}$ possible values of the location vector $\mathbf{d}$, and calculate the corresponding MSE. The one that renders the smallest MSE is the optimum PMU placement vector. Such an approach can provide the optimum performance, at the cost of a high complexity. It is well known that the combinatorial optimization problem is NP hard. In each repetition, the calculation of the MSE requires the inverse of a size $n \times n$ matrix, with a complexity on the order of $\mathcal{O}(n^3)$. Thus the complexity of the exhaustive search algorithm scales with $\mathcal{O}\left(n^3 \binom{n}{k}\right)$.

2.5.2 A Greedy Algorithm

We propose a greedy algorithm to balance the tradeoff between complexity and performance. The greedy algorithm sequentially adds the PMUs to the power system, one at a time. The PMU is added in a greedy manner, that is, each newly added PMU is placed at a location that can minimize the MSE of the current system configuration, without considering possible

future PMU placements. The greedy algorithm is described in Algorithm 2.

Algorithm 2 Greedy algorithm

Initialize the index set of all buses $\mathcal{I} = \{1, 2, \cdots, n\}$
Initialize $\mathbf{b} = \mathbf{0}_n$, a length-$n$ all-zero vector.
for $s = 1$ to k **do**
    for $u \in \mathcal{I}$ **do**
        Formulate $\mathbf{b}_u$ by flipping the u-th bit of $\mathbf{b}$.
        Calculate $\sigma_0^2(u)$ with $\mathbf{b}_u$ and (2.13)
    end for u
    $\hat{u} = \text{argmin}_u \, \sigma_0^2(u)$
    Update $b_{\hat{u}} = 1$.
    Update $\mathcal{I} = \mathcal{I} \backslash \hat{u}$
end for s
Output $\mathbf{b}$

The greedy algorithm requires k steps to find the solution, and one PMU is added at the end of each step in a greedy manner. At step s, there are $n - s + 1$ buses without PMU, and the algorithm will try to place the PMU on each one of the $n - s + 1$ buses to find the one that can minimize the MSE of the current system configuration.

In the greedy algorithm, there are a total of $\sum_{s=1}^{k}(n - s + 1) = (n + 1)k + \frac{1}{2}k(k + 1)$ repetitions. Inside each repetition, we need to calculate the MSE, which involves the inverse of a size $n \times n$ matrix, with a complexity on the order of $\mathcal{O}(n^3)$. Thus the complexity of the greedy algorithm scales with $\mathcal{O}\left(n^3(n + 1)k + \frac{1}{2}n^3k(k + 1)\right)$.

2.5.3 PMU Placement based on Ordered MSE

To further reduce the complexity, we propose a heuristic PMU placement algorithm by ordering the MSE of the estimation results.

When there is no PMU, from (2.14), the MSE covariance matrix can be written as

$$\mathbf{\Sigma}_\epsilon = \left[\mathbf{\Sigma}_x^{-1} + \sigma^{-2}\mathbf{H}^T\mathbf{H}\right]^{-1} \tag{2.16}$$

The error variance for the estimated phase on each bus are located on the diagonal of $\mathbf{\Sigma}_\epsilon$, and it can be expressed as $\boldsymbol{\sigma}_\epsilon = \text{diag}\left(\mathbf{\Sigma}_\epsilon\right)$.

A high error variance means a less accurate estimate of the state variable. Intuitively, we should place the PMUs on the buses with higher error variances to improve the estimation accuracy, such that more information can be collected regarding the phase angle on that bus.

Based on the above heuristic argument, we propose to order the elements in the MSE vector $\boldsymbol{\sigma}_\epsilon$ from high to low, and place the k PMUs at the k buses associated with the highest estimation MSE. Details of the ordered MSE algorithm is given in Algorithm 3. Such a heuristic algorithm only requires performing the inverse of an $n \times n$ matrix once, with a complexity scales with $\mathcal{O}(n^3)$.

Algorithm 3 Ordered MSE Algorithm

1: Calculate the MSE covariance matrix $\mathbf{\Sigma}_\epsilon$ with (2.16).
2: Extract the MSE vector $\boldsymbol{\sigma}_\epsilon = \text{diag}\left(\mathbf{\Sigma}_\epsilon\right)$
3: Order the elements in $\boldsymbol{\sigma}_\epsilon$ from high to low
4: Place the k PMUs at the buses with the k highest MSE

Table I compares the complexity of the three algorithms for IEEE 14-, 57-, and 118-bus system. The number of PMUs is chose as $k = \lfloor \frac{n}{2} \rfloor$. The complexities are measured by using the scaling factors with respect to n and k. The exhaustive search algorithm has the highest complexity, followed by the greedy algorithm, and the one based on covariance matrix has

the lowest complexity.

Table 2.1: Comparison of Complexities of Different Algorithms

Algorithms	Exhaustive	Greedy	Ordered MSE
14-bus	9.40×10^6	3.65×10^5	2.74×10^3
57-bus	2.78×10^{21}	3.76×10^8	1.85×10^5
118-bus	4.0×10^{40}	1.44×10^{10}	1.64×10^6

2.6 Simulation Results

In this section, we present the simulation results by using several standard IEEE bus configurations. The simulations are performed by using the MATPOWER software. In the simulation, it is assumed that the state variables $\mathbf{x}$ are Gaussian distributed with zero mean and covariance matrix $\mathbf{\Sigma_x} = \sigma_x^2 \mathbf{I}_n$. The covariances of the measurement noise are $\mathbf{\Sigma}_{\bar{e}} = \sigma_e^2 \mathbf{I}_{\bar{m}}$ and $\mathbf{\Sigma}_e = \sigma_e^2 \mathbf{I}_m$ for the systems with and without PMU, respectively. The signal-to-noise ratio (SNR) in dB is defined as $10 \log \frac{\sigma_x^2}{\sigma_e^2}$.

Fig. 2.1 shows the MSE as a function of the number of PMUs for the IEEE 57-bus system. This system has 57 buses and 80 branches. Hence, the number of state variables is $n = 57 - 1 = 56$ because bus 1 is used as the reference bus. We consider a reduced IEEE bus 57 system in [1] to simplify the comparison between our algorithm and the one in [1]. As a result, the number of measurements is $m = 33 + 32 = 65$, of which $m_1 = 32$ are power injection measurements and $m_2 = 33$ are the real power flow measurements. The SNR is 10 dB. The performance of system without PMU is also shown in the figure for reference. The simulated MSE curves are obtained through Monte-Carlo simulations by generating zero-mean Gaussian distributed state variables, and the theoretical MSE curves are obtained from (2.13) and (2.14) for the systems with and without PMUs, respectively. The performance

of the proposed algorithms are compared to the existing critical measurement-based PMU placement algorithm [1]. For systems with PMU, the performance improves as more PMUs are deployed in the system. Among all algorithms considered in this example, the greedy algorithm has the best performance. The algorithm based on ordered MSE is slightly worse than the greedy algorithm when the number of PMUs k is between 10 and 20, and it is almost identical to the greedy algorithm when $k \geq 20$. The performance of the algorithm based on critical measurement [1] is far worse than the proposed algorithms. All algorithms achieve the best performance when there are $k = n = 56$ PMUs. The majority of the performance improvement of the proposed algorithms is achieved when k is small (e.g. $k \leq 20$), and the performance improvement gradually diminishes as k becomes large. For the proposed algorithms, the MSE achieved at $k = 30$ is almost identical to that of system with $k = n = 56$ PMUs. Thus the optimum performance can be achieved by using only $k = 30$ PMUs. When $k = 30$, the proposed algorithms outperform the system with the critical measurement algorithm and the system without PMU by 31.3% and 75%,respectively.

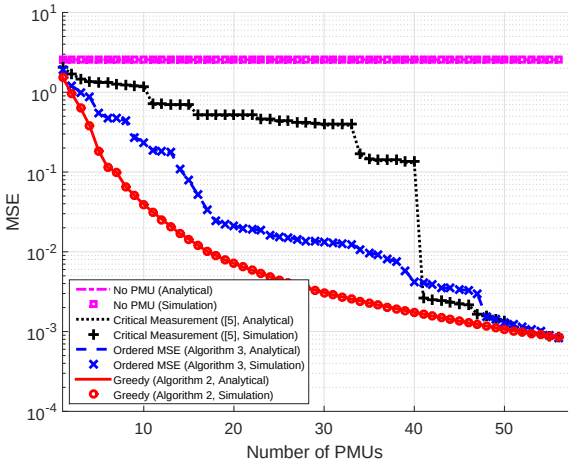

Figure 2.1: The MSE as a function of the number of PMUs for IEEE 57-bus system.

To demonstrate the performance of the exhaustive algorithm, Fig. 2.2 shows the performance of various algorithms for the IEEE 14-bus system. This system has 14 buses and 20 branches. Hence, the number of state variables is $n = 14 - 1 = 13$. The number of measurements is $m = 20 + 14 = 34$, with $m_1 = 14$ real power injection measurements and $m_2 = 20$ real power flow measurements. The SNR is 10 dB. The performance of the greedy algorithm is almost the same as that of the exhaustive search algorithm, but with a much lower complexity. This means the greedy algorithm can almost achieve the optimum performance. The performance of the ordered MSE algorithm is slightly worse than that of the greedy algorithm. All proposed algorithms significantly outperform the conventional critical measurement-based algorithm.

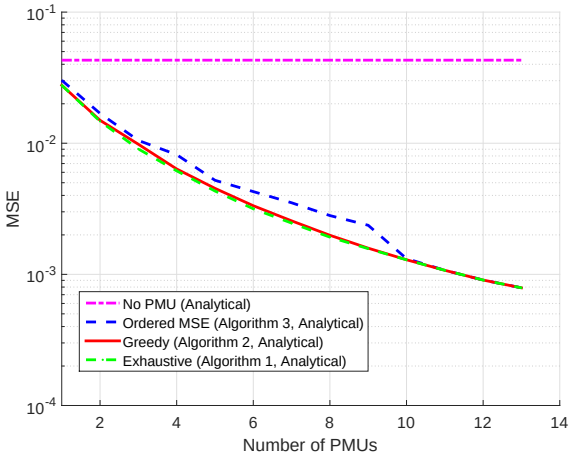

Figure 2.2: The MSE as a function of the number of PMUs for IEEE 14-bus system.

2.7 Conclusion

We have studied the PMU placement problem to improve the accuracy of phase estimation in a power system. The design metric was the MSE of the estimated phase angles. With a linear MMSE estimator, the estimation MSE has been expressed as an explicit function of the

PMU placement vector. The optimum PMU placement can be achieved with an exhaustive search algorithm with combinatorial complexity. Two low complexity algorithms have been proposed to balance the tradeoff between performance and complexity. Simulation results have shown that the performance of the low complexity algorithms approach that of the exhaustive search algorithm, but with a much lower complexity. All proposed algorithms achieve significant performance gains over conventional algorithms designed based on the concept of critical measurement. For a 57-bus system, the proposed algorithms can achieve the optimum performance with only 30 PMUs while conventional algorithms require 56 PMUs.

2.8 Appendix of the Copyright

2.8.1 Copyright Clearance


Copyright Clearance Center RightsLink®

Home | Create Account | Help

IEEE Requesting permission to reuse content from an IEEE publication

Title: Optimum PMU placement for power system state estimation
Conference Proceedings: Power & Energy Society General Meeting, 2017 IEEE
Author: Israel Akingeneye
Publisher: IEEE
Date: July 2017
Copyright © 2017, IEEE

LOGIN
If you're a copyright.com user, you can login to RightsLink using your copyright.com credentials.
Already a RightsLink user or want to learn more?

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK | CLOSE WINDOW

2.9 References

[1] J. Chen and A. Abur. Placement of pmus to enable bad data detection in state estimation. *IEEE Trans. on Power Systems*, 21(4):1608–1615, Nov 2006.

[2] A Simoes Costa, TS Piazza, and A Mandel. Qualitative methods to solve qualitative problems in power system state estimation. *IEEE Trans. on Power Systems*, 5(3):941–949, 1990.

[3] AG Expsito and Ali Abur. Generalized observability analysis and measurement classification. In *20th International Conference on Power Industry Computer Applications*, pages 97–103. IEEE, 1997.

[4] O. Kosut, L. Jia, R. J. Thomas, and L. Tong. Malicious data attacks on the smart grid. *IEEE Trans. on Smart Grid*, 2(4):645–658, Dec 2011.

[5] Q Li, T Cui, Y Weng, R Negi, F Franchetti, and M. D. Ilic. An information-theoretic approach to pmu placement in electric power systems. *IEEE Trans. on Smart Grid*, 4(1):446–456, March 2013.

[6] Q Li, Ri Negi, and M. D. Ilic. Phasor measurement units placement for power system state estimation: A greedy approach. In *IEEE Power and Energy Society General Meeting*, pages 1–8, July 2011.

[7] A. G. Phadke, J. S. Thorp, R. F. Nuqui, and M. Zhou. Recent developments in state estimation with phasor measurements. In *IEEE Power and Energy Society Power Systems Conf. Expo.*, pages 1–7, March 2009.

[8] B. Xu and A. Abur. Observability analysis and measurement placement for systems with pmus. In *IEEE Power and Energy Society Power Systems Conf. Expo.*, pages 943–946 vol.2, Oct 2004.

Chapter 3

Optimum PMU Placement for Bad Data Detection in Power Systems

3.1 Abstract

We study the detection of bad data maliciously injected in a power grid by strategically placing phasor measurement units (PMUs) at various buses across the power system. The employment of PMU in power grids can greatly improve the capability of bad data detection. However, a power grid is normally so large that it can be costly to install PMUs at every bus in the grid. Thus it is critical to identify the buses on which the PMU should be installed. We propose to optimize PMU placements by maximizing the probability of detecting the attack of bad data injection, subject to a constraint on the probability of false alarm. We first develop an optimum bad data detector by following the Neyman Pearson criterion. The corresponding detection probability is derived as a closed-form expression of the Kullback Leibler (KL) divergence between the measurement data with and without attack. It is shown that the detection probability is monotonically increasing in the KL divergence, thus we propose to design the PMU placement algorithms by maximizing the KL divergence. Since the data used in an attack is unknown, the PMU algorithms are developed by following a max-min criterion, that is, maximizing the minimum KL divergence (or detection probability) under the assumption of the least detectable attack vector. Under the max-min criterion, we present an optimum PMU placement algorithm based on exhaustive search and a low complexity greedy algorithm based on sequential search. Simulation results show that using

the KL divergence as a design metric results in significant performance gains over existing methods that are developed based on critical measurements.

Keywords

phasor measurement units, PMU placement, Anomaly detection, Kullback-Leibler divergence, probability of detection.

3.2 Introduction

Ever since their introduction in the power grid in the early 1980s, phasor measurement units (PMUs) or synchrophasors have been installed in many power systems around the globe at an increasing rate [12]. A PMU device installed at a certain bus accurately measures the positive sequence voltage phasors at the bus and the current phasors of the branches incident to that bus [16]. The state estimator benefits from the PMU data by using these data along with the conventional measurements to improve the performance of state estimations and error detections. However, due to the financial limitations of the utilities and the cost associated with the installation of these devices in the power grid, there are usually far less PMUs than buses. Therefore, one of the critical problems faced by power system design is PMU placement, that is, identifying the buses on which the PMUs should be installed.

Many existing approaches seek to solve the PMU placement problem by converting the system's critical measurements into redundant ones, thus to render the power system fully observable. The critical measurements, as opposed to the redundant measurements, are those measurements whose removal results in the system being unobservable [2] and [4]. In other words, an erroneous critical measurement cannot be detected by the statistical tests

based on measurement residual unless it is converted into a redundant measurement [1].

The problem of bad data injection was first studied in [11]. Contrary to the bad measurements due to faults, equipment failures, and other random causes, the bad data injection attacks are arbitrary measurement errors introduced into the state estimation of an electric power grid by an adversary. [11] demonstrated that an attacker can take advantage of the configuration of the power system by compromising certain meters and therefore misleading the state estimator. [14] discussed various ways that an intruder may use to carry out these attacks: First,the intruder may corrupt the power flow measurements by remotely accessing the internet protocol (IP)-based automation devices such as the remote terminal units (RTU) installed at the substations that allow control engineers to perform system diagnostic functionality from a remote area. Second, the intruder may tamper with the communication network or break into the supervisory control and data acquisition (SCADA) system through the control center office Local Area Network (LAN).

The countermeasures have been studied in [10], [3], and [17] based on the residue test that uses the difference between the observed and estimated measurements to detect and identify the bad measurements. However, [11] shows that an attack designed with knowledge of the network topology can bypass these detection algorithms. The introduction of PMU has been discussed in the literature to improve the system robustness and enhance the measurement residual tests by converting the critical measurements into redundant measurements. This approach, however, suffers, in turn, of its reliance on measurement residual testing.

The critical measurement based PMU placement approach has been adopted in the development of numerous existing algorithms, such as [12], [16], [4], and [1]. These works

thoroughly explained how the critical measurements can be converted into redundant measurements to improve system observability and thus error detection. While making the system observable can increase the probability of detecting some specific attacks, such as those that only target the critical measurements, this approach may not be sufficient to detect any random attacks. Furthermore, most of these approaches deploy the residual based statistical tests to measure the performance of the designed PMU placement algorithms, and these tests possess well documented limitations in detecting certain attacks. It is very difficult, if not impossible, to predict how the adversary will design the attack. Therefore, it is critical to develop PMU placement algorithms that can enhance the power state estimator's capabilities of detecting not merely specific attacks but any random attacks.

In this chapter, we propose to develop PMU placement algorithms that can maximize the probability of detecting bad data maliciously injected in the power system, subject to a constraint on the probability of false alarm. The bad data detection is performed by using the measurement data collected by the SCADA (Supervisory Control and Data Acquisition) system and the PMUs. We first develop an optimum detector by using the Neyman Pearson criterion. The probability of detection of the optimum detector are expressed as closed-form expressions of the Kullback-Leibler (KL) divergence between the measurements with and without malicious attacks. It is shown that the probability of detection is monotonically increasing in the KL divergence, thus maximizing the probability of detection is equivalent to maximizing the KL divergence. The KL divergence measures the difference between the distributions of the normal and corrupted data, and it has been used to assist line outage detection [15]. The KL divergence depends on the data covariance matrices, which are in turn determined by the locations of the PMU in the power system. Therefore we can identify

the optimum PMU locations by maximizing the KL divergence.

By using the KL divergence as the design metric, we identify the least detectable attack vector, that is, the attack vector that will result in the smallest KL divergence with the normal data. The PMU placement problem is then formulated based on the max-min criterion, that is, maximizing the minimum KL divergence or detection probability due to the least detectable attack vector. The problem is a combinatorial optimization problem, the optimum solution of which can be achieved by means of exhaustive search. To balance the tradeoff between complexity and performance, we also propose a low complexity algorithm that sequentially finds the best PMU location that can maximize the current KL divergence without considering future PMU placements. It is worth pointing out greedy PMU placement algorithms are adopted in [8] and [9] to improve the accuracy of state estimation, and they have different objectives as the greedy algorithm in this chapter. Simulation results show that using KL divergence as a design metric for bad data detection results in significant performance gains over conventional methods based on critical measurements.

The remainder of this chapter is organized as follows. The system model is described in Section 3.3. In Section 3.4, the Neyman Pearson detector is developed, and the corresponding probability of detection is expressed as a function of the KL divergence between corrupted and normal measurements. In Section 3.5, we first identify the least detectable attack vector, which is used to formulate the max-min optimization problem. The optimum and greedy PMU placement algorithms are then present. Simulation results are given in Section 3.6, and Section 3.7 concludes this chapter.

3.3 System Model

We consider a power system with $n+1$ buses. Each bus is equipped with a meter measuring the power flow and power injections. Without loss of generality, we will only consider a system model of active power flows and power injections. Define the set of buses connected to bus i as $\mathcal{X}_i$ with cardinality $c_i = |\mathcal{X}_i|$. Denote the power injection into bus i as P_i, and the power flow from bus i to bus j as P_{ij}, $\forall\, j \in \mathcal{X}_i$. The SCADA (Supervisory Control and Data Acquisition) system provides a total of $m = m_1 + m_2$ measurements, where $m_1 = n + 1$ is the number of power injections and $m_2 = \frac{1}{2}\sum_{i=1}^{n+1} |\mathcal{X}_i|$ is the number of power flows. Define the power measurement vector as $\mathbf{z} = [\mathbf{z}_1^T, \mathbf{z}_2^T]^T \in \mathcal{R}^{m\times 1}$, where $(\cdot)^T$ is the matrix transpose operator, $\mathbf{z}_1 \in \mathcal{R}^{m_1\times 1}$ and $\mathbf{z}_2 \in \mathcal{R}^{m_2\times 1}$ are the power injection measurement vector and power flow measurement vector, respectively, with $\mathcal{R}$ being the set of real numbers.

In phase measurement, one of the $n+1$ buses will serve as a reference, and we only need to measure or estimate the phases of the remaining n buses relative to that of the reference bus. Without loss of generality, assume that the $(n+1)$-th bus is the reference, and define the phase vector of the remaining n buses as $\mathbf{x} = [\theta_1, \theta_2, \ldots, \theta_n]^T$, where θ_i is the phase of the i-th bus.

The relationship between the observation vector $\mathbf{z}$ and the state vector $\mathbf{x}$ can be expressed as

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{e}, \tag{3.1}$$

where $\mathbf{e} = [e_1, e_2, \ldots, e_m]^T \in \mathcal{R}^{m\times 1}$ is the measurement error vector, and the function $\mathbf{h}(\mathbf{x}) =$

$[h_1(\theta_1, \theta_2, \ldots, \theta_n), \ldots, h_m(\theta_1, \theta_2, \ldots, \theta_n)]^T$ is a function of bus phase angles.

In this chapter we use the standard DC power flow model [13], which results in a linear approximation of the model in (3.1) as

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{e}, \tag{3.2}$$

where $\mathbf{H} \in \mathcal{R}^{m \times n}$ is the measurement Jacobian matrix for the real power flow and power injection measurements. As in [7], we assume that the measurement noise $\mathbf{e}$ is zero-mean Gaussian with covariance matrix $\mathbf{\Sigma}_e$, that is, $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma_e^2 \mathbf{I}_m)$, where $\mathbf{I}_m$ is a size-m identity matrix.

Adding PMUs to a power system provides additional measurements to the SCADA system. A PMU installed on a bus can measure both the voltage phasor of the bus and the current phasors on incident branches of the bus. The measurement results are sent directly to the state estimator of the SCADA system through communication networks. Similar to [1], we assume that all branch impedances and bus voltages are $j1.0$ p.u. and 1.0 p.u, respectively. For a PMU installed on the i-th bus, it can measure the voltage phase angle θ_i, and the real part of the current phasor I_{ij} from bus i to bus j, for all $j \in \chi_i$. According to [1], the current phasor can be approximated by

$$I_{ij} = \theta_i - \theta_j, \tag{3.3}$$

Based on the voltage and current measurements, define a PMU measurement matrix for

the i-th bus as

$$\mathbf{H}_i = \begin{array}{c} \\ \theta_i \\ \vdots \\ I_{i,j} \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccccc} & \theta_i & & \theta_j & \end{array} \\ \begin{pmatrix} \cdots & 1 & \cdots & 0 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & 1 & \cdots & -1 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix} \end{array}, \tag{3.4}$$

The matrix $\mathbf{H}_i$ is of size $(c_i + 1) \times n$. The first row corresponds to the voltage phasor measurement, and the remaining rows correspond to the current phasor measurements on the incident branches of the i-th bus. The i-th column of $\mathbf{H}_i$ is an all-one vector. If $j \in \mathcal{X}_i$, then there will be exactly one -1 on the j-th column, with all other elements being 0.

If a PMU is placed on the i-th bus, define the corresponding PMU measurement vector as $\mathbf{v}_i = [\theta_i, I_{ij_{i,1}}, \cdots, I_{i,j_{i,c_i}}]^T \in \mathcal{R}^{(c_i+1)\times 1}$, where $j_{i,u} \in \mathcal{X}_i$ is the index of the u-th bus connected to the i-th bus, for $u = 1, \cdots, c_i$. With the above notation, the measurement from the PMU on the i-th bus can be written as

$$\mathbf{y}_i = \mathbf{H}_i\mathbf{x} + \mathbf{e}_i \tag{3.5}$$

where $\mathbf{e}_i \in \mathcal{N}(0, \sigma_e^2\mathbf{I}_{c_i})$ is the measurement noise.

Assume k PMUs are used in the power network. Denote the indices of the buses with the k PMUs as $d_1 < d_2 < d_k$. Then the measurement vector provided by the PMUs is $\mathbf{z}_{\text{PMU}} = [\mathbf{y}_{d_1}^T, \cdots, \mathbf{y}_{d_k}^T]^T \in \mathcal{R}^{m_3\times 1}$, where $m_3 = \sum_{i=1}^{k} c_{d_i} + k$ is the total number of measurements

provided by the PMUs. Similarly, define $\mathbf{H}_{\text{PMU}} = [\mathbf{H}_{d_1}^T, \cdots, \mathbf{H}_{d_k}^T]^T \in \mathcal{R}^{m_3 \times n}$, and $\mathbf{e}_{\text{PMU}} = [\mathbf{e}_{d_1}^T, \cdots, \mathbf{e}_{d_k}^T]^T \in \mathcal{R}^{m_3 \times 1}$.

Then we can represent the measurement vector of the system with k PMUs as

$$\begin{bmatrix} \mathbf{z} \\ \mathbf{z}_{\text{PMU}} \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H}_{\text{PMU}} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{e} \\ \mathbf{e}_{\text{PMU}} \end{bmatrix} \tag{3.6}$$

or in a more compact form

$$\bar{\mathbf{z}} = \bar{\mathbf{H}}\mathbf{x} + \bar{\mathbf{e}}, \tag{3.7}$$

where $\bar{\mathbf{z}} = [\mathbf{z}^T, \mathbf{z}_{\text{PMU}}^T]^T \in \mathcal{R}^{\bar{m} \times 1}$, $\bar{m} = m_1 + m_2 + m_3$ is the total number of measurements, $\bar{\mathbf{e}} = [\mathbf{e}^T, \mathbf{e}_{\text{PMU}}^T]^T \sim \mathcal{N}(0, \sigma_e^2 \mathbf{I}_{\bar{m}})$, and $\bar{\mathbf{H}} = [\mathbf{H}^T, \mathbf{H}_{\text{PMU}}^T]^T \in \mathcal{R}^{\bar{m} \times n}$.

The PMUs provide additional measurements that can improve the accuracy of state estimation and bad data detection. The objective is to find the optimum PMU placement vector, $\mathbf{d} = [d_1, d_2, \cdots, d_k]^T$, that can maximize the probability of detecting malicious data injection in the power grid.

To do so, we will first assume that the PMU location vector $\mathbf{d}$ is given, and develop optimum detection algorithms that can maximize the probability of detection. The maximum probability of detection will be expressed as a function of the PMU locations. The results will then be used to identify the optimum PMU locations.

3.4 Bad Data Detection

The optimum detection of bad data in the SCADA measurements is studied in this section for systems with a given PMU location vector $\mathbf{d}$. In a power grid, malicious data can be injected into the SCADA measurements by an attacker so as to mislead the SCADA into making wrong state estimations. An attacker seeks to find an attack vector $\mathbf{a} = [a_1, a_2, \ldots, a_m]^T$ that can cause a high mean squared error (MSE) at the estimator while keeping the probability of detection as low as possible.

It is assumed that the attack vector will only affect the traditional SCADA measurements vector $\mathbf{z}$, which includes the power flows and power injections at different buses, The attacker has no access to the PMU measurements, because the locations of the PMUs are usually unknown to the attacker and the PMU data are usually encrypted [6].

With the measurement model given in (3.2) after the malicious data injection by an adversary, the corrupted measurement SCADA can be written as

$$\mathbf{z}_a = \mathbf{H}\mathbf{x} + \mathbf{a} + \mathbf{e}, \tag{3.8}$$

where $\mathbf{a} \in \mathcal{R}^{m\times 1}$ is the attack vector.

For a system with PMU, from (3.7), the corrupted measurement can be written as

$$\bar{\mathbf{z}}_a = \bar{\mathbf{H}}\mathbf{x} + \bar{\mathbf{a}} + \bar{\mathbf{e}}, \tag{3.9}$$

where $\bar{\mathbf{z}}_a = \left[\mathbf{z}_a^T, \mathbf{z}_{\text{PMU}}^T\right]^T \in \mathcal{R}^{\bar{m}\times 1}$, and $\bar{\mathbf{a}} = \left[\mathbf{a}^T, \mathbf{0}_{m_3}^T\right]^T \in \mathcal{R}^{\bar{m}\times 1}$ with $\mathbf{0}_{m_3}$ being a length-m_3 all-zero vector.

3.4.1 Conventional Residual-based Bad Data Detectors

Many conventional bad data detectors rely on the residual errors of state estimation to detect the presence of malicious data. Denote the residual error of state estimation as $\mathbf{r} = \mathbf{z} - \mathbf{H}\hat{\mathbf{x}}$, where $\hat{\mathbf{x}}$ as the minimum mean squared error (MMSE) estimate of $\mathbf{x}$ by using the SCADA measurement $\mathbf{z}$. The properties of $\mathbf{r}$ can be used to detect the presence of bad data.

An example of such detectors is the largest normalized residue (LNR) test [7], which can be written as

$$\max_i \frac{|r_i|}{\sigma_{r_i}} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau, \tag{3.10}$$

where τ is a pre-defined threshold, r_i is the i-th entry of the residual error vector $\mathbf{r}$ with σ_{r_i} being its standard deviation.

One of the limitations of the detector in (3.10) is, according to [11], that it cannot detect an attack vector $\mathbf{a} = \mathbf{Hc}$, where $\mathbf{c} = [c_1, c_2, \ldots, c_n]^T$ is an arbitrary vector chosen by the adversary. To design such an attack vector, an adversary needs knowledge of the Jacobian matrix, $\mathbf{H}$.

3.4.2 Optimum Detector

In this subsection, we develop the optimum detector that can maximize the probability of detection for a given probability of false alarm (PFA).

For the bad data detection, the null hypothesis $\mathcal{H}_0$ and the alternative hypothesis $\mathcal{H}_1$,

respectively, are

$$\mathcal{H}_0 : \mathbf{a} = \mathbf{0} \quad \text{and} \quad \mathcal{H}_1 : \mathbf{a} \neq \mathbf{0}, \tag{3.11}$$

The attack vector can be any arbitrary vector and is unknown to the detector.

During the detection, we adopt the common assumption that the state vector $\mathbf{x}$ is zero mean Gaussian distributed with covariance matrix $\mathbf{\Sigma}_x$, that is, $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_x)$. Under the null hypothesis, from (3.9), the measurement vector $\bar{\mathbf{z}}_a$ is zero-mean Gaussian distributed with covariance matrix $\bar{\mathbf{\Sigma}}_z = \bar{\mathbf{H}}\mathbf{\Sigma}_x\bar{\mathbf{H}}^T + \bar{\mathbf{\Sigma}}_e$, where $\bar{\mathbf{\Sigma}}_e = \sigma_e^2\mathbf{I}_{\bar{m}}$ is the covariance matrix of the noise vector $\bar{\mathbf{e}}$. On the other hand, under the alternative hypothesis, the measurement vector $\bar{\mathbf{z}}_a$ is Gaussian distributed with mean vector $\bar{\mathbf{a}}$ and covariance matrix $\bar{\mathbf{\Sigma}}_z$. That is

$$\begin{aligned} &\mathcal{H}_0 : \bar{\mathbf{z}}_a \sim \mathcal{N}(\mathbf{0}, \bar{\mathbf{\Sigma}}_{\mathbf{z}}) \\ &\mathcal{H}_1 : \bar{\mathbf{z}}_a \sim \mathcal{N}(\bar{\mathbf{a}}, \bar{\mathbf{\Sigma}}_{\mathbf{z}}), \quad \bar{\mathbf{a}} \neq \mathbf{0}, \end{aligned} \tag{3.12}$$

Given the hypotheses in (3.12), we wish to design a detector that can maximize the probability of detection subject to a constraint on the upper bound of the probability of false alarm P_{FA}. Based on the Neyman-Pearson Lemma, the optimum detector is the likelihood ratio test (LRT) as

$$L(\bar{\mathbf{z}}_a) \equiv \log \frac{f(\bar{\mathbf{z}}_a \mid \mathcal{H}_1)}{f(\bar{\mathbf{z}}_a \mid \mathcal{H}_0)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau'. \tag{3.13}$$

where $L(\bar{\mathbf{z}}_a)$ is the log likelihood ratio (LLR), $f(\bar{\mathbf{z}}_a \mid \mathcal{H}_b)$ is the conditional probability density

function (pdf) of the measurement vector $\bar{\mathbf{z}}_a$ under hypothesis $\mathcal{H}_b$, for $b = 0, 1$, and τ' is a threshold determined by the probability of false alarm P_{FA}.

After some algebraic operations and simplification, the detector in (3.13) is equivalent to

$$y \equiv \bar{\mathbf{z}}_a^T \bar{\mathbf{\Sigma}}_z^{-1} \bar{\mathbf{a}} - \frac{1}{2} \bar{\mathbf{a}}^T \bar{\mathbf{\Sigma}}_{\mathbf{z}}^{-\mathbf{1}} \bar{\mathbf{a}} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau, \tag{3.14}$$

where τ is a predefined threshold determined by P_{FA} and y is the test statistic that depends on the measurement results $\bar{\mathbf{a}}$. It should be noted that the knowledge of $\bar{\mathbf{a}}$ is required in the test described in (3.14). However, in practical systems, $\bar{\mathbf{a}}$ is unknown to the detector. The optimum detector is used to provide a guideline for the PMU placement. The PMU placement algorithm developed in the next section will not require the knowledge of $\bar{\mathbf{a}}$.

Since $\bar{\mathbf{z}}_a|\mathcal{H}_b$ is conditionally Gaussian distributed and the test statistic y is a linear function of $\bar{\mathbf{z}}_a$, it is straightforward that $y|\mathcal{H}_b$ is conditionally Gaussian distributed, for $b = 0, 1$. Based on (3.14), we have

$$\begin{aligned} \mathcal{H}_0 &: y \sim \mathcal{N}(\mu_{0y}, \sigma_y^2) \\ \mathcal{H}_1 &: y \sim \mathcal{N}(\mu_{1y}, \sigma_y^2), \end{aligned} \tag{3.15}$$

where $\mu_{0y} = -\frac{1}{2} \bar{\mathbf{a}}^T \bar{\mathbf{\Sigma}}_{\mathbf{z}}^{-\mathbf{1}} \bar{\mathbf{a}}$ and $\mu_{1y} = \frac{1}{2}\sigma_y^2 = \frac{1}{2} \bar{\mathbf{a}}^T \bar{\mathbf{\Sigma}}_{\mathbf{z}}^{-\mathbf{1}} \bar{\mathbf{a}}$.

Denote the distributions of y under the null and alternative hypothesis as $p_0(y)$ and $p_1(y)$, respectively. The performance of the threshold test depends on the KL divergence between $p_1(y)$ and $p_0(y)$. Based on [5], the KL divergence between $p_1(y)$ and $p_0(y)$ can be calculated

as

$$D(p_1\|p_0) = \frac{1}{2}\left[\mathrm{tr}\left(\sigma_y^2\sigma_y^{-2}\right) + \mu\sigma_y^{-2}\mu - 1 + \ln\left(\frac{\sigma_y^2}{\sigma_y^2}\right)\right]$$
$$= \frac{1}{2}\bar{\mathbf{a}}^T\bar{\mathbf{\Sigma}}_{\mathbf{z}}^{-\mathbf{1}}\bar{\mathbf{a}}, \tag{3.16}$$

where $\mu = \mu_{0_y} - \mu_{1_y}$. It is interesting to note that $\mu_{1_y} = -\mu_{0_y} = D(p_1\|p_0)$.

With the threshold test given in (3.14), we have the following lemma that establishes the relationship between P_{FA} and the threshold τ.

Lemma 3.1: With the threshold test defined in (3.14), the probability of false alarm is

$$P_{\mathrm{FA}} = Q\left(\frac{\tau - \mu_{0_y}}{\sqrt{2D(p_1\|p_0)}}\right), \tag{3.17}$$

where $Q(t) = \frac{1}{\sqrt{2\pi}}\int_t^\infty \exp\left(-\frac{u^2}{2}\right)du$ is the Gaussian-Q function, and $D(p_1\|p_0)$ is the KL divergence given in (3.16).

Proof: False alarm happens when a bad data is detected under the null hypothesis. Based on the threshold test, the probability of false alarm can be calculated as

$$P_{\mathrm{FA}} = \Pr\left(y > \tau \mid \mathcal{H}_0\right)$$
$$= \int_\tau^\infty \frac{1}{\sqrt{2\pi\sigma_y^2}}\exp\left(-\frac{1}{2\sigma_y^2}(y-\mu_{0_y})^2\right)dy$$
$$= Q\left(\frac{\tau - \mu_{0_y}}{\sigma_y}\right) \tag{3.18}$$

The result in (3.17) can then be obtained by substituting μ_{0_y} and $\sigma_y^2 = 2D(p_1\|p_0)$ into the above equation. ■

With the results in Lemma 3.1, we can select the threshold as

$$\tau = Q^{-1}(P_{\text{FA}})\sqrt{2D(p_1\|p_0)} + \mu_{0_y} \tag{3.19}$$

where $Q^{-1}(x)$ is the inverse function of $Q(x)$.

With the threshold given in (3.19), the maximum probability of detection is given in the following theorem.

Theorem 3.1: Consider the threshold detector given in (3.14). For a given probability of false alarm, P_{FA}, the probability of detection is

$$P_{\text{D}} = Q\left(Q^{-1}\left(P_{\text{FA}}\right) - \sqrt{2D(p_1 \,||\, p_0)}\right), \tag{3.20}$$

where $D(p_1 \,||\, p_0) = \frac{1}{2}\mathbf{a}^T\boldsymbol{\Sigma}_z^{-1}\mathbf{a}$ is the KL divergence given in (3.16).

Proof: From (3.14) and (3.15), the probability of detection can be calculated by

$$\begin{aligned} P_{\text{D}} &= \Pr\left(y > \tau \mid \mathcal{H}_1\right) \\ &= \int_{\tau}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_y} \exp\left[-\frac{(y-\mu_{1_y})^2}{2\sigma_y^2}\right] dy, \\ &= Q\left(\frac{\tau - \mu_{1_y}}{\sqrt{\sigma_y^2}}\right) \end{aligned} \tag{3.21}$$

where $\sigma_y^2 = 2\mu_{1_y} = 2D(p_1\|p_0)$. Substituting (3.19) into the above equation yields (3.20). ■

From the results in Theorem 3.1, it is easy to show that for a given P_{FA}, the probability of detection is an increasing function in the KL divergence $D(p_1\|p_0)$, because $Q(x)$ is a decreasing function in x. Therefore, maximizing the probability of detection is equivalent to

maximizing the KL divergence. On the other hand, the objective of a malicious attacker is to design an attack vector such that $D(p_1\|p_0)$ is minimized to minimize the probability of detection.

3.5 PMU Placement Algorithms

In this section, we study the placement of PMUs into a power grid that can maximize the probability of detecting bad data injections. The probability of detection is maximized by maximizing the KL divergence between the probability distributions of the measurements with and without attack. This is different from most existing PMU placement techniques that are designed based on critical measurements or observability of the power system [1], [2] and [4].

The KL divergence in (3.16) is a function of the attack vector $\bar{\mathbf{a}}$ and the inverse of the measurement covariance matrix $\bar{\mathbf{\Sigma}}_z$. The covariance matrix $\bar{\mathbf{\Sigma}}_z = \bar{\mathbf{H}}\mathbf{\Sigma}_x\bar{\mathbf{H}}^T + \bar{\mathbf{\Sigma}}_e$ can be alternatively expressed as

$$\bar{\mathbf{\Sigma}}_z = \begin{bmatrix} \mathbf{\Sigma}_{zz} & \mathbf{\Sigma}_{zp} \\ \mathbf{\Sigma}_{pz} & \mathbf{\Sigma}_{pp} \end{bmatrix}, \tag{3.22}$$

where

$$\mathbf{\Sigma}_{zz} = \mathbb{E}[\mathbf{z}\mathbf{z}^T] = \mathbf{H}\mathbf{\Sigma}_x\mathbf{H}^T + \sigma_e^2\mathbf{I}_m \tag{3.23}$$

$$\mathbf{\Sigma}_{pp} = \mathbb{E}[\mathbf{z}_{\text{PMU}}\mathbf{z}_{\text{PMU}}^T] = \mathbf{H}_{\text{PMU}}\mathbf{\Sigma}_x\mathbf{H}_{\text{PMU}}^T + \sigma_e^2\mathbf{I}_{m_3} \tag{3.24}$$

$$\mathbf{\Sigma}_{zp} = \mathbb{E}[\mathbf{z}\mathbf{z}_{\text{PMU}}^T] = \mathbf{H}\mathbf{\Sigma}_x\mathbf{H}_{\text{PMU}}^T \tag{3.25}$$

and $\mathbf{\Sigma}_{zp} = \mathbf{\Sigma}_{pz}^T$.

The KL divergence $D(p_1 \| p_0)$ can be calculated by employing the following matrix inversion result

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} (\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1} & -\mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \\ -\mathbf{D}^{-1}\mathbf{C}(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1} & (\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \end{bmatrix} \tag{3.26}$$

Combining (3.16), (3.26) and the fact that $\bar{\mathbf{a}} = [\mathbf{a}^T, \mathbf{0}_{m_3}^T]^T$, we can rewrite the KL divergence as

$$D(p_1 \| p_0) = \frac{1}{2}\mathbf{a}^T(\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz})^{-1}\mathbf{a}. \tag{3.27}$$

The KL divergence is a function of the attack vector $\mathbf{a}$ and the covariance matrices $\boldsymbol{\Sigma_{zp}}$ and $\boldsymbol{\Sigma_{pp}}$, which in turn depend on the PMU locations. We propose to identify the PMU locations that can maximize the KL divergence, thus to maximize the probability of detection. However, the attack vector $\mathbf{a}$ is generally unknown at the detector. To address this problem, we propose to develop the PMU placement algorithm by following the max-min criterion, that is, maximizing the KL divergence under the least detectable attack vector.

3.5.1 Least Detectable Attack Vector

The least detectable attack vector can be obtained by solving the following problem

$$\begin{aligned} \min_{\mathbf{a}} \quad & \mathbf{a}^T(\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz})^{-1}\mathbf{a} \\ \text{s.t.} \quad & \mathbf{a}^T\mathbf{a} = 1. \end{aligned} \tag{3.28}$$

It is well known that the solution to the above problem is the eigenvector associated with the minimum eigenvalue of the matrix $\left(\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}\right)^{-1}$, or equivalently, the eigenvector associated with the maximum eigenvalue of the inverse matrix, $\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}$.

Therefore, the least detectable attack vector, $\mathbf{a}^*$, is the eigenvector corresponding to the eigenvector corresponding to to the maximum eigenvalue of the matrix $\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}$, and we denote the eigenvalue of $\lambda_{\max}\left(\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}\right)$.

The constraint $\mathbf{a}^T\mathbf{a} = 1$ in (3.28) is merely for convenience of representation. In practice the attack vector might be scaled by a certain factor according to the amount of the mean squared error (MSE) that the attacker intends to impose to the state estimator. That is, the actual attack vector might be scaled with a factor α such that the attack MSE, as defined in [7], is

$$\sigma_a^2 = \alpha^2 \|\mathbf{\Sigma}_x \mathbf{H}^T \mathbf{\Sigma}_{zz}^{-1}\mathbf{a}\|_2^2 \tag{3.29}$$

It should be noted that the choice of the scaling factor α will not affect the maximum eigenvalue $\lambda_{\min}\left(\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}\right)^{-1}$, or the probability of detection.

3.5.2 Optimum PMU Placement Algorithm

Define an indicator vector of the PMU locations $\mathbf{b} \in \mathcal{B}^n$, where $\mathcal{B} = \{0, 1\}$, with $b_i = 1$ if a PMU is placed on the i-th bus and $b_i = 0$ otherwise. For a system with k PMUs, we have the constraint $\|\mathbf{b}\|_0 = k$, where $\|\mathbf{b}\|_0$ is the ℓ_0-norm of the vector $\mathbf{b}$. The covariance matrices $\mathbf{\Sigma}_{zp}$ and $\mathbf{\Sigma}_{pp}$ are functions of $\mathbf{H}_{\text{PMU}}$, the structure of which depends on the PMU location indicator vector $\mathbf{b}$.

With the least detectable attack vector identified through the solution of (3.28), we can formulate the max-min PMU placement problem as

$$\begin{aligned} \max_{\mathbf{b}} . \quad & \lambda_{\max}\left(\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}\right) \\ \text{s.t.} \quad & \mathbf{b} \in \mathcal{B}^n \\ & \|\mathbf{b}\|_0 = k \end{aligned} \tag{3.30}$$

This is a non-convex combinatorial problem with a complexity scales with $\binom{n}{k}$. The optimal solution for (3.30) can be obtained by exhaustively searching the $\binom{n}{k}$ possible PMU location vectors $\mathbf{b}$ and find the one that can maximize the largest eigenvalue of $\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}$. The exhaustive search algorithm is given in Algorithm 4.

In the exhaustive search algorithm, we try all the $\binom{n}{k}$ possible values of the location indicator vector $\mathbf{b}$. The one that renders the largest minimum KL divergence, or equivalently the largest maximum eigenvalue of the matrix $\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}$, is the optimum PMU placement vector. Such an approach can provide the optimum performance, at the cost of a high complexity. It is well known that the combinatorial optimization problem is NP hard.

Algorithm 4 Exhausitve Search Algorithm

1: Formulate the set of $\binom{n}{k}$ possible PMU location indicator vectors $\mathcal{D} = \{\mathbf{b}|\mathbf{b} \in \mathcal{B}^n, \|\mathbf{b}\|_0 = k\}$
2: **for** $s = 1$ to $\binom{n}{k}$ **do**
3: pick $\mathbf{b}_s \in \mathcal{D}$
4: formulate $\mathbf{\Sigma}_{zp}$ and $\mathbf{\Sigma}_{pp}$ based on $\mathbf{b}_s$.
5: Calculate the maximum eigenvalue λ_s of the matrix $\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}$.
6: **end for**
7: $s^* = \text{argmax}_s\, \lambda_s$
8: Output: $\mathbf{b}^* = \mathbf{b}_{s^*}$

In each repetition, the calculation of the KL divergence requires the inverses of a $m_3 \times m_3$ matrix, the complexity of which scales with $\mathcal{O}(m_3^3)$, and and finding the maximum eigenvalue of a $m \times m$ matrix, the complexity of which scales with $\mathcal{O}(m^3)$. Therefore, the complexity of the complexity of the exhaustive search algorithm scales with $\mathcal{O}\left((m_3^3 + m^3)\binom{n}{k}\right)$.

Since the optimum PMU locations are identified off-line during the design of a power grid, we can utilize the exhaustive search algorithm for a power system with a moderate number of buses. However, for large systems with large n, we have to resort to low complexity sub-optimum algorithms to reduce the computation complexity.

3.5.3 A Greedy Algorithm

We propose a greedy algorithm to balance the tradeoff between complexity and performance. The greedy algorithm sequentially adds the PMUs to the power system, one at a time. The PMU is added in a greedy manner, that is, each newly added PMU is placed at a location that can maximize the minimum KL divergence of the current system configuration, without considering possible future PMU placements. The greedy algorithm is described in Algorithm 5.

The greedy algorithm requires k steps to find the solution, and one PMU is added at the

Algorithm 5 Greedy algorithm

Initialize the index set of all buses $\mathcal{I} = \{1, 2, \cdots, n\}$
Initialize $\mathbf{b}^* = \mathbf{0}_n$, a length-$n$ all-zero vector.
for $s = 1$ to k **do**
    for $u \in \mathcal{I}$ **do**
        Formulate $\mathbf{b}_u$ by flipping the u-th bit of $\mathbf{b}$.
        formulate $\mathbf{\Sigma}_{zp}$ and $\mathbf{\Sigma}_{pp}$ based on $\mathbf{b}_u$.
        Calculate the maximum eigenvalue of $\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}$
    end for u
    $\hat{u} = \text{argmax}_u \, \lambda_u$
    Set $b^*_{\hat{u}} = 1$.
    Update $\mathcal{I} = \mathcal{I} \backslash \hat{u}$
end for s
Output $\mathbf{b}^*$

end of each step in a greedy manner. At step s, there are $n - s + 1$ buses without PMU, and the algorithm will try to place the PMU on each one of the $n - s + 1$ buses to find the one that can maximize the maximum eigenvalue of the matrix $\mathbf{\Sigma}_{zz} - \mathbf{\Sigma}_{zp}\mathbf{\Sigma}_{pp}^{-1}\mathbf{\Sigma}_{pz}$ under the current system configuration.

In the greedy algorithm, there are a total of $\sum_{s=1}^{k}(n - s + 1) = (n + 1)k + \frac{1}{2}k(k + 1)$ iterations. Inside each iteration, we need to find the inverse of a $m_3 \times m_3$ matrix and perform the eigenvalue decomposition of a $m \times m$ matrix. Thus the complexity of the greedy algorithm scales with $\mathcal{O}\left((m^3 + m_3^3)((n + 1)k + \frac{1}{2}k(k + 1))\right)$.

It should be noted that the PMU placement algorithms are developed by using the least detectable attack vector $\mathbf{a}^*$. Thus they do not require the knowledge of the actual attack vector $\mathbf{a}$, which is unknown and in general different from the least detectable vector.

3.6 Simulation Results

In this section, we present the simulation results by using several standard IEEE bus configurations. The simulations are performed by using the MATPOWER software [18]. In the simulations, it is assumed that the state variables $\mathbf{x}$ are Gaussian distributed with zero mean and covariance matrix $\mathbf{\Sigma_x} = \sigma_x^2 \mathbf{I}_n$. The covariance matrices of the measurement noise are $\mathbf{\Sigma}_{\bar{e}} = \sigma_e^2 \mathbf{I}_{\bar{m}}$ and $\mathbf{\Sigma}_e = \sigma_e^2 \mathbf{I}_m$ for the systems with and without PMUs, respectively. The signal-to-noise ratio (SNR) in dB is defined as $10 \log \frac{\sigma_x^2}{\sigma_e^2}$.

Fig. 3.1 shows the probability of detection as a functions of the number of PMUs installed in a reduced IEEE 57-bus system [1]. The reduced 57-bus system is adopted such that we can compare our algorithms with the critical measurement based PMU placement algorithm proposed in [1]. The system has 57 buses. Hence, the number of state variables is $n = 57 - 1 = 56$, with bus 1 used as the reference bus. In the reduced 57-bus system, there are $m_1 = 32$ power injection measurements and $m_2 = 33$ real power flow measurements, which results in a total number of $m = 65$ measurements if we do not count the ones from the PMUs. The performance of the proposed greedy algorithm is compared to that of the critical measurement based algorithm [1]. The probability of detection increases monotonically with the number of PMUs for both algorithms, because the measurements provided by the PMUs provide extra information for the bad data detection. Under the current system configuration, the maximum detection probability is 0.34, which is achieved when $k = n = 56$, that is, a PMU is installed on each bus. The proposed greedy PMU placement algorithm consistently outperforms the critical measurement based algorithm. The greedy algorithm achieves the maximum detection probability with as little as $k = 31$

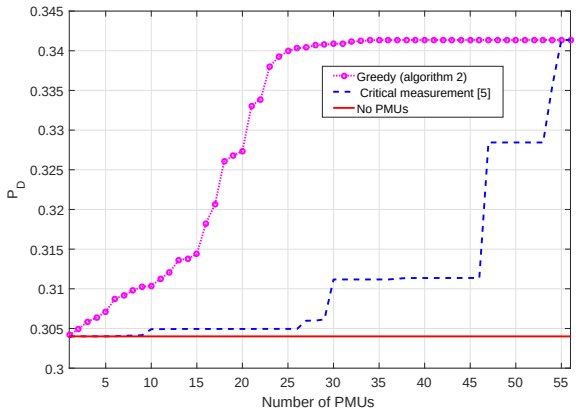

Figure 3.1: The probability of detection as a function of the number of PMUs for IEEE 57-bus system.

PMUs, yet the critical measurement based algorithm needs $k = 55$ PMUs to achieve the same performance. Therefore, the proposed algorithm requires less PMUs than the critical measurement based algorithm, and can achieve a higher detection probability.

Fig. 3.2 shows the probability of detection as a function of the probability of false alarm for the reduced IEEE 57-bus system, that is, the receiver operating characteristic (ROC) curve. The number of PMUs is $k = 31$ for systems with PMUs. The attack vector is the least detectable attack vector scaled by a constant to achieve an attack MSE of -10 dB as in (3.30). All other configurations are the same as Fig. 3.1. The ROC curve for system without PMU is also provided as a baseline. The ROC curves illustrate the tradeoff between probabilities of false alarm and detection. The proposed greedy algorithm significantly outperforms the critical measurement based system and the system without PMU. When $k = 31$, the performance of the critical measurement based system is only slightly better than that of the system without PMU. This is consistent with the results in Fig. 3.1. When the probability of false alarm is 0.2, the detection probabilities for the greedy algorithm, the critical measurement based algorithm, and the system without PMU

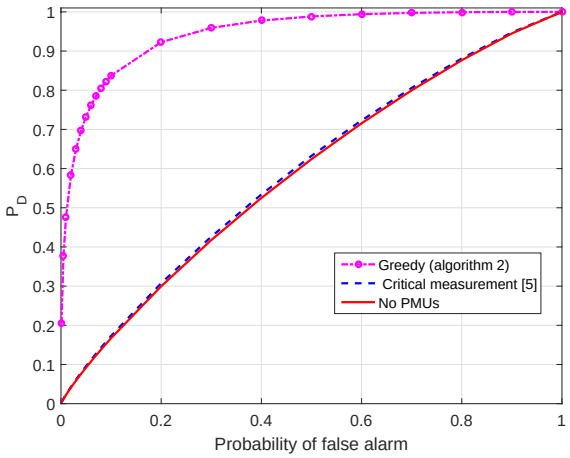

Figure 3.2: The probability of detection as a function of the probability of false alarm for IEEE 57-bus system.

are 0.912, 0.301, and 0.299, respectively.

In the above two examples we do not show the performance of the exhaustive search algorithm due to the prohibitive complexity in a 57-bus system. To demonstrate the performance of the exhaustive algorithm, Fig. 3.3 shows the performance of various algorithms for the IEEE 14-bus system. This system has 14 buses and 20 branches. Hence, the number of state variables is $n = 14 - 1 = 13$. The number of measurements is $m = 20 + 14 = 34$, with $m_1 = 14$ real power injection measurements and $m_2 = 20$ real power flow measurements. The SNR is 10 dB, the probability of false alarm is 0.3, and the attack vector is the least detectable attack vector $\mathbf{a}^*$. The exhaustive search algorithm outperforms the greedy algorithm as expected. However, the performance gap is very small. When $k \geq 8$, systems with the greedy algorithm and the exhaustive search algorithm achieve almost the same performance. Since the optimum performance is achieved at $k \geq 8$, the greedy algorithm can approach the optimum performance with a much lower complexity than the exhaustive search algorithm.

Fig. 3.4 shows the probability of detection as a function of the attack MSE for the

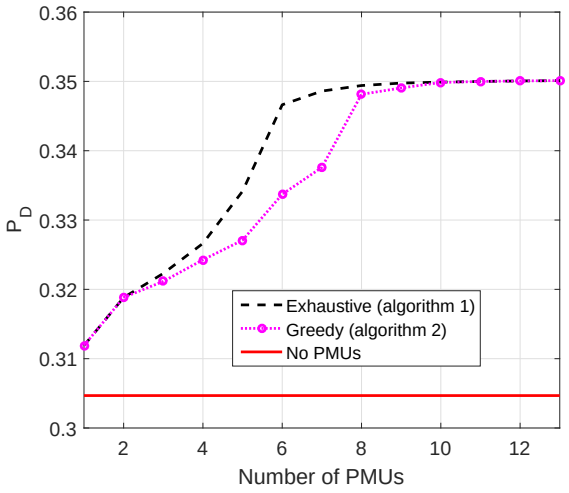

Figure 3.3: The probability of detection as a function of the number of PMUs for IEEE 14-bus system.

reduced IEEE 57-bus system. The number of PMUs is $k = 31$ for systems with PMUs. The attack vector is the least detectable attack vector scaled by a constant to achieve different attack MSEs. All other configurations are the same as Fig. 3.1. As depicted in this figure, the adversary may attempt to scale the optimum attack vector so as to cause more errors or increase the MSE of the state estimator. This, however, comes with a price for it will increase the probability of detection. The greedy algorithm significantly outperforms the other two systems under all system configurations. The greedy algorithm achieves perfect detection (detection probability is 1) when the attack MSE is higher than -7 dB. On the other hand, at the same attack MSE, the detection probabilities of the critical measurement algorithm and system without PMU are only 0.482 and 0.475, respectively.

3.7 Conclusion

We have studied the optimum placement of k PMUs in a power grid with $n \geq k$ buses, with an objective to maximize the probability of detecting malicious data injection under a

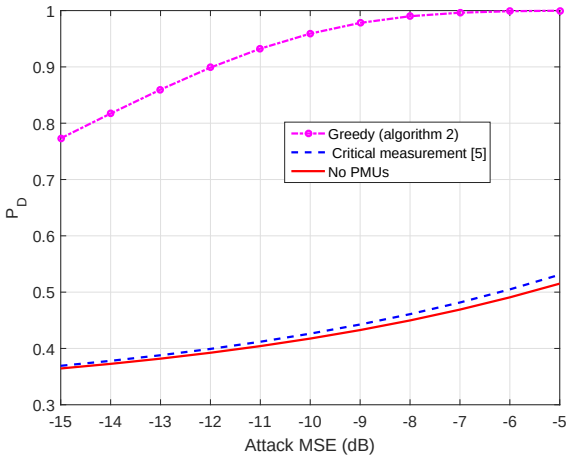

Figure 3.4: The probability of detection as a function of the attack MSE for IEEE 57-bus system.

constraint on probability of false alarm. For a given PMU placement, we first proposed an optimum Neyman-Pearson detector and derived the corresponding probability of detection, which is expressed as an explicit function of the KL divergence between measurement distributions under the alternative and null hypothesis, respectively. It has been shown that the probability of detection is an increasing function in the KL divergence, which in turn depends on the PMU locations. Thus the PMU placement algorithms were developed by using the KL divergence as a design metric. Since the attack vector is unknown, we resorted to a max-min criterion in the PMU placement algorithms, that is, maximizing the KL divergence under the least detectable attack vector. Two PMU placement algorithms have been developed under the max-min criterion. It has been shown by simulations that KL divergence based algorithms achieve significant performance gains over conventional algorithms developed based on critical measurements. For a 57-bus system, the proposed low complexity greedy algorithm achieves the maximum detection probability with only 31 PMUs while 55 PMUs are required by conventional algorithms to achieve the same performance.

3.8 References

[1] J. Chen and A. Abur. Placement of pmus to enable bad data detection in state estimation. *IEEE Trans. on Power Syst.*, 21(4):1608–1615, Nov 2006.

[2] A Simoes Costa, TS Piazza, and A Mandel. Qualitative methods to solve qualitative problems in power system state estimation. *IEEE Trans. on Power Systems*, 5(3):941–949, 1990.

[3] Handschin E, Schweppe F C, Kohlas J, and Fiechter A. Bad data analysis for power system state estimation. *IEEE Trans. Power Apparatus and Systems*, 94(2):329–337, Mar 1975.

[4] AG Expsito and Ali Abur. Generalized observability analysis and measurement classification. In *20th International Conference on Power Industry Computer Applications*, pages 97–103. IEEE, 1997.

[5] John R Hershey and Peder A Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 4, pages IV–317, 2007.

[6] Tung T Kim and H Vincent Poor. Strategic protection against data injection attacks on power grids. *IEEE Trans. Smart Grid*, 2(2):326–333, 2011.

[7] O. Kosut, L. Jia, R. J. Thomas, and L. Tong. Malicious data attacks on the smart grid. *IEEE Trans. on Smart Grid*, 2(4):645–658, Dec 2011.

[8] Q Li, T Cui, Y Weng, R Negi, F Franchetti, and M. D. Ilic. An information-theoretic approach to pmu placement in electric power systems. *IEEE Trans. on Smart Grid*, 4(1):446–456, March 2013.

[9] Q Li, Ri Negi, and M. D. Ilic. Phasor measurement units placement for power system state estimation: A greedy approach. In *IEEE Power and Energy Soc. General Meeting*, pages 1–8, July 2011.

[10] Jeu-Min Lin and Heng Yau Pan. A static state estimation approach including bad data detection and identification in power systems. In *IEEE Power Eng. Soc. General Meeting*, pages 1–7, June 2007.

[11] Yao Liu, Peng Ning, and Michael K Reiter. False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security*, 14(1):13, 2011.

[12] A. G. Phadke, J. S. Thorp, R. F. Nuqui, and M. Zhou. Recent developments in state estimation with phasor measurements. In *IEEE Power and Energy Soc. Conf. Expo.*, pages 1–7, March 2009.

[13] Henrik Sandberg, André Teixeira, and Karl H Johansson. On security indices for state estimators in power networks. In *First Workshop on Secure Control Syst. (SCS), Stockholm, 2010*, 2010.

[14] Dong Wei, Yan Lu, Mohsen Jafari, Paul M Skare, and Kenneth Rohde. Protecting smart grid automation systems against cyberattacks. *IEEE Trans. Smart Grid*, 2(4):782–795, Dec 2011.

[15] J Wu, J Xiong, P Shil, and Y Shi. Optimal pmu placement for identification of multiple power line outages in smart grids. In *IEEE 57th Int. Midwest Symp. on Circuits and Syst. (MWSCAS)*, pages 354–357, Aug 2014.

[16] B. Xu and A. Abur. Observability analysis and measurement placement for systems with pmus. In *IEEE Power and Energy Soc. Conf. Expo.*, pages 943–946 vol.2, Oct 2004.

[17] Zong-Han. Yu and Chin Wen-Long. Blind false data injection attack using pca approximation method in smart grid. *IEEE Trans. Smart Grid*, 6(3):1219–1226, May 2015.

[18] R. D Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas. Matpower: Steady-state operations, planning and analysis tools for power systems research and education. In *IEEE Trans. Power Syst.*, pages 12–19, Feb 2011.

Chapter 4

Low Latency Detection of Sparse False Data Injections in Smart Grids

4.1 Abstract

We study the low latency detection of sparse false data injection in power grids, where an adversary can maliciously manipulate power grid operations by modifying measurements at a small number of smart meters. When a power grid is under attack, the detection delay, which is defined as the time difference between the occurrence and detection of the attack, is critical to the cybersecurity of power grids. A shorter detection delay can ensure the timely deployment of countermeasures to prevent catastrophic impacts from the attack. The objective of this chapter is to develop low latency false data detection algorithms that can minimize the detection delay subject to constraints on false alarm probability. The false data injection can be modeled with a sparse attack vector, with each non-zero element corresponding to one meter under attack. Since neither the support nor the values of the sparse attack vector is known, a new orthogonal matching pursuit cumulative sum (OMP-CUSUM) algorithm is proposed to identify the meters under attack while minimizing the detection delay. In order to recover the support of the sparse vector, we develop a new stopping condition for the iterative OMP algorithm by analyzing the statistical properties of the power grid measurements. Theoretical analysis and simulation results show that the proposed OMP-CUSUM algorithm can efficiently identify the meters under attack, and reliably detect false data injections with low delays while maintaining good detection accuracy.

Keywords

low latency detection, orthogonal matching pursuit, false data injection, cumulative sum.

4.2 Introduction

Smart grid is a combination of power infrastructure, smart meters, and a network of computers [13]. Compared to traditional power grids, smart grid is more robust and efficient owing to the improvement in energy management, control, and system monitoring enabled by the incorporation of networks of computers and smart meters. This, though, comes with a price of grid security and privacy.

Attackers can exploit the cyber-infrastructure of the grid to launch cyber-attacks that can compromise normal grid operations. Some malicious party can launch a cyber-attack by modifying the measurement results obtained by the supervisory control and data acquisition (SCADA) system, such as the power injected or flowing on different buses, and the phase angle of the voltage phasors at different buses. False data injected in the measurement results will affect the real time control of grid operations, thus cause significant damages to power grids. In [11], it is demonstrated that an attacker can take advantage of the configuration of the power system by compromising a small number of meters. The cyber-attacks can be performed by breaking into the communication network of the SCADA system, or by remotely accessing the automation devices such as the remote terminal units (RTU) installed at the substations [20].

A large number of methods have been developed to detect various forms of cyber-attacks in smart grids [11, 7, 10, 3, 22]. Most of these methods rely on residual based detection,

where the detection is performed by analyzing the difference between the estimated and actual power measurements. In addition, almost all existing detection methods are developed to improve detection accuracy or state observability, with little or no attention given to detection delay, which is defined as the time difference between the occurrence and detection of cyberattacks. Detection delay of cyber-attacks is crucial to the stability and operations of power grids. A longer detection delay might comprise the entire power grids and cause power loss to millions of people. On the other hand, a lower detection delay can shorten the response time, such that remedial actions and/or counter measures can be taken to significantly reduce the damages and economic losses caused by cyber-attacks.

Low latency detection can be performed by employing theories from quickest change detection (QCD), which is designed to detect a change in the statistical distribution of a random process [17, 18].The time instant of the occurrence of the change in distribution is denoted as a change point. The objective of QCD is to minimize the detection delay of the change point under the constraints of an upper bound on probability of false alarm (PFA) or a lower bound on average run length (ARL). QCD can be classified into two categories: Bayesian and non-Bayesian change detections. For Bayesian change detection such as the well known Shiryaev procedure [17], the change point is modeled as a random variable, and Bayesian detection methods rely on knowledge of the prior distribution of the change point. When the change point prior distribution is unknown, we can resort to non-Bayesian methods such as the cumulative sum (CUSUM) test [14], which follows the min-max criterion to minimize the detection delay under the worst case change point distribution.

Both Bayesian and non-Bayesian QCD methods require precise knowledge of the statistical distributions of the random process before and after the change. However, it might

be difficult, if not impossible, to obtain the exact distributions in many practical applications, especially the post-change distribution that usually corresponds to abnormal operation conditions. In case of false data injection, it is impossible to obtain the exact post-change distribution, which depends on the unknown attack vector. In [8], the classical CUSUM algorithm is extended with the generalized likelihood ratio test (GLRT), which estimates the unknown parameter in the distribution through maximum likelihood estimation.

There are limited works on low latency detection of false data injection in smart grids. A generalized CUSUM detector is proposed in [9] for false data detection, where the GLRT is utilized to estimate the unknown parameters. The complexity of the generalized CUSUM detector grows exponentially with the number of meters. The complexity mainly arises from the need to identify the meters under attack. A low complexity approximation of the generalized CUSUM is developed in [9], where each meter tracks the false data injection separately. In [5, 6], an adaptive multi-thread CUSUM algorithm is proposed for false data detection in power grids. It is pointed out in [6] that the complexity of GLRT might be too high for practical implementation, thus the Rao test is used for unknown parameter estimation. The elements in the attack vector are assumed to be positive in [5], and such assumption is not always true in practical attacks.

For a large power grid with a large number of buses and meters, it is extremely difficult, if not impossible, for an attacker to attack all meters at once. In almost all cases the attacker can modify the measurements from a small number of meters, that is, the attack is sparse among meters [7]. In recognition of the sparse nature of false data injections, we propose a new orthogonal matching pursuit (OMP) CUSUM algorithm, which utilizes sparse recovery to identify the meters under attack. In the OMP-CUSUM algorithm, the attack vector is

modeled as a sparse vector with dimension equal to the number of power measurements in the grid. The indices of the non-zero elements of the attack vector correspond to meters under attack, and the number of non-zero elements is called the sparsity level. A naive way to locate the meters under attack will be to perform exhaustive search of all possible combinations of attack patterns with GLRT, the complexity of which grows exponentially with the number of buses. To reduce complexity, we resort to the OMP algorithm [19, 15, 1, 2], which is a well known algorithm for sparse signal recovery. Given the fact that neither the sparsity nor the support of the attack vector is known, we develop a new stopping condition for the OMP algorithm by analyzing the statistical properties of the measurements in the grid. The stopping condition can accurately terminate the iterative OMP procedure once all meters under attack are successfully identified, without the prior knowledge of the sparsity level. The results of the OMP are then used in the CUSUM algorithm to minimize the detection delay of false data injection, subject to constraints on the detection accuracy and probability of false alarm. The OMP algorithm and CUSUM is combined in an iterative and sequential manner, that is, for each new group of measurements, OMP is used to estimate the support of the attack vector, and the results are then used for the sequential CUSUM test. Theoretical analysis and simulation results show that the newly proposed OMP-CUSUM algorithm can efficiently and promptly detect false data injections with low complexity, low detection delays, and good detection accuracy.

The remainder of this chapter is organized as follows. The system model and problem formulation are described in Section 4.3. In Section 4.4, we study the quickest attack detection problem using CUSUM test, and highlight the high computational complexity of GLRT-based CUSUM. The OMP-CUSUM algorithm is presented in Section 4.5. In Section

4.6, we develop a worst case attack vector, which will be used to test the performance of the proposed algorithm. Simulation results are given in Section 4.7, and Section 4.8 concludes this chapter.

4.3 Problem Formulation

4.3.1 System Model

We consider a power system with $n+1$ buses. Each bus is equipped with a meter measuring the power flow and power injections. Without loss of generality, we will only consider a system model of active power flows and power injections. Define the set of buses connected to bus i as $\mathcal{X}_i$ with cardinality $c_i = |\mathcal{X}_i|$. Denote the power injection into bus i as P_i, and the power flow from bus i to bus j as P_{ij}, $\forall\ j \in \mathcal{X}_i$. The SCADA system provides a total of $m = m_1 + m_2$ measurements, where $m_1 = n + 1$ is the number of power injections and $m_2 = \frac{1}{2}\sum_{i=1}^{n+1} |\mathcal{X}_i|$ is the number of power flows. Define the power measurement vector as $\mathbf{z} = [z_1, z_2, \ldots, z_m]^T \in \mathcal{R}^{m\times 1}$, where $(\cdot)^T$ is the matrix transpose operator and $\mathcal{R}$ is the set of real numbers.

In phase measurement, one of the $n+1$ buses will serve as a reference, and we only need to measure or estimate the phases of the remaining n buses relative to that of the reference bus. Without loss of generality, assume that the $(n+1)$-th bus is the reference, and define the phase vector of the remaining n buses as $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$, where x_i is the phase of the i-th bus.

The relationship between the observation vector $\mathbf{z}_l$ and the state vector $\mathbf{x}$ can be expressed

as

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{e}, \tag{4.1}$$

where $\mathbf{e} = [e_1, e_2, \ldots, e_m]^T \in \mathcal{R}^{m\times 1}$ is the measurement error vector at the sampling instant l, and $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), \ldots, h_m(\mathbf{x})]^T$ is a function of bus phase angles.

In this chapter we use the standard DC power flow model [16], which results in a linear approximation of the model in (5.1) as

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{e}, \tag{4.2}$$

where $\mathbf{H} \in \mathcal{R}^{m\times n}$ is the measurement Jacobian matrix for the real power flow and power injection measurements. As in [7], we assume that both the state variables $\mathbf{x}$ and measurement noise $\mathbf{e}_l$ are zero-mean Gaussian with covariance matrices $\mathbf{\Sigma}_x$ and $\mathbf{\Sigma}_e$, respectively. That is, $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma_e^2\mathbf{I}_m)$ and $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \sigma_x^2\mathbf{I}_{m_1})$, where $\mathbf{I}_m$ is a size-m identity matrix and σ_e^2 and σ_x^2 are the variances of $\mathbf{e}$ and $\mathbf{x}$, respectively. It is not hard to see that $\mathbf{z}_l$ is Gaussian distributed with zero mean and covariance matrix $\mathbf{\Sigma}_z = \sigma_x^2\mathbf{H}\mathbf{H}^T + \sigma_e^2\mathbf{I}_m$.

Based on the observations in (4.2), the state estimator can obtain an estimate $\hat{\mathbf{x}}$ of the state variable $\mathbf{x}$, such that the mean squared error (MSE) $\sigma_0^2 = \mathbb{E}\left[\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2\right]$ between the estimated and the actual state variables is minimized. This can be achieved with the minimum mean squared error (MMSE) estimator [7], $\hat{\mathbf{x}} = \mathbf{K}\mathbf{z}$, where $\mathbf{K} = \mathbf{\Sigma}_x\mathbf{H}^T\mathbf{\Sigma}_z^{-1}$.

The adversary's intention is to mislead the state estimator into making more estimation errors by modifying the power measurements at certain meters. This could lead to wrong

decisions by the control center, which may decide to increase/decrease power injections at certain buses in the system based on the faulty state estimate.

4.3.2 Mathematical Problem Formulation

An intruder can launch an attack on certain meter readings and intentionally modify the measurements corresponding to these meters. Assume attack happens at time θ and it modifies the measurements on $s < m$ meters. The attack vector can thus be modeled by using a s-sparse attack vector $\mathbf{a}$ of dimension m, which has s non-zero values corresponding to the s meters under attack. The observed measurement vector at the sampling instant l is

$$\mathbf{z}_l = \begin{cases} \mathbf{H}\mathbf{x} + \mathbf{e}, & \text{if } l < \theta \\ \mathbf{H}\mathbf{x} + \mathbf{a} + \mathbf{e}, & \text{if } l > \theta \end{cases}. \tag{4.3}$$

Under the Bayesian setting, the attack time θ is modeled as a random variable with prior probability $\Pr(\theta = k) = \pi_k$, for $k = 1, 2, \ldots$. We want to detect the attack as soon as it occurs, subject to certain performance constraints, such as the probability of false alarm. The detection is performed by using all historical measurement data $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_l$ up to this moment l. To this end, we define the detection procedure δ as a mapping from the observed measurement sequence $\mathbf{z}^{1:l} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_l]$ to a positive integer as

$$\delta : \mathbf{z}^{1:l} \to \{k : k \leq l\}, l = 1, 2, \ldots \tag{4.4}$$

The estimated attack time is thus $\delta(\mathbf{z}^{1:l}) = \hat{\theta} \leq l$ for some l.

Following the detection procedure in (4.4), we define, respectively, the probability of false

alarm (PFA) and the average detection delay (ADD) as

$$\text{PFA}(\delta) = \Pr(\hat{\theta} < \theta), \tag{4.5}$$

and

$$\text{ADD}(\delta) = \mathbb{E}\left[\hat{\theta} - \theta \mid \hat{\theta} > \theta\right]. \tag{4.6}$$

The problem of quickest detection aims to minimize the average detection delay under the constraint of an upper bound of the probability of false alarm. Thus, our problem is formulated as

$$\begin{aligned} \min_{\delta} \quad & \text{ADD}(\delta) \\ \text{s.t.} \quad & \text{PFA}(\delta) \leq \beta. \end{aligned} \tag{4.7}$$

Solving the above problem requires knowledge of the distributions of the observed measurements before and after the attack. From (4.3), define the null hypothesis $\mathcal{H}_0$, which corresponds to the distribution before the attack, and the alternative hypothesis $\mathcal{H}_1$, which corresponds to the distribution after the attack, as

$$\begin{aligned} &\mathcal{H}_0 : \mathbf{z}_l \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma_z}) \\ &\mathcal{H}_1 : \mathbf{z}_l \sim \mathcal{N}(\mathbf{a}, \mathbf{\Sigma_z}), \quad \|\mathbf{a}\|_{\mathbf{0}} = \mathbf{s}. \end{aligned} \tag{4.8}$$

where $\|\mathbf{a}\|_0$ is the ℓ_0 norm that returns the number of non-zero elements in $\mathbf{a}$.

Denote the distributions of $\mathbf{z}_l$ before and after the attack as $f_0(\mathbf{z}_l)$ and $f_1(\mathbf{z}_l|\mathbf{a})$, respectively. It should be noted that the attack vector $\mathbf{a}$ is unknown at the receiver. Thus the post-attack distribution is unknown.

4.4 Quickest Detection with Unknown Attack Vector

In this section, we develop the quickest detection algorithm with an unknown attack vector. The quickest detection algorithm is developed by extending the CUSUM procedure [12] with GLRT.

First we will formulate the CUSUM procedure by assuming that the attack vector $\mathbf{a}$ is known. Then we will extend the CUSUM procedure to the case with unknown attack vector.

If the attack vector $\mathbf{a}$ is known, then the likelihood ratio (LR) at time instant l can be calculated as

$$\lambda_l = \frac{f_1(\mathbf{z}_l \mid \mathbf{a})}{f_0(\mathbf{z}_l)} = \exp\left(\mathbf{z}_l^T \mathbf{\Sigma}_z^{-1}\mathbf{a} - \frac{1}{2}\mathbf{a}^T\mathbf{\Sigma}_z^{-1}\mathbf{a}\right). \tag{4.9}$$

Define the cumulative log-likelihood ratio (LLR) of the samples $\mathbf{z}^{k:l} = \{\mathbf{z}_k, \mathbf{z}_{k+1}, \ldots, \mathbf{z}_l\}$ as

$$\eta_{k:l} = \log\left(\prod_{i=k}^{l}\lambda_i\right) = \sum_{i=k}^{l}(\mathbf{z}_i^T\mathbf{\Sigma}_z^{-1}\mathbf{a} - \frac{1}{2}\mathbf{a}^T\mathbf{\Sigma}_z^{-1}\mathbf{a}). \tag{4.10}$$

With the cumulative LLR given in (4.10), the CUSUM procedure with known attack

vector can be written as [14]

$$\delta = \inf\{l : C_l \geq B\}, \quad \text{with} \;\; C_l = \max_{1\leq k\leq l} \eta_{k:l} \tag{4.11}$$

where B is the threshold chosen such that the constraint on the false alarm probability in (4.7) is satisfied.

The test statistics C_l can be recursively calculated as

$$C_l = \max(0, C_{l-1}) + \log \lambda_l \tag{4.12}$$

with $C_0 = 0$.

The classical CUSUM procedure in (4.11) requires the knowledge of the attack vector $\mathbf{a}$. In practice, $\mathbf{a}$ is unknown at the detector. As a result, we cannot directly calculate the cumulative LLR $\eta_{k:l}$ or the test statistics C_l. This problem can be solved by using GLRT, where we estimate the value of $\mathbf{a}$ by maximizing the cumulative log-likelihood ratio (GLR) as [7]

$$\hat{\mathbf{a}}_{k,l} = \underset{\mathbf{a}\in\Omega_s, s=1,2,\ldots,m}{\operatorname{argsup}} \sum_{i=k}^{l} \left(\mathbf{z}_i^T \boldsymbol{\Sigma}_z^{-1} \mathbf{a} - \frac{1}{2} \mathbf{a}^T \boldsymbol{\Sigma}_z^{-1} \mathbf{a} \right). \tag{4.13}$$

where Ω_s is the set of all s-sparse attack vectors.

For a length-m s-sparse vector $\mathbf{a}$, there are $Q_s = \binom{m}{s}$ sparse patterns. For the q-th sparse pattern, denote the indices of the non-zero elements as $k_{q,1} < k_{q,2} < \cdots < k_{q,s}$, for $q = 1, \cdots, Q_s$. If the attack vector assumes the q-th sparse pattern, then removing the zero elements in $\mathbf{a}$ results in $\mathbf{a}_q = [a_{q,1}, a_{q,2}, \cdots, a_{q,s}]^T$. The cumulative LLR in (4.10) can be

alternatively written as

$$\eta_{k:l}^{(q)} = \sum_{i=k}^{l} (\mathbf{z}_i^T \mathbf{\Lambda}_q \mathbf{a}_q - \frac{1}{2} \mathbf{a}_q^T \mathbf{\Phi}_q \mathbf{a}_q). \tag{4.14}$$

where $\mathbf{\Lambda}_q$ is a $m \times s$ submatrix of $\mathbf{\Sigma}_z^{-1}$, and it is obtained by removing the $m - s$ columns with indices corresponding to the zero elements in the q-th sparse pattern. Similarly, $\mathbf{\Phi}_q$ is a $s \times s$ submatrix of $\mathbf{\Sigma}_z^{-1}$, and it is obtained by removing the $m - s$ rows and columns with indices corresponding to the zero elements in the q-th sparse pattern.

The cumulative GLR can then be obtained by solving the following two optimization problems.

$$\hat{\eta}_{k:l}^{(q)} = \sup_{\mathbf{a}_q \in \mathcal{R}^s} \sum_{i=k}^{l} (\mathbf{z}_i^T \mathbf{\Lambda}_q \mathbf{a}_q - \frac{1}{2} \mathbf{a}_q^T \mathbf{\Phi}_q \mathbf{a}_q) \tag{4.15}$$

$$\hat{\eta}_{k,l} = \max_{q=1,\cdots,Q_s, s=1,2,\ldots,m} \hat{\eta}_{k,l}^{(q)} \tag{4.16}$$

In the above two-step procedure, for each sparsity $1 \leq s \leq m$, we first identify the maximum cumulative GLR for a certain sparse pattern, and the optimum cumulative GLR is then obtained by comparing the results from all $Q = \sum_{s=1}^{m} Q_s = 2^m - 1$ sparse patterns .

We first solve the optimization problem in (4.15). The objective function in (4.15) is quadratic in $\mathbf{a}_q$, so it has a unique solution. Taking the first derivative of the objective function, and setting it to zero, we have

$$\mathbf{\Lambda}_q^T \sum_{i=k}^{l} \mathbf{z}_i - (l - k + 1) \mathbf{\Phi}_q \mathbf{a}_q = 0 \tag{4.17}$$

Thus the vector $\mathbf{a}_q$ that maximizes the objective function is

$$\hat{\mathbf{a}}_q = \frac{1}{l-k+1}\mathbf{\Phi}_q^{-1}\mathbf{\Lambda}_q^T \sum_{i=k}^{l} \mathbf{z}_i \tag{4.18}$$

Combining (4.15) with (4.18) yields

$$\hat{\eta}_{k:l}^{(q)} = \frac{1}{2(l-k+1)} \left(\sum_{i=k}^{l} \mathbf{z}_i \right) \mathbf{\Lambda}_q \mathbf{\Phi}_q^{-1} \mathbf{\Lambda}_q^T \left(\sum_{i=k}^{l} \mathbf{z}_i \right) \tag{4.19}$$

From (4.16) and (4.19), the CUSUM with GLR can be alternatively represented as

$$\delta = \inf \left\{ l : \max_{1 \le k \le l} \quad \max_{q=1 \cdots Q_s, s=1,2,\ldots,m} \hat{\eta}_{k:l}^{(q)} \ge B \right\}. \tag{4.20}$$

The above quickest detection algorithm requires the exhaustive search of all $\sum_{s=1}^{m} Q_s = 2^m - 1$ sparse patterns, and the exhaustive search needs to be performed for each value of $1 \le k \le l$. The complexity grows exponentially with m and it becomes prohibitively high when m is large. A low complexity OMP-CUSUM algorithm is proposed in the next section to balance the tradeoff between complexity and performance.

4.5 Orthogonal Matching Pursuit-CUSUM (OMP-CUSUM) Test

A low complexity OMP-CUSUM algorithm is proposed in this section to balance the tradeoff between complexity and performance. Instead of performing exhaustive search over all sparse patterns, we propose to adopt the OMP algorithm [19] [15] and modify it for the CUSUM test. The OMP algorithm will be used to identify the sparse attack vector that can maximize the cumulative GLR as in (4.13).

In order to employ the OMP in the cumulative GLR calculation, we need to rewrite the optimization problem in (4.13) in the form of a linear optimization. The result is given as follows.

Lemma 4.1: The optimization problem in (4.13) can be alternatively expressed as

$$\min . \quad \|\mathbf{y} - \mathbf{A}\mathbf{a}\|_2^2 \tag{4.21}$$

$$\text{s.}t. \quad \|\mathbf{a}\|_0 = s \tag{4.22}$$

where $\|\mathbf{b}\|_2 = \sqrt{\mathbf{b}^T\mathbf{b}}$ is the ℓ_2-norm of a vector, $\|\mathbf{b}\|_0$ is the ℓ_0-norm, $\mathbf{A} = \mathbf{D}^{-\frac{1}{2}}\mathbf{U}$, $\mathbf{D}$ is a diagonal matrix with the eigenvalues of $\mathbf{\Sigma}_z$ on its main diagonal, $\mathbf{U}$ is the corresponding orthonormal eigenvector matrix, that is, $\mathbf{\Sigma}_z = \mathbf{U}^T\mathbf{D}\mathbf{U}$, and

$$\mathbf{y} = \frac{1}{l-k+1}\mathbf{A}\sum_{i=k}^{l}\mathbf{z}_i. \tag{4.23}$$

The proof is shown in Appendix 4.9.1.

We propose to solve the problem in Lemma 4.1 by using OMP. The basic idea of OMP is to sequentially identify the columns of $\mathbf{A}$ that has the strongest correlation with the vector $\mathbf{y}$, given the fact that $\mathbf{y}$ is a linear combination of the columns of $\mathbf{A}$ corresponding to the non-zero elements of the sparse vector $\mathbf{a}$.

We first describe the OMP algorithm when the sparsity level s is known. The results are then used to develop an OMP algorithm with unknown sparsity level.

4.5.1 OMP with Known Sparsity Level

Based on the optimization problem in Lemma 4.1, the OMP algorithm is described as follows.

- Step 1. Initialize the residual $\mathbf{r}_0 = \mathbf{y}$ and the iteration counter $t = 1$. Initialize the set of non-zero index vector as $\mathcal{I}_0 = \varnothing$. Define the index set $\mathcal{I} = \{1, 2, \cdots, m\}$

- Step 2. At the t-th iteration, find the column of $\mathbf{A}$ that has the maximum absolute inner product with the residual $\mathbf{r}_{t-1}$ as

$$i_t^* = \operatorname{argmax} i \in \mathcal{I} \backslash \mathcal{I}_{t-1} \left| \mathbf{r}_{t-1}^T \mathbf{A}_j \right|, \tag{4.24}$$

  where $\mathbf{A}_j$ denotes the j-th column of $\mathbf{A}$. Update $\mathcal{I}_t = \{i_t^*\} \cup \mathcal{I}_{t-1}$. Denote $\mathbf{A}_{\mathcal{I}_t}$ as a submatrix of $\mathbf{A}$ consisting the columns $\mathbf{A}_i$ with $i \in \mathcal{I}_t$.

- Step 3. Update the residual $\mathbf{r}_t$ by projecting $\mathbf{y}$ onto the null space of $\mathbf{A}_{\mathcal{I}_t}$

$$\mathbf{r}_t = (\mathbf{I}_m - \mathbf{P}_t)\, \mathbf{y} \tag{4.25}$$

  where $\mathbf{I}_m$ is a size m identity matrix, $\mathbf{P}_t = \mathbf{A}_{\mathcal{I}_t} (\mathbf{A}_{\mathcal{I}_t}^T \mathbf{A}_{\mathcal{I}_t})^{-1} \mathbf{A}_{\mathcal{I}_t}^T$ is the projection onto the linear space spanned by the columns of $\mathbf{A}_{\mathcal{I}_t}$.

- Step 4. Set $t = t + 1$, and go back to step 2 until the stopping conditions are met. If the sparsity level s is known, we can stop at the s-th iteration. When the sparsity level is unknown, the stopping condition will be discussed in the next subsection.

- Step 5. If the stopping conditions are met, then stop and output an estimate of $\mathbf{a}$ by

solving the following optimization problem

$$\hat{\mathbf{a}} = \text{argmax} \, \|\mathbf{y} - \mathbf{A}_{\mathcal{I}_t}\mathbf{a}\|_2^2 = (\mathbf{A}_{\mathcal{I}_t}^T \mathbf{A}_{\mathcal{I}_t})^{-1} \mathbf{A}_{\mathcal{I}_t}^T \mathbf{y} \tag{4.26}$$

At time instant l, we need to perform the OMP algorithm for each $1 \leq k \leq l$ to identify the attack vector $\hat{\mathbf{a}}_{k,l}$. Once $\hat{\mathbf{a}}_{k,l}$ is identified, then we can update the cumulative GLR as

$$\hat{\eta}_{k:l} = \sum_{i=k}^{l} \left(\mathbf{z}_i^T \mathbf{\Sigma}_z^{-1} \hat{\mathbf{a}}_{k,l} - \frac{1}{2} (\hat{\mathbf{a}}_{k,l})^T \mathbf{\Sigma}_z^{-1} \hat{\mathbf{a}}_{k,l} \right) \tag{4.27}$$

With the cumulative GLR defined in (4.27), the CUSUM with GLR can then be written as

$$\delta = \inf \left\{ l : \max_{1 \leq k \leq l} \hat{\eta}_{k:l} \geq B \right\}. \tag{4.28}$$

In some sparse sensing applications such as those in [19] [15] where the signal sparsity is known, the above algorithm stops in Step 4 when the iteration $t = s$. In contrast, for our problem, the detector has no knowledge of the sparsity level, s, of the attack vector, which is crucial to the performance of the OMP algorithm. In the next sub-section, we solve this problem by developing new stopping conditions for the OMP algorithm based on the residual analysis.

4.5.2 OMP with Unknown Sparsity Level

We propose to develop the stopping condition of the OMP algorithm by analyzing the statistical properties of the residual $\mathbf{r}_t$ in (4.25) at each iteration t. From (4.3), (4.23), and

(4.25), the residual can be written as

$$\mathbf{r}_t = \frac{1}{L}\mathbf{P}_t^{\perp}\mathbf{A}\sum_{i=k}^{l}\mathbf{z}_i \tag{4.29}$$

where $\mathbf{P}_t^{\perp} = \mathbf{I}_m - \mathbf{P}_t \in \mathcal{R}^{m\times m}$ projects to the null space of the column space spanned by $\mathbf{A}_{\mathcal{I}_t}$, $L = l - k + 1$, and $\mathbf{z}_i$ is the observation vector defined in (4.3).

Denote the true support set of $\mathbf{a}$ as $\mathcal{I}_s^*$, that is, the elements of $\mathcal{I}_s^*$ are the indices of the non-zero elements of $\mathbf{a}$. If the support set of the attack vector is successfully recovered at the t-th iteration, that is, $\mathcal{I}_s^* \subseteq \mathcal{I}_t$, then the residual $\mathbf{r}_t$ does not contain any information of $\mathbf{a}$. We denote this as the null hypothesis $\mathcal{H}_0$, and the OMP algorithm should stop once $\mathcal{H}_0$ is detected. On the other hand, if $\mathcal{I}_s^* \bigcap \mathcal{I}_t \neq \mathcal{I}_s^*$, that is, the index of at least one non-zero elements of $\mathbf{a}$ is not in $\mathcal{I}_t$, then $\mathbf{r}_t$ still depends on $\mathbf{a}$. This is denoted as the alternative hypothesis $\mathcal{H}_1$, and the algorithm needs to continue to the next iteration under $\mathcal{H}_1$.

From (4.3) and (4.29), the hypothesis test on the residual $\mathbf{r}_t$ can be written as

$$\begin{aligned}
\mathcal{H}_0 &: \mathbf{r}_t = \frac{1}{L}\mathbf{P}_t^{\perp}\mathbf{A}\sum_{i=k}^{l}\mathbf{v}_i \\
\mathcal{H}_1 &: \mathbf{r}_t = \frac{1}{L}\mathbf{P}_t^{\perp}\mathbf{A}\sum_{i=k}^{l}\left(\mathbf{v}_i + \mathbf{a}\right), \quad \text{if } \mathbf{a} \neq \mathbf{0},
\end{aligned} \tag{4.30}$$

where $\mathbf{v}_i = \mathbf{H}\mathbf{x}_i + \mathbf{e}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{z}})$.

Since $\mathbf{P}_t^{\perp} \in \mathcal{R}^{m\times m}$ projects to the null space of $\mathbf{A}_{\mathcal{I}_t} \in \mathcal{R}^{m\times t}$, which has a column rank of t, the rank of $\mathbf{P}_t^{\perp}$ is $m - t$. Due to row-rank deficiency of the matrix $\mathbf{P}_t^{\perp}$, the residual $\mathbf{r}_t$ in (4.30) is a degenerate Gaussian distribution. However, as illustrated in [21], we can formulate a full rank sub-matrix $\mathbf{C}_t \in \mathcal{R}^{(m-t)\times m}$ by choosing $m - t$ arbitrary rows of $\mathbf{P}_t^{\perp}$.

Without loss of generality, we formulate $\mathbf{C}_t \in \mathcal{R}^{(m-t)\times m}$ by using the first $m-t$ rows of $\mathbf{P}_t^{\perp}$. Then the the hypothesis in (4.30) can be reformulated as

$$\begin{aligned}\mathcal{H}_0 : \tilde{\mathbf{r}}_t &= \frac{1}{L}\mathbf{C}_t\mathbf{A}\sum_{i=k}^{l}\mathbf{v}_i \quad \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma_t}) \\ \mathcal{H}_1 : \tilde{\mathbf{r}}_t &= \frac{1}{L}\mathbf{C}_t\mathbf{A}\sum_{i=k}^{l}(\mathbf{v}_i + \mathbf{a}) \quad \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{\Sigma}_t),\end{aligned} \tag{4.31}$$

We have the following lemma regarding the distribution of $\tilde{\mathbf{r}}_t$ under the null and alternative hypotheses, respectively.

Lemma 4.2: The reduced residual vector $\tilde{\mathbf{r}}_t$ is Gaussian distributed under both the null and alternative hypotheses, and

$$\begin{aligned}\tilde{\mathbf{r}}_t|\mathcal{H}_0 &\sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_t) \\ \tilde{\mathbf{r}}_t|\mathcal{H}_1 &\sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{\Sigma}_t)\end{aligned} \tag{4.32}$$

where $\mathbf{\Sigma}_t = \frac{1}{L}\mathbf{C}_t\mathbf{C}_t^T$, $\boldsymbol{\mu}_t = \frac{\rho}{L}\mathbf{C}_t\mathbf{A}\mathbf{a}$, and ρ is the number of corrupted measurements out of the L collected measurement samples.

The proof is shown in Appendix 4.9.2.

It should be noted that the value of ρ used in Lemma 4.2 is unknown, and we do not need ρ for the stopping condition developed in this subsection.

We can perform likelihood ratio test (LRT) at Step 4 of the OMP algorithm to detect between the null and alternative hypothesis. If $\mathcal{H}_0$ is detected at iteration t, then the algorithm stops. Otherwise we move on to the next iteration. The development of the

optimum decision rule requires the statistical distribution of $\mathbf{y}$, which is given in the following lemma.

Lemma 4.3: The distributions of $\mathbf{y} = \frac{1}{L}\mathbf{A}\sum_{i=k}^{l}\mathbf{z}_i$ under the null and alternative hypotheses are, respectively,

$$\begin{aligned}\mathbf{y}|\mathcal{H}_0 &\sim \mathcal{N}\left(\mathbf{0}, \frac{1}{L}\mathbf{I}_m\right) \\ \mathbf{y}|\mathcal{H}_1 &\sim \mathcal{N}\left(\mathbf{A}\mathbf{a}, \frac{1}{L}\mathbf{I}_m\right).\end{aligned} \tag{4.33}$$

The proof is shown in Appendix 4.9.3.

Theorem 4.1: For a given probability of false positive σ, The OMP algorithm with unknown sparsity stops at the t-th iteration if the following condition is met

$$T_t = \mathbf{y}^T\mathbf{C}_t^T\mathbf{\Sigma}_t^{-1}\mathbf{C}_t\mathbf{y} < \lambda_t. \tag{4.34}$$

The threshold λ_t is calculated as a function of the probability of false positive $\sigma = \Pr(T > \lambda_t|\mathcal{H}_0)$ as

$$\lambda_t = 2\Gamma^{-1}\left(\frac{m-t}{2}, \sigma\Gamma(\frac{m-t}{2})\right), \tag{4.35}$$

where $\Gamma(m) = \int_0^\infty x^{m-1}\exp(-x)dx$ is the Gamma function, $\Gamma(M,b) = \int_b^\infty y^{M-1}\exp(-y)dy$ is the upper incomplete Gamma function, and $\Gamma^{-1}(M,y)$ is its inverse.

The proof is shown in Appendix 4.9.4.

4.6 Optimum Attack Vector From Adversary's Perspective

In order to evaluate the performance of the proposed algorithm, we design a worst case attack vector that is difficult to detect, but can cause large damage to the system. Then we can evaluate the performance of the proposed OMP-CUSUM algorithm by using the worst case attack vector.

For the CUSUM procedure, the average detection delay is asymptotically inversely proportional to the Kullback-Leibler (KL) divergence between the distributions before and after change [18, 8]. Thus the adversary can make the attack harder to detect by minimizing the KL divergence between $f_1(\mathbf{z}|\mathbf{a})$ and $f_0(\mathbf{z})$. Under the Gaussian assumption, the KL divergence between $f_1(\mathbf{z}|\mathbf{a})$ and $f_0(\mathbf{z})$ can be calculated as [4]

$$\begin{aligned} D(f_1\|f_0) &= \frac{1}{2}\left[\mathrm{tr}\left(\mathbf{\Sigma}_z\mathbf{\Sigma}_z^{-1}\right) + \mathbf{a}^T\mathbf{\Sigma}_z^{-1}\mathbf{a} - 1 + \ln\left(\frac{\mid \mathbf{\Sigma}_z \mid}{\mid \mathbf{\Sigma}_z \mid}\right)\right] \\ &= \frac{1}{2}\mathbf{a}^T\mathbf{\Sigma}_\mathbf{z}^{-\mathbf{1}}\mathbf{a}. \end{aligned} \tag{4.36}$$

In general, it is difficult for an attacker to gain access to every meter in the system. Instead, the adversary might have access to a subset of s meters with indices $\mathcal{I}_s^* = \{i_1, i_2, \cdots, i_s\}$. Denote $\mathbf{a}_s = \mathbf{a}_{\mathcal{I}_s^*}$, which contains the elements a_k with $k \in \mathcal{I}_s^*$. Thus the KL divergence can be rewritten as

$$D(f_1\|f_0) = \frac{1}{2}\mathbf{a}_s^T\mathbf{\Phi}_\mathbf{s}\mathbf{a}_\mathbf{s} \tag{4.37}$$

where $\mathbf{\Phi}_s$ is an $s \times s$ submatrix of $\mathbf{\Sigma}_z^{-1}$, and it contains the rows and columns of $\mathbf{\Sigma}_z^{-1}$ with indices in $\mathcal{I}_s^*$.

The damage caused by the attack vector can be measured by the energy of the attack, or equivalently, the additional mean square error of state estimation due to the attack. From [7], the energy of the attack can be calculated as

$$\sigma_a^2 = \|\mathbf{\Sigma}_x \mathbf{H}^T \mathbf{\Sigma}_z^{-1} \mathbf{a}\|_2^2 = \|\mathbf{\Sigma}_x \mathbf{H}^T \mathbf{\Lambda}_s \mathbf{a}_s\|_2^2 = \|\mathbf{K}_s \mathbf{a}_s\|_2^2. \tag{4.38}$$

where $\mathbf{\Lambda}_s$ contains the columns of $\mathbf{\Sigma}_z^{-1}$ with indices in $\mathcal{I}_s^*$, and $\mathbf{K}_s = \mathbf{\Sigma}_x \mathbf{H}^T \mathbf{\Lambda}_s$.

We can then design the worst case attack vector by solving the following optimization problem.

$$\begin{aligned} \min_{\mathbf{a}_s \in \Omega_s} \quad & \mathbf{a}_s^T \mathbf{\Phi}_s \mathbf{a}_s \\ \text{s.t.} \quad & \|\mathbf{K}_s \mathbf{a}_s\|_2^2 \geq \gamma, \end{aligned} \tag{4.39}$$

where γ is the minimum attack energy desired by the adversary. A similar approach, but with different objective function, has been taken in [7], where the attack vector is designed to minimize the estimation residue error subject to the constraint on a lower bound of the attack energy.

The optimization problem in (4.39) can be solved analytically, and the solution is given as follows.

Corollary 4.1: The optimum attack vector that solves the optimization problem in (4.39)

is

$$\mathbf{a}_s^* = \sqrt{\frac{\gamma}{\|\mathbf{K}_s \mathbf{u}_{\min}\|_2^2}} \mathbf{u}_{\min} \tag{4.40}$$

where $\mathbf{u}_{\min}$ is the generalized eigenvector corresponding to the minimum eigenvalue of the matrix pair $(\mathbf{\Phi}_s, \mathbf{K}_s^T \mathbf{K}_s)$.

The proof is shown in Appendix 4.9.5.

4.7 Simulation Results

In this section, we present the simulation results by using several standard IEEE bus configurations. The simulations are performed by using the MATPOWER software [23]. In the simulations, it is assumed that the state variables $\mathbf{x}$ are Gaussian distributed with zero mean and covariance matrix $\mathbf{\Sigma}_{\mathbf{x}} = \sigma_x^2 \mathbf{I}_n$. The covariance matrices of the measurement noise is $\mathbf{\Sigma}_e = \sigma_e^2 \mathbf{I}_m$. The signal-to-noise ratio (SNR) in dB is defined as $10 \log \frac{\sigma_x^2}{\sigma_e^2}$. The change point is assumed to follow a geometric distribution with parameter p_0, that is, $\pi_k = (1 - p_0)^{k-1} p_0$. In all simulations, we set SNR = 10 dB and $p_0 = 0.1$.

Fig. 4.1 shows the probability that the OMP algorithm will meet the stopping conditions in Theorem 4.1 as a function of the probability of false positive (PFP) in (4.56) for the IEEE 14-bus system. The attack vector is randomly generated with sparsity $s = 5$ and then scaled as in (4.40) with $\gamma = \sigma_a^2 = 0.0217$. Each point on the curves in Fig. 4.1 was obtained by running 10,000 trials. As predicted by our theoretical analysis, the probability that the OMP algorithm stops increases with the number of iterations. With $s = 5$, ideally the algorithm should stop at the 5-th iteration. Stopping at $t < 5$ iterations means that some attacks are

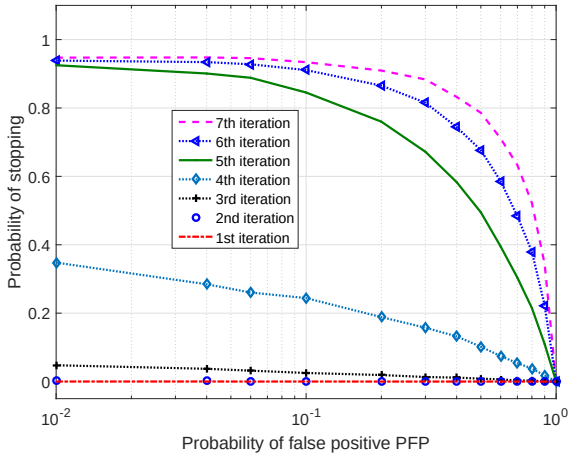

Figure 4.1: The probability of stopping versus the probability of false positive for the IEEE 14-bus system.

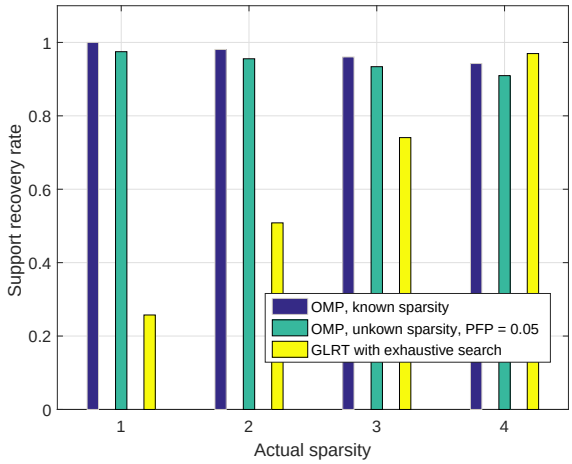

Figure 4.2: The average recovered sparsity for the OMP with known sparsity , OMP with unknown sparsity, and GLR with exhaustive search as a function of the actual sparsity for the IEEE 14-bus system.

not identified. In this experiment, the probability of stopping is relatively low (less than 0.27 at PFP $= 0.05$) during the first four iterations because the residual vector (4.25) still contains non-zero components of the attack vector. On the other hand, the stopping probability rises significantly after the 4th iterations. At PFP $= 0.05$, the probabilities of stopping at the 5th, 6th, and 7th iterations are 0.89, 0.93, and 0.95, respectively.

In Fig. 4.2, the support recovery rates of the proposed OMP algorithms with and without known sparsity level are compared with the GLRT algorithm with exhaustive search

described in Section 4.4. The support recovery rate is defined as the percentage of trials that can successfully recover the exact support of the attack vector. For the case with known sparsity level, the value of s is known at the detector, but the actual support of the sparse vector is unknown. For the case with unknown sparsity level, neither s nor its support is known. The recovery rate of the OMP algorithm with unknown sparsity is very close to that with known sparsity, which means the stopping condition in Theorem 4.1 is very effective. The recovery rates of OMP algorithms with known or unknown sparsity are consistently above 90% for all sparsity levels. It is interesting to note that the performance of the OMP algorithms increases with s, yet this trend is reversed for the GLRT algorithm with exhaustive search. This can be explained by the fact that the primarily objective of the OMP is to recover the correct support of the attack vector and then subsequently find the optimum values of the entries in the support. On the contrary, the GLRT algorithm with exhaustive search attempts to find the support and values together to maximize the objective in (4.19). The GLRT with exhaustive search performs poorly when $s < 4$. At $s = 4$, the GLRT algorithm with exhaustive search outperforms the OMP algorithm, because the OMP algorithm's performance generally deteriorates as the ratio of the sparsity and number of measurements grows.

The average detection delays are shown as functions of the false alarm probability for the IEEE 14-bus and 57-bus systems in Figs. 4.3 and 4.4, respectively. The attack vectors are designed in the same way as those used in Fig. 4.1 with $\gamma = 0.0217$. As in [5], the threshold B in (4.11) is obtained by fixing the probability of false alarm, PFA $= \beta$, and setting $B = \log \frac{\beta^{-1}}{p_o}$. The probability of false positive for the OMP stopping conditions is $\sigma = 0.01$. Each point in the figures was obtained through Monte Carlo simulations with 4,000

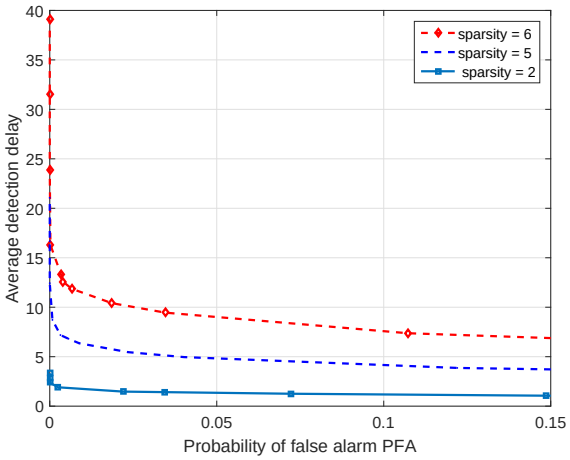

Figure 4.3: The Average detection delay of the OMP-CUSUM as a function of false alarm probability with various sparsity values (number of attacked meters) for IEEE 14-bus system.

trials. The average detection delay is represented as the number of observation samples. For both the 14- and 57-bus systems, the ADD decreases as the sparsity s becomes smaller, mainly due to the fact that a smaller s results in a better recovery rate of the support of the attack vector. As expected, the ADD is a monotonic decreasing function in PFA for all system configurations. At PFA $= 0.05$, the ADDs of the 14-bus system with $s = 2, 5$, and 6 are 1.4, 4.8, and 9.0 samples, respectively; the ADDs of the 57-bus system with $s = 2, 6$, and 15 are 1.04, 1.32, and 1.95 samples, respectively. Therefore the algorithms can detect various attacks with low latency and high accuracy.

Fig. 4.5 shows the average detection delay as a function of the normalized attack energy $\frac{\gamma}{\sigma_x^2}$. Two different attack vectors are considered. One is the optimum attack vector as designed in Corollary 4.1, and the other one is a random attack vector normalized to meet the attack energy constraint. For the random attack vector, each point in the curve is obtained by averaging over 10,000 different realization of the random attack vectors. The probability of false alarm is set as $\beta = 0.072$, and the sparsity level is $s = 2$. All other

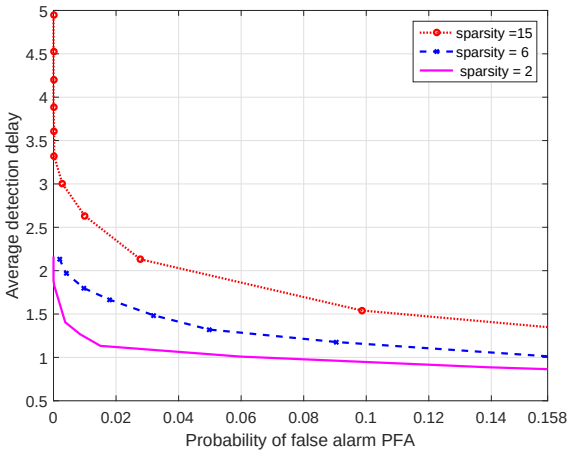

Figure 4.4: The Average detection delay of the OMP-CUSUM as a function of false alarm probability with various sparsity values (number of attacked meters) for IEEE 57-bus system.

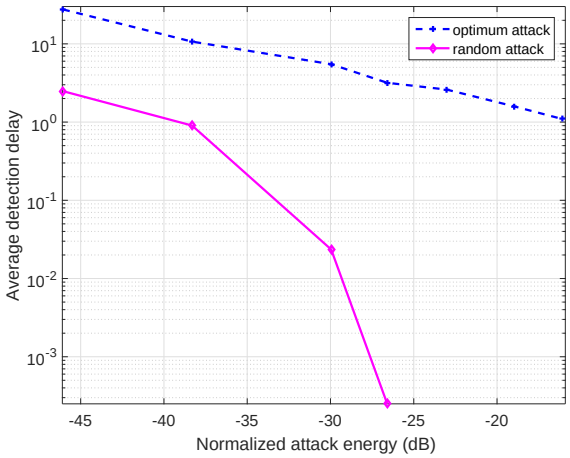

Figure 4.5: The Average detection delay of the OMP-CUSUM with unknown sparsity as a function of attack energy with both random and optimum attack vectors for IEEE 14-bus system.

parameters are the same as in Fig. 4.3. The proposed OMP-CUSUM algorithm can reliably and quickly detect both types of attack vectors, but the optimum attack vector is more difficult to detect than the random one. For instance, at the average detection delay of 1 sample, the minimum detectable normalized attack energy of the optimum and random attack vectors are -16.1 dB and -38.3 dB, respectively.

4.8 Conclusion

Quickest detections of false data injected to smart grids have been studied in this chapter. Motivated by the fact that a malicious party usually can only attack a small number of meters, we have developed a new OMP-CUSUM algorithm for low latency detection of false data injections. Unlike conventional CUSUM algorithm that relies on the knowledge of the attack vector, the OMP-CUSUM algorithm can efficiently identify the meters under attack, and minimize the detection delay of bad data injection under the constraint of the probability of false alarm. An optimum attack vector has also been developed to test and validate the performance of the proposed algorithm. Simulation results have shown that the proposed OMP-CUSUM algorithm can accurately and reliably detect intrusions with small delays and low complexities, and the detection performance improves as the sparsity level of the attack vector decreases.

4.9 Appendix

4.9.1 Proof of Lemma 4.1

Define $\mathbf{w}_{k:l} = \frac{1}{l-k+1}\sum_{i=k}^{l}\mathbf{z}_i$. Then the objective function in (4.13) can be alternatively written as

$$J = (l-k+1)\left(\mathbf{w}_{k:l}^T\boldsymbol{\Sigma}_z^{-1}\mathbf{a} - \frac{1}{2}\mathbf{a}^T\boldsymbol{\Sigma}_z^{-1}\mathbf{a}\right) \tag{4.41}$$

Based on the eigenvalue decomposition of $\mathbf{\Sigma}_z$, we have

$$\mathbf{\Sigma}_z^{-1} = \mathbf{U}^T\mathbf{D}^{-1}\mathbf{U} = \mathbf{A}^T\mathbf{A} \tag{4.42}$$

In addition, $\mathbf{y} = \mathbf{A}\mathbf{w}_{k:l}$.

Then (4.41) can be alternatively represented as

$$\frac{2J}{l-k+1} = 2\mathbf{y}^T\mathbf{A}\mathbf{a} - \mathbf{a}^T\mathbf{A}^T\mathbf{A}\mathbf{a} - \mathbf{y}^T\mathbf{y} + \mathbf{y}^T\mathbf{y} \tag{4.43}$$

$$= -\|\mathbf{y} - \mathbf{A}\mathbf{a}\|^2 + \mathbf{y}^T\mathbf{y} \tag{4.44}$$

Finding $\mathbf{a}$ to maximize J is thus equivalent to minimize $\|\mathbf{y} - \mathbf{A}\mathbf{a}\|^2$. This completes the proof.

4.9.2 Proof of Lemma 4.2

Since $\mathbf{v}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_z)$ and $\tilde{\mathbf{r}}_t$ is a linear combination of $\mathbf{v}_i$, $\tilde{\mathbf{r}}_t$ is Gaussian distributed under both the null and alternative hypotheses.

Under $\mathcal{H}_1$, the mean of $\tilde{\mathbf{r}}_t$ is $\mathbb{E}[\tilde{\mathbf{r}}_t|\mathcal{H}_1] = \frac{1}{L}\mathbf{C}_t\mathbf{A}\sum_{i=k}^{l}\mathbf{a} = \frac{\rho}{L}\mathbf{C}_t\mathbf{A}\mathbf{a}$.

The covariance matrices under both $\mathcal{H}_0$ and $\mathcal{H}_1$ are

$$\mathbf{\Sigma}_t = \mathbb{E}\left[\tilde{\mathbf{r}}_t\tilde{\mathbf{r}}_t^T \mid \mathcal{H}_0\right] = \mathbb{E}\left[\left(\tilde{\mathbf{r}}_t - \boldsymbol{\mu}_t\right)\left(\tilde{\mathbf{r}}_t - \boldsymbol{\mu}_t\right)^T \mid \mathcal{H}_1\right]$$

$$= \frac{1}{L}\mathbf{C}_t\mathbf{A}\mathbf{\Sigma}_z\mathbf{A}^T\mathbf{C}_t^T = \frac{1}{L}\mathbf{C}_t\mathbf{C}_t^T \tag{4.45}$$

where the last equality is based on the fact that $\mathbf{A} = \mathbf{D}^{-\frac{1}{2}}\mathbf{U}$ and $\mathbf{\Sigma}_z = \mathbf{U}^T\mathbf{D}\mathbf{U}$. This

completes the proof.

4.9.3 Proof of Lemma 4.3

Since $\mathbf{y}$ is the linear combination of $\mathbf{z}_i$, which is Gaussian distributed, $\mathbf{y}$ is Gaussian distributed under both the null and alternative hypotheses.

Under the null hypothesis, we have $\mathbf{z}_i = \mathbf{v}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_z)$. Given that $\mathbf{z}_i$ and $\mathbf{z}_j$ are independent for $i \neq j$, the covariance matrix of $\mathbf{y}$ is

$$\mathbb{E}[\mathbf{y}\mathbf{y}^T|\mathcal{H}_0] = \frac{1}{L^2}\mathbf{A}\sum_{i=k}^{l}\sum_{j=k}^{l}\mathbb{E}[\mathbf{z}_i\mathbf{z}_j^T]\mathbf{A}^T \tag{4.46}$$

$$= \frac{1}{L^2}\sum_{i=k}^{l}\mathbf{A}\mathbf{\Sigma}_z\mathbf{A}^T = \frac{1}{L}\mathbf{I}_m \tag{4.47}$$

Under the alternative hypothesis, we have $\mathbf{z}_i = \mathbf{v}_i + \mathbf{a} \sim \mathcal{N}(\mathbf{a}, \mathbf{\Sigma}_z)$. Thus $\mathbb{E}[\mathbf{y}|\mathcal{H}_1] = \frac{1}{L}\mathbf{A}\sum_{i=k}^{l}\mathbf{a} = \mathbf{A}\mathbf{a}$. The covariance matrix under the alternative hypothesis can be derived in a similar manner as (4.46). This completes our proof.

4.9.4 Proof of Theorem 4.1

The LLR for the hypothesis test in (4.31) is

$$\log L(\tilde{\mathbf{r}}_t, \mathbf{a}) = \log\frac{\Pr(\tilde{\mathbf{r}}_t|\mathbf{a}, \mathcal{H}_1)}{\Pr(\tilde{\mathbf{r}}_t|\mathcal{H}_0)}$$

$$= \tilde{\mathbf{r}}_t^T\mathbf{\Sigma}_t^{-1}\boldsymbol{\mu}_t - \frac{1}{2}\boldsymbol{\mu}_t^T\mathbf{\Sigma}_t^{-1}\boldsymbol{\mu}_t \tag{4.48}$$

where $\boldsymbol{\mu}_t = \frac{\rho}{L}\mathbf{C}_t\mathbf{A}\mathbf{a}$. Since $\mathbf{a}$ is unknown, we can perform GLRT, where $\mathbf{a}$ can be estimated by maximizing the LLR. Setting $\frac{\partial \log L(\tilde{\mathbf{r}}_t)}{\partial \mathbf{a}} = 0$ yields

$$\frac{\rho}{L}\mathbf{C}_t\mathbf{A}\hat{\mathbf{a}} = \tilde{\mathbf{r}}_t, \tag{4.49}$$

where $\hat{\mathbf{a}}$ is the maximum likelihood (ML) estimate of $\mathbf{a}$.

Replacing $\mathbf{a}$ in (4.48) with $\hat{\mathbf{a}}$ in (4.49) results in

$$\log L(\tilde{\mathbf{r}}_t, \hat{\mathbf{a}}) = \frac{1}{2}\tilde{\mathbf{r}}_t^T \boldsymbol{\Sigma}_t^{-1}\tilde{\mathbf{r}}_t^T \tag{4.50}$$

Given that $\tilde{\mathbf{r}}_t = \frac{1}{L}\mathbf{C}_t\mathbf{A}\sum_{i=k}^{l}\mathbf{z}_i = \mathbf{C}_t\mathbf{y}$ and $\boldsymbol{\Sigma}_t = \frac{1}{L}\mathbf{C}_t\mathbf{C}_t$, the LLR can be written as

$$\log L(\tilde{\mathbf{r}}_t, \hat{\mathbf{a}}) = \frac{L}{2}\mathbf{y}^T\mathbf{C}_t^T(\mathbf{C}_t\mathbf{C}_t^T)^{-1}\mathbf{C}_t\mathbf{y} \tag{4.51}$$

It is apparent that $\mathbf{C}_t^T(\mathbf{C}_t\mathbf{C}_t^T)^{-1}\mathbf{C}_t$ is a projection matrix that projects to the linear space spanned by the $(m-t)$ rows of $\mathbf{C_t}$, thus it can be represented as

$$\mathbf{C}_t^T(\mathbf{C}_t\mathbf{C}_t^T)^{-1}\mathbf{C}_t = \mathbf{V}\cdot\mathrm{Diag}[\mathbf{1}_{m-t};\mathbf{0}_t]\cdot\mathbf{V}^T \tag{4.52}$$

where $\mathrm{diag}[\mathbf{1}_{m-t};\mathbf{0}_t]$ is a $m\times m$ diagonal matrix with the vector $[\mathbf{1}_{m-t};\mathbf{0}_t]$ on its main diagonal, that is, the first $m-t$ elements of the main diagonal are 1s, and all the rest are 0s.

Define $\mathbf{w} = \sqrt{L}\mathbf{V}\mathbf{y}$. Since $\mathbf{y}$ is Gaussian distributed as in Lemma 4.3, it can be easily

shown that

$$\begin{aligned}\mathbf{w}|\mathcal{H}_0 &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m) \\ \mathbf{w}|\mathcal{H}_1 &\sim \mathcal{N}\left(\sqrt{L}\mathbf{V}\mathbf{A}\mathbf{a}, \mathbf{I}_m\right)\end{aligned} \tag{4.53}$$

The LLR in (4.51) can then be alternatively written as

$$\log L(\tilde{\mathbf{r}}_t, \hat{\mathbf{a}}) = \frac{1}{2}\mathbf{w} \cdot \mathrm{Diag}[\mathbf{1}_{m-t}; \mathbf{0}_t] \cdot \mathbf{w} = \frac{1}{2}\sum_{k=1}^{m-t} w_i^2 \tag{4.54}$$

From (4.54), the hypothesis test can be represented as

$$\begin{aligned}T_t &= \mathbf{y}^T \mathbf{C}_t^T \mathbf{\Sigma}_t^{-1} \mathbf{C}_t \mathbf{y} = L\mathbf{y}\mathbf{C}_t^T(\mathbf{C}_t\mathbf{C}_t^T)^{-1}\mathbf{C}_t\mathbf{y} \\ &= \sum_{k=1}^{m-t} w_i^2 \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \lambda_t\end{aligned} \tag{4.55}$$

where the threshold λ_t will be determined based on the probability of false positive σ.

Given the distribution of $\mathbf{w}$ in (4.53), it can be easily shown that T follows the χ^2 distribution with $m-t$ degrees of freedom under $\mathcal{H}_0$ and the non-central χ^2 distribution with $m-t$ degrees of freedom under $\mathcal{H}_1$.

Therefore, the distributions of T under $\mathcal{H}_0$ is

$$f_0(x \mid \mathcal{H}_0) = \frac{x^{\frac{m-t}{2}-1}\exp(-\frac{x}{2})}{2^{\frac{m-t}{2}}\Gamma(\frac{m-t}{2})}.$$

The probability of false positive is

$$\sigma = \Pr\left(T > \lambda_t | \mathcal{H}_0\right) = \int_{\lambda_t}^{\infty} f_0(x \mid \mathcal{H}_0) dx$$
$$= \frac{1}{\Gamma\left(\frac{M-t}{2}\right)} \Gamma\left(\frac{m-t}{2}, \frac{\lambda_t}{2}\right). \tag{4.56}$$

The above equation can then be used to obtain the threshold in (4.35). It should be noted that the threshold does not require the knowledge of $\mathbf{a}$. This completes the proof.

4.9.5 Proof of Corollary 4.1

The Lagrangian of the optimization problem in (4.39) is

$$L(\mathbf{a}_s, \lambda) = \mathbf{a}_s^T \mathbf{\Phi}_s \mathbf{a}_s - \lambda \left(\|\mathbf{K}_s \mathbf{a}_s\|_2^2 - \gamma \right) \tag{4.57}$$

$$= \mathbf{a}_s^T (\mathbf{\Phi}_s - \lambda \mathbf{K}_s^T \mathbf{K}_s) \mathbf{a}_s + \lambda\gamma \tag{4.58}$$

If $\mathbf{\Phi}_s - \lambda \mathbf{K}_s^T \mathbf{K}_s$ is positive semidefinite, i.e., $\mathbf{\Phi}_s - \lambda \mathbf{K}_s^T \mathbf{K}_s \succeq 0$, then $L(\mathbf{a}_s, \lambda)$ is convex. It can be minimized by setting $\frac{\partial L(\mathbf{a}, \lambda)}{\mathbf{a}} = 0$, and the solution is

$$\mathbf{\Phi}_s \mathbf{a}_s = \lambda \mathbf{K}_s^T \mathbf{K}_s \mathbf{a}_s \tag{4.59}$$

From (4.59), λ must be a generalized eigenvalue of $(\mathbf{\Phi}_s, \mathbf{K}_s^T \mathbf{K}_s)$. From (4.57) and (4.59), the minimum Lagrangian is $\lambda\gamma$ when $\mathbf{\Phi}_s - \lambda \mathbf{I}_s \succeq 0$.

On the other hand, $\mathbf{\Phi}_s - \lambda \mathbf{I}_s \prec 0$ implies $\inf_{\mathbf{a}_s} L(\mathbf{a}_s, \lambda) = -\infty$.

Thus the dual function of the optimization problem in (4.39) is

$$g(\lambda) = \inf_{\mathbf{a}_s} L(\mathbf{a}_s, \lambda) = \begin{cases} \lambda\gamma, & \text{if } \mathbf{\Phi}_s - \lambda \mathbf{K}_s^T \mathbf{K}_s \succeq 0 \\ -\infty, & \text{if } \mathbf{\Phi}_s - \lambda \mathbf{K}_s^T \mathbf{K}_s \prec 0 \end{cases} \tag{4.60}$$

The dual problem of (4.39) can then be written as

$$\begin{aligned} \max_{\mathbf{a} \in \Omega_s} \ & \lambda\gamma \\ \text{s.t.} \ & \mathbf{\Phi}_s - \lambda \mathbf{K}_s^T \mathbf{K}_s \succeq 0, \end{aligned} \tag{4.61}$$

The maximum λ that satisfies $\mathbf{\Phi}_s - \lambda \mathbf{K}_s^T \mathbf{K}_s \succeq 0$ is $\lambda_{\min}$, which is the minimum generalized eigenvalue of $(\mathbf{\Phi}_s, \mathbf{K}_s^T \mathbf{K}_s)$.

Substituting λ with $\lambda_{\min}$ in (4.59), we can see that the optimum attack vector $\mathbf{a}_s^*$ should be in the form $\mathbf{a}_s^* = c \cdot \mathbf{u}_{\min}$, where $\mathbf{u}_{\min}$ is the generalized eigenvector corresponding to $\lambda_{\min}$, and c is a constant used to ensure that $\|\mathbf{K}_s \mathbf{a}_s^*\|^2 = \gamma$. Solving $\|\mathbf{K}_s c \mathbf{u}_{\min}\|_2^2 = \gamma$ yields (4.40).

Even though the primal problem in (4.39) is non-convex, it can be easily shown that primal and dual problems have zero duality gap. That is

$$(\mathbf{a}_s^*)^T \mathbf{\Phi}_s \mathbf{a}_s^* \geq \inf_{\mathbf{a}_s \in \Omega_s} L(\mathbf{a}_s, \lambda_{\min}) \tag{4.62}$$

$$= (\mathbf{a}_s^*)^T \mathbf{\Phi}_s \mathbf{a}_s^* - \lambda_{\min} \left(\|\mathbf{K}_s \mathbf{a}_s^*\|_2^2 - \gamma \right) \tag{4.63}$$

$$= (\mathbf{a}_s^*)^T \mathbf{\Phi}_s \mathbf{a}_s^* \tag{4.64}$$

where (4.62) is based on the fact that the dual function is a lower bound of the original objective function, (4.63) is true because $L(\mathbf{a}_s, \lambda)$ is convex in $\mathbf{a}_s$ and can be minimized by

$\mathbf{a}_s^*$, and (4.64) is true because $\|\mathbf{K}_s\mathbf{a}_s^*\|_2^2 = \gamma$. Thus the inequality in (4.62) is actually an equality, which means the primal and dual problems can achieve the same objective. As a result, $(\mathbf{a}_s^*, \lambda_{\min})$ are the optimum solutions to the problem. This completes our proof.

4.10 References

[1] Geoff Davis, Stephane Mallat, and Marco Avellaneda. Adaptive greedy approximations. *Constructive approximation*, 13(1):57–98, 1997.

[2] Ronald A DeVore and Vladimir N Temlyakov. Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187, 1996.

[3] Handschin E, Schweppe F C, Kohlas J, and Fiechter A. Bad data analysis for power system state estimation. *IEEE Trans. Power Apparatus and Systems*, 94(2):329–337, Mar 1975.

[4] John R Hershey and Peder A Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 4, pages IV–317, 2007.

[5] Yi Huang, Husheng Li, Kristy A Campbell, and Zhu Han. Defending false data injection attack on smart grid network using adaptive cusum test. In *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, pages 1–6. IEEE, 2011.

[6] Yi Huang, Jin Tang, Yu Cheng, Husheng Li, Kristy A Campbell, and Zhu Han. Real-time detection of false data injection in smart grid networks: an adaptive cusum method and analysis. *IEEE Systems Journal*, 10(2):532–543, 2016.

[7] O. Kosut, L. Jia, R. J. Thomas, and L. Tong. Malicious data attacks on the smart grid. *IEEE Trans. on Smart Grid*, 2(4):645–658, Dec 2011.

[8] Tze Leung Lai. Information bounds and quick detection of parameter changes in stochastic systems. *IEEE Transactions on Information Theory*, 44(7):2917–2929, 1998.

[9] Shang Li, Yasin Yilmaz, and Xiaodong Wang. Quickest detection of false data injection attack in wide-area smart grids. *IEEE Transactions on Smart Grid*, 6(6):2725–2735, 2015.

[10] Jeu-Min Lin and Heng Yau Pan. A static state estimation approach including bad data detection and identification in power systems. In *IEEE Power Eng. Soc. General Meeting*, pages 1–7, June 2007.

[11] Yao Liu, Peng Ning, and Michael K Reiter. False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security*, 14(1):13, 2011.

[12] G Lorden. Procedures for reacting to a change in distribution. *Ann. Math. Statist.*, 42(6):1897–1908, 12 1971.

[13] Patrick McDaniel and Stephen McLaughlin. Security and privacy challenges in the smart grid. *IEEE Security & Privacy*, 7(3), 2009.

[14] ES Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

[15] Yagyensh Chandra Pati, Ramin Rezaiifar, and PS Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.

[16] Henrik Sandberg, André Teixeira, and Karl H Johansson. On security indices for state estimators in power networks. In *First Workshop on Secure Control Systems (SCS), Stockholm, 2010*, 2010.

[17] Albert N Shiryaev. On optimum methods in quickest detection problems. *Theory of Probability & Its Applications*, 8(1):22–46, 1963.

[18] Alexander G Tartakovsky and George V Moustakides. State-of-the-art in bayesian changepoint detection. *Sequential Analysis*, 29(2):125–145, 2010.

[19] Joel A Tropp and Anna C Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Information Theory*, 53(12):4655–4666, 2007.

[20] Dong Wei, Yan Lu, Mohsen Jafari, Paul M Skare, and Kenneth Rohde. Protecting smart grid automation systems against cyberattacks. *IEEE Trans. Smart Grid*, 2(4):782–795, Dec 2011.

[21] Wenhui Xiong, Jin Cao, and Shaoqian Li. Sparse signal recovery with unknown signal sparsity. *EURASIP Journal on Advances in Signal Processing*, 2014(1):178, 2014.

[22] Zong-Han. Yu and Chin Wen-Long. Blind false data injection attack using pca approximation method in smart grid. *IEEE Trans. Smart Grid*, 6(3):1219–1226, May 2015.

[23] R. D Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas. Matpower: Steady-state operations, planning and analysis tools for power systems research and education. In *IEEE Trans. Power Systems*, pages 12–19, Feb 2011.

Chapter 5

Dynamic State Estimation and False Data Detection in Power Systems

5.1 abstract

We study the detection of false data injection in dynamic power grids, where an adversary can maliciously manipulate power grid operations by modifying measurements collected at certain system meters. Due to dynamic system changes such as sudden load changes and changes in network configuration, the state of a power grid changes with time. Dynamic state estimations are used to track this dynamism. Nevertheless, the presence of malicious false data in the measurements may be confused to transients due to these sudden changes and lead to inaccurate state estimations, which can, in turn, have catastrophic impacts on the grid. It is necessary to detect and discriminate these false data from sudden system changes to alleviate their impact on the state estimation. The objective of this chapter is to develop a false data detection algorithm that can effectively detect and identify false data in dynamic power systems. Theoretical analysis and simulation results show that the proposed algorithm can accurately detect and remove false leading to a high state estimation accuracy.

IEEEkeywords

false data injection, dynamic state estimation, dynamic load change, power system .

5.2 Introduction

Smart grid is a combination of power infrastructure, smart meters, and a network of computers [11]. Compared to traditional power grids, smart grid is more robust and efficient owing to the improvement in energy management, control, and system monitoring enabled by the incorporation of networks of computers and smart meters. This, though, comes with a price of grid security and privacy.

Attackers can exploit the cyber-infrastructure of the grid to launch cyber-attacks that can compromise normal grid operations. Some malicious party can launch a cyber-attack by modifying the measurement results obtained by the supervisory control and data acquisition (SCADA) system, such as the power injected into different buses or flowing into the lines between the buses. False data injected in the measurement results will affect the real time control of grid operations, thus cause significant damages to power grids. The cyber-attacks can be performed by breaking into the communication network of the SCADA system, or by remotely accessing the automation devices such as the remote terminal units (RTU) installed at the substations [15].

A large number of methods have been developed to detect various forms of cyber-attacks in smart grids [10, 6, 9, 4, 16, 8, 13]. All of these methods assume a static system model, where the system is in a steady state and its measurements are quasi-static over time. In reality, though, the state of a power system changes with time due to the dynamic nature of system loads [2]. Therefore, state estimation and false data detection methods need a dynamic model to track the time evolution of the system states, which can be used to detect and replace corrupt measurements in the system. A dynamic state estimator can capture

the system transients due to sudden system changes faster and more accurately than its static counterpart does. A dynamic state estimator owes these properties to its capability to use past state estimations to predict the future state of the system one step ahead. The predicted states can be used to initialize the state estimation algorithm during the next step and detect measurements that deviate from these predictions. A mismatch between newly collected measurements and their predicted values indicates that there has been sudden changes in the system such as a loss of a large load and changes in network configurations, or malicious attacks that have modified some system measurements. It is necessary to detect and identify these malicious attacks in order to replace the corrupt measurements before they are processed by the state estimator.

The problem of dynamic state estimation has been studied before in [2, 3, 14, 7, 5]. These works use different versions of an extended Kalman filter (EKF) to perform dynamic state estimation by filtering the predicted state variables. All these algorithms utilize an amplitude test on the innovation vector, difference vector between the newly collected measurements and their predictions, to test the presence of false data and sudden changes in the system. Once the magnitude of the innovation vector exceeds a certain threshold, a flag is raised indicating that there is a sudden change in the system's operating point or false data injection attacks on the system. The false data are discriminated from sudden system changes by analyzing correlated measurements in the region near the abnormality and if the correlated measurements simultaneously fail the detection test, a sudden change is characterized. Otherwise, the suspected measurements contain false data and they are replaced with their predictions. This method of discriminating attacks from sudden change in the system operating point, however, may not be effective if the attacks are simultaneously injected

in the correlated measurements. This may lead to a mischaracterization of the attacks as sudden changes, and therefore, fail to remove and prevent the corrupt measurements from entering the state estimation stage. In this chapter, we propose a new detection algorithm that can accurately detect the bad data and discriminate them from sudden changes in the system. Based on the statistical distribution of the innovation vector, a hypothesis test is developed to study the system behavior with and without false data injections. From the hypothesis test, a chi-square test is then designed to detect the attacks. Once the false data are detected, corrupt measurements are identified and replaced with their predictions and then forwarded to the state estimator. Theoretical analysis and simulation results show that the newly proposed detection algorithm can effectively detect and replace false data injections including those injected in correlated measurements.

The remainder of this chapter is organized as follows. The system model and problem formulation are described in Section 5.3. In Section 5.4, we explain the dynamic state estimation algorithm in details. The problem of false data detection is studied in Section 5.5, where we present the design procedures of our proposed detection algorithm. Simulation results are given in Section 5.6, and Section 5.7 concludes this chapter.

5.3 Mathematical Model

We consider a power system with N buses. Each bus is equipped with a meter measuring the active and reactive power injections. The line connecting two buses is equipped with two sensors, one at each end, measuring the active and reactive power flows. Without loss of generality, assume that the first bus is the reference.

Define the set of buses connected to bus i as $\mathcal{X}_i$ with cardinality $c_i = |\mathcal{X}_i|$. Denote the

active and reactive power injections into bus i as P_i and Q_i, respectively. Similarly, the active and reactive power flows from bus i to bus j are denoted P_{ij} and Q_{ij}, respectively, $\forall\ j \in \mathcal{X}_i$. The SCADA system provides a total of $m = m_1 + m_2 + 1$ measurements, where $m_1 = 2N$ is the number of active and reactive power injections, $m_2 = \sum_{i=1}^{N} |\mathcal{X}_i|$ is the number of active and reactive power flows. In addition to the power measurements, the measurement of the voltage magnitude at the reference bus is also available. Define the power measurement vector as $\mathbf{z} = [z_1, z_2, \ldots, z_m]^T \in \mathcal{R}^{m\times 1}$, where $(\cdot)^T$ is the matrix transpose operator and $\mathcal{R}$ is the set of real numbers.

In phase measurement, one of the N buses will serve as a reference, and we only need to measure or estimate the phases of the remaining $N-1$ buses relative to that of the reference bus. Define the state vector as $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T \in \mathcal{R}^{n\times 1}$ for $n = 2N - 1$, where the first $N - 1$ elements of $\mathbf{x}$ are the voltage angles of $N - 1$ non-reference buses and the last N elements are the voltage magnitudes of N buses.

The relationship between the measurement vector $\mathbf{z}_k$ and the state vector $\mathbf{x}_k$, at an instant of time k can be expressed as

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{e}_k, \tag{5.1}$$

where $\mathbf{e}_k \in \mathcal{R}^{m\times 1}$ is the measurement error vector at the sampling instant k, and $\mathbf{h}(\mathbf{x}_k) = [h_1(\mathbf{x}_k), \ldots, h_m(\mathbf{x}_k)]^T$ is a function of voltage magnitudes and phase angles. As in [2], we assume that the measurement noise $\mathbf{e}_k$ is zero-mean Gaussian with covariance matrix $\mathbf{R}_k$.

Based on the observations in (5.1), the state estimator can obtain an estimate $\hat{\mathbf{x}}_k$ of the state variable $\mathbf{x}_k$, such that the error $\mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_k)$ between the estimated and the actual

measurements is minimized.

5.4 Dynamic State Estimation

In this section, we present the dynamic state estimation, which relies on previous estimates to predict the future state of the system. The predicted states can, in turn, be used by the system operator for timely anomaly detection and other control decisions such as economic dispatch and other related functions.

Consider the following state transition model, which describes the time behavior of the state vector, as

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k + \mathbf{w}_k, \tag{5.2}$$

where $\mathbf{F}_k \in \mathcal{R}^{n \times n}$ is a non-zero diagonal matrix, $\mathbf{G}_k \in \mathcal{R}^{n \times 1}$ is a non-zero column vector, and $\mathbf{w}_k \in \mathcal{R}^{n \times 1}$ is a white Gaussian noise with zero mean and covariance matrix $\mathbf{Q}_k$.

The parameters $\mathbf{F}_k$ and $\mathbf{G}_k$ can be identified according to the Holt's exponential smoothing methods [2] for forecasting by smoothing an original series with two smoothing parameters, α and β, with values between 0 and 1. Denote the predicted state vector at time k as $\tilde{\mathbf{x}}_k$ and the predicted state vector at time $k+1$ as $\tilde{\mathbf{x}}_{k+1}$. The Holt's method is expressed as

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{a}_k + \mathbf{b}_k, \tag{5.3}$$

where

$$\mathbf{a}_k = \alpha \mathbf{x}_k + (1 - \alpha)\tilde{\mathbf{x}}_k$$

and

$$\mathbf{b}_k = \beta \left[\mathbf{a}_k - \mathbf{a}_{k-1}\right] + (1 - \beta)\mathbf{b}_{k-1}.$$

Rewrite (5.3) as

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k, \tag{5.4}$$

where $\mathbf{F}_k = \alpha(1 + \beta)\mathbf{I}_n$ and $\mathbf{G}_k = (1 + \beta)(1 - \alpha)\tilde{\mathbf{x}}_k - \beta \mathbf{a}_{k-1} + (1 + \beta)\mathbf{b}_{k-1}$. By adding a zero mean Gaussian noise $\mathbf{w}_k$ with covariance matrix $\mathbf{Q}_k$ to (5.4) to account for model uncertainties yields the equation in (5.2).

5.4.1 System State Forecasting

The main advantage of the dynamic state estimator that sets it apart from the static state estimator is its ability to use the past state estimates to predict the future system states. Let $\hat{\mathbf{x}}_k$ be the estimated state vector at time k and $\mathbf{\Sigma}_k$ its error covariance matrix. The predicted state vector $\tilde{\mathbf{x}}_{k+1}$ and its error covariance matrix $\mathbf{M}_{k+1}$ at time k can be obtained by performing the conditional expectation on (5.2) as follows

$$\tilde{\mathbf{x}}_{k+1} = \mathbb{E}\left[\mathbf{x}_{k+1} \mid \mathbf{x}_k = \hat{\mathbf{x}}_k\right] = \mathbf{F}_k \hat{\mathbf{x}}_k + \mathbf{G}_k \tag{5.5}$$

$$\mathbf{M}_{k+1} = \mathbb{E}\left[\left(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}\right)\left(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}\right)^T \mid \mathbf{x}_k = \hat{\mathbf{x}}_k\right]$$
$$= \mathbf{F}_k \mathbf{\Sigma}_k \mathbf{F}_k + \mathbf{Q}_k, \tag{5.6}$$

where $\mathbb{E}[.]$ is the expectation operator.

5.4.2 System State Estimation

The state estimation, also known as state filtering, seeks to filter the predicated state vector $\tilde{\mathbf{x}}_{k+1}$, obtained at the preceding step k, by using the newly received measurement vector $\mathbf{z}_{k+1}$ at time $k+1$. During this stage, a new estimate $\hat{\mathbf{x}}_{k+1}$ along with its error covariance matrix $\mathbf{\Sigma}_{k+1}$ are obtained at time $k+1$ by minimizing the objective function

$$J\left(\mathbf{x}_{k+1}\right) = \left[\mathbf{z}_{k+1} - \mathbf{h}(\mathbf{x}_{k+1})\right] \mathbf{R}_{k+1}^{-1} \left[\mathbf{z}_{k+1} - \mathbf{h}(\mathbf{x}_{k+1})\right]^T$$
$$+ \left[\left(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}\right) \mathbf{M}_{k+1}^{-1} \left(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}\right)^T\right]. \tag{5.7}$$

The estimate $\hat{\mathbf{x}}_{k+1}$ that minimizes the objective function (5.7) can be obtained through an iterative Extended Kalman Filter (EKF) [2] as

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \mathbf{\Sigma}^{(i)}\{\mathbf{H}^T(\mathbf{x}^{(i)})\mathbf{R}^{-1}[\mathbf{z} - \mathbf{h}(\mathbf{x}^{(i)})]$$
$$-\mathbf{M}^{-1}[\mathbf{x}^{(i)} - \tilde{\mathbf{x}}]\}, \tag{5.8}$$

where i denotes the iteration counter and

$$\boldsymbol{\Sigma}^{(i)} = \left[\mathbf{H}^T(\mathbf{x}^{(i)})\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{(i)}) + \mathbf{M}^{-1}\right]^{-1}. \tag{5.9}$$

It should be noted that the subscript $k+1$ was omitted in (5.8) and (5.9) for simplicity.

One of the main benefits of the state forecasting stage is that it provides the initial states to the iterative EKF algorithm in (5.8). Thus, the convergence of the EKF algorithm partly depends on the accuracy of the forecast state vector. A High state forecasting accuracy leads to a faster convergence of the EKF algorithm.

Let the initial guess $\mathbf{x}^{(i)} = \tilde{\mathbf{x}}$ at time $k+1$ and by perform only one iteration in (5.8), the estimated state vector is approximated as

$$\hat{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_{k+1} + \mathbf{K}_{k+1}\mathbf{v}_{k+1}, \tag{5.10}$$

where

$$\mathbf{v}_{k+1} = \mathbf{z}_{k+1} - \mathbf{h}(\tilde{\mathbf{x}}_{k+1}) \tag{5.11}$$

is the innovation vector,

$$\mathbf{K}_{k+1} = \boldsymbol{\Sigma}_{k+1}\mathbf{H}^T(\tilde{\mathbf{x}}_{k+1})\mathbf{R}_{k+1}^{-1}$$

is the gain matrix, and

$$\mathbf{\Sigma}_{k+1} = \left[\mathbf{H}^T(\tilde{\mathbf{x}}_{k+1})\mathbf{R}_{k+1}^{-1}\mathbf{H}(\tilde{\mathbf{x}}_{k+1}) + \mathbf{M}_{k+1}^{-1}\right]^{-1}.$$

5.5 False Data Detection and Identification

The problem of detecting and identifying false data injections in the measurement vector is studied in this section. Based on (5.10), the estimated state vector $\hat{\mathbf{x}}_{k+1}$ is a function the innovation vector $\mathbf{v}_{k+1}$, the difference between newly received measurements at time $k+1$ and its corresponding predictions $\mathbf{h}(\tilde{\mathbf{x}}_{k+1}$. The newly received measurement vector $\mathbf{z}_{k+1}$ may deviate from its predicated value $\mathbf{h}(\tilde{\mathbf{x}}_{k+1})$. This mismatch between the measured and predicted measurements may be a result of two factors: a sudden change in the system's operating point due to a loss of a large load [3] and false data injections in the measurements. The change in system's operating point is considered a normal event. However the presence of false data injections is abnormal and can be harmful to the system. It is important to detect and remove any false data injections from the measurements $\mathbf{z}_{k+1}$ before performing the state estimation.

5.5.1 False Data Detection

Unusual events, sudden system changes and false data, in the measurements can be detected by means of statistical analysis on the innovation or residual vector $\mathbf{v}_{k+1}$ in (5.11) and its covariance matrix $\mathbf{S}_{k+1}$ written as

$$\mathbf{S}_{k+1} = \mathbf{H}^T(\tilde{\mathbf{x}}_{k+1})\mathbf{M}_{k+1}\mathbf{H}(\tilde{\mathbf{x}}_{k+1}) + \mathbf{R}_{k+1}. \tag{5.12}$$

The proof for (5.12) is given in Appendix 5.8.2.

A frequently used bad data detector [2] applies a threshold test to the magnitude of each component $\mathbf{v}_{k+1}(i)$ of the innovation vector as follows.

$$|\mathbf{v}_{k+1}(i)| \leq \gamma \sigma_{S_i}, \tag{5.13}$$

where $|\mathbf{v}_{k+1}(i)|$ is the absolute value of the i-th element of $\mathbf{v}_{k+1}$, σ_{S_i} is the standard deviation of the i-th element of $\mathbf{v}_{k+1}$, and γ defines the limit of confidence (usually equal to 3 for Gaussian variables). To account for approximation errors due to linearization during the calculation of (5.12), γ is set slightly higher than 3.

If the component $\mathbf{v}_{k+1}(i)$ of the innovation vector does not satisfy the test (5.13) then the i-th element of measurement vector $\mathbf{z}_{k+1}$ carries bad data or has experienced sudden changes in the system's operating point. To discriminate bad data from sudden changes, the correlated or adjacent measurements to the suspect measurement are checked. If they fail the test (5.13) then the suspect measurement is a sudden change in the system. Otherwise, the suspect measurement has bad data and it is removed by setting $\mathbf{z}_{k+1}(i)$ equal to the i-th component of the predicted measurement vector $\mathbf{h}(\tilde{\mathbf{x}}_{k+1})$.

It should be noted that the detector (5.13) cannot distinguish bad data from sudden changes if the correlated measurements of a certain region simultaneously experience bad data. To this end, we propose a detector that can detect false data injections including those injected into correlated measurements.

5.5.2 Proposed Bad Data Detector

Define the null hypothesis $\mathcal{H}_0$, which corresponds to the measurements without bad data at time $k+1$, and the alternative hypothesis $\mathcal{H}_1$, which corresponds to the measurements with bad data at time $k+1$, as

$$\begin{aligned} \mathcal{H}_0 &: \mathbf{z}_{k+1} = \mathbf{h}(\mathbf{x}_{k+1}) + \mathbf{e}_{k+1} \\ \mathcal{H}_1 &: \mathbf{z}_{k+1} = \mathbf{h}(\mathbf{x}_{k+1}) + \mathbf{e}_{k+1} + \mathbf{a}, \end{aligned} \tag{5.14}$$

where $\mathbf{a}$ is a vector of bad data injected in the measurements.

From (5.11), (5.14), and (5.26), the hypothesis test on the innovation vector $\mathbf{v}_{k+1}$ can be written as

$$\begin{aligned} \mathcal{H}_0 &: \mathbf{v}_{k+1} = \mathbf{H}(\tilde{\mathbf{x}}_{k+1})(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}) + \mathbf{e}_{k+1} \\ \mathcal{H}_1 &: \mathbf{v}_{k+1} = \mathbf{H}(\tilde{\mathbf{x}}_{k+1})(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}) + \mathbf{e}_{k+1} + \mathbf{a}. \end{aligned} \tag{5.15}$$

The innnovation vector $\mathbf{v}_{k+1}$ under the null hypothesis $\mathcal{H}_0$ is known to be a zero mean Gaussian vector [2] and [12] with covariance matrix $\mathbf{S}_{k+1}$ in (5.12). Assuming that $\mathbf{a}$ is a deterministic vector, under $\mathcal{H}_1$, $\mathbf{v}_{k+1}$ is Gaussian with mean $\mathbf{a}$ and covariance matrix $\mathbf{S}_{k+1}$.

Based on the hypotheses in (5.15), we have the following theorem regarding the bad data detection.

Theorem 5.1: Given a whitening matrix $\mathbf{W}_{k+1}$ at time $k+1$ and a whitened innovation

vector $\bar{\mathbf{v}}_{k+1} = \mathbf{W}_{k+1}\mathbf{v}_{k+1}$, an attack vector $\mathbf{a}$ in the measurements $\mathbf{z}_{k+1}$ is detected if

$$\sum_{i=1}^{m} \bar{\mathbf{v}}_{k+1}^2(i) > \tau, \tag{5.16}$$

where τ is the threshold, $\mathbf{W}_{k+1} = \mathbf{D}^{-\frac{1}{2}}\mathbf{U}$, $\mathbf{D}$ is a diagonal matrix with the eigenvalues of $\mathbf{S}_{k+1}$ on its main diagonal, $\mathbf{U}$ is the corresponding orthonormal eigenvector matrix, that is, $\mathbf{S}_{k+1} = \mathbf{U}^T\mathbf{D}\mathbf{U}$.

The proof is shown in Appendix 5.8.3.

Owing to the normal distribution of $\bar{\mathbf{v}}_{k+1}(i)$, as shown in (5.28), the test $\sum_{i=1}^{m} \bar{\mathbf{v}}_{k+1}^2(i)$ is theoretically a chi-square test and the threshold τ can be theoretically calculated. In practice, however, $\bar{\mathbf{v}}_{k+1}(i)$ is not distributed normally with accuracy. This is because $\bar{\mathbf{v}}_{k+1} = \mathbf{W}_{k+1}\mathbf{v}_{k+1}$ is a linear transformation of $\mathbf{v}_{k+1}$ whose covariance matrix $\mathbf{S}_{k+1}$ is only an approximation of its true value [12]. As a result, the threshold τ is calculated offline.

At time $k+1$, the test (5.16) is applied and the presence of bad data or an attack vector $\mathbf{a}$ in the measurements is declared if the test is greater than the threshold τ. Otherwise, there is no bad data in the measurements. In case the bad data are detected, the test (5.13) is used to identify affected components of the measurement vector $\mathbf{z}_{k+1}$, which are replaced with their corresponding components of the predicted measurement vector $\mathbf{h}(\tilde{\mathbf{x}}_{k+1})$.

5.6 Simulation Results

In this section, we present the simulation results performed on a 13-bus two area system shown in Fig. 5.1. Bus 1 is used as the reference bus. The measurement vector is composed of $m = 55$ components: the voltage magnitude of bus 1, the active and reactive power

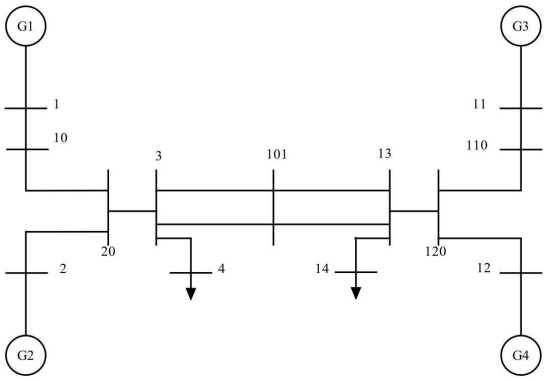

Figure 5.1: Single Line Diagram Two Area System

injections at all 13 buses, the active and reactive power flows at all 14 lines. The state vector is composed of $n = 25$ components: the voltage magnitudes at all 13 buses and the phase angles at the non reference buses. The system dynamic is simulated by increasing the active load at bus 4 by 0.5 per unit (p.u) and the resulting measurement and state vectors are considered the true values of the system. The noisy measurement vector (5.1) is then obtained by adding a zero mean Gaussian noise to each of the true measurements and covariance bfR_k. The noise variances, diagonal elements of bfR_k, are 10^{-5} and 10^{-6} for the voltage magnitude of bus 1 and the active and reactive power measurements, respectively. The matrix $\mathbf{Q}_k = 10^{-6}\mathbf{I}_n$ is kept constant at every sampling time k. The parameters $\mathbf{F}_k$ and $\mathbf{G}_k$ are obtained according to the Holt's exponential smoothing method [2] with $\alpha = 0.95$ and $\beta = 0.001$. For the detector in (5.13) $\gamma = 3.5$.

To test the performance of the proposed detector, two scenarios are simulated: bad data and sudden load change conditions. The bad data condition is simulated by injecting errors of -1.5 and 1 p.u into the active power measurements at buses 3 and 4, respectively, during a time period $20 \leq k \leq 60$ unless specified otherwise. The sudden load change condition is simulated by cutting the active power injection of bus 4 by 1 p.u. In each figure, every point

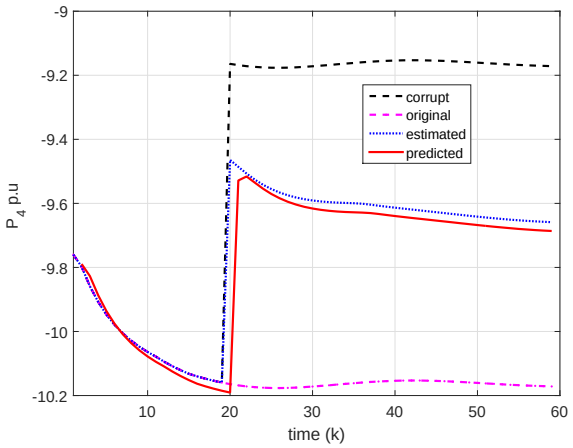

Figure 5.2: The real power at bus 4 vs time k with bad data at $k \geq 20$ and the detector in (5.13)

on the curves is obtained by averaging over 1000 observations.

Fig. 5.2 shows the active power at bus 4 with bad data injected into the active power measurements at buses 3 and 4. The detector (5.13) is used to detect and remove the bad data. As expected the detector (5.13) is unable to distinguish the bad data from the sudden changes in the system because the bad data are injected into the correlated measurements of adjacent buses 3 and 4. Hence, no measures are taken to remove these bad data. Consequently, the performance of the estimator deteriorates an estimation error of as high as 0.64 p.u at $k = 25$.

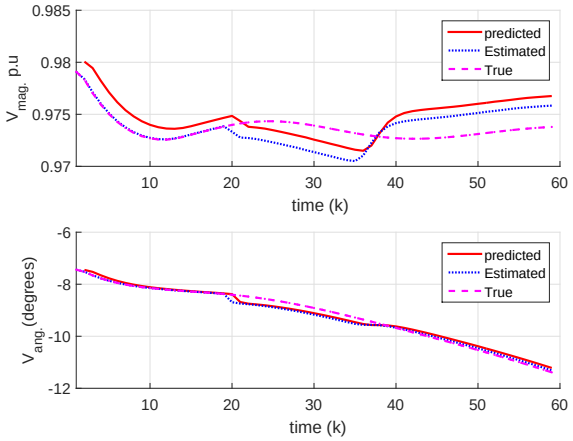

Figure 5.3: The voltage magnitude (top) and phase angle (bottom) at bus 3 vs time k with bad data at $k \geq 20$ and the detector in (5.13).

Fig. 5.3 shows the voltage magnitude (top) and phase angle (bottom) at bus 3 with bad data injected into the active power measurements at buses 3 and 4. As in Fig. 5.2 the detector (5.13) fails to detect the bad data. As a result, the state estimation is inaccurate at $k \geq 20$.

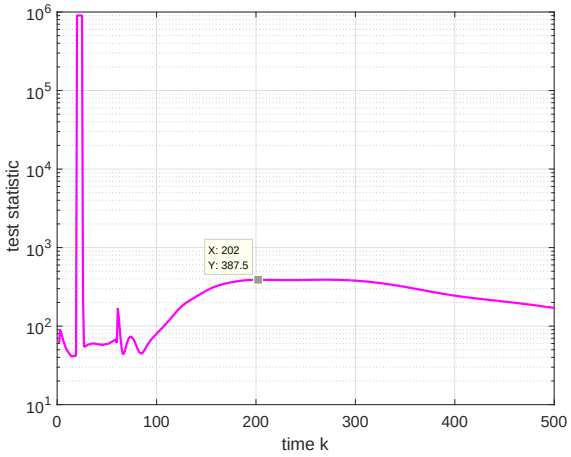

Figure 5.4: The test statistic (5.16) vs time k with bad data at $20 \leq k \leq 25$ and a sudden load change at $k = 60$.

Fig. 5.4 illustrates how the proposed test statistic (5.16) changes over time. Different scenarios are simulated to test our test statistic. During the time period $20 \leq k \leq 25$ errors (bad data) of -1.5 and 1 p.u are injected into the active power measurements at buses 3 and 4, respectively. During the time period $k \geq 60$, a sudden change is triggered by cutting the active power at bus 4 by 1 p.u. As depicted in the figure, the value of the test statistic drastically increases during the bad data injection period, $20 \leq k \leq 25$, to 10^6 while its maximum value is only 387.5 during the period, $k \geq 60$, of the sudden change. Therefore, with a carefully selected threshold τ the proposed detector (5.16) can effectively distinguish bad data from sudden changes with high accuracy.

Fig. 5.5 and Fig. 5.6 show the effectiveness of our proposed detector in removing bad data. The proposed detector (5.16) with a threshold $\tau = 500$ is used to detect the bad

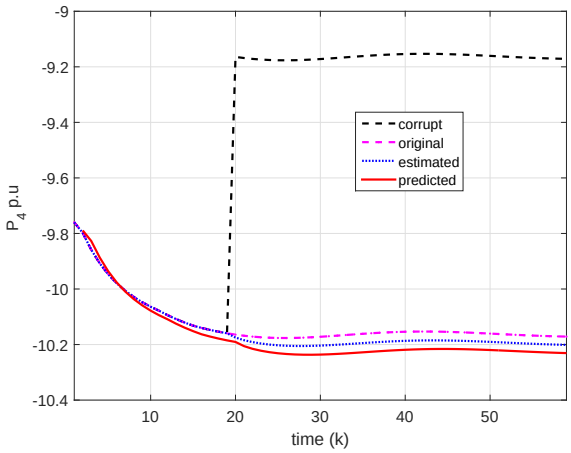

Figure 5.5: The real power at bus 4 vs time k with bad data at $k \geq 20$ and the detector in (5.16).

data while the detector (5.13) is used to identify the affected measurements to be replaced. As for Fig. 5.2, the errors (bad data) of -1.5 and 1 p.u are injected into the active power measurements at buses 3 and 4, respectively, during a time period $20 \leq k \leq 60$. In Fig. 5.5, the proposed detector detects and replaces the bad data in the active power measurement at bus 4. As a result, the active power at bus 4 is accurately estimated with an estimation error of as low as 0.03 p.u. Similarly, Fig. 5.6 shows the voltage magnitude (top) and phase angle (bottom) at bus 3 with a high state estimation accuracy after the bad data are removed.

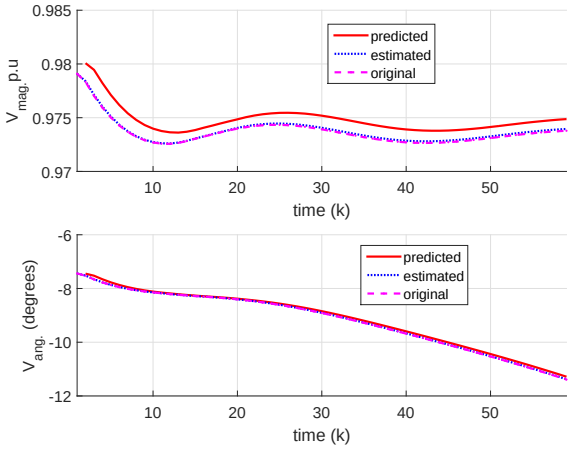

Figure 5.6: The voltage magnitude (top) and phase angle (bottom) at bus 3 vs time k with bad data at $k \geq 20$ and the detector in (5.16).

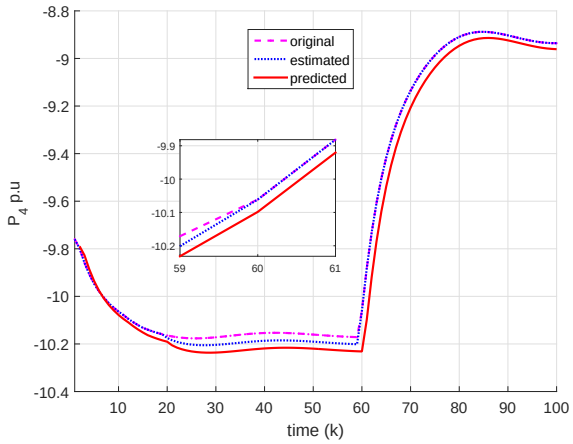

Figure 5.7: The real power at bus 4 vs time k with a load change at $k = 60$.

Fig. 5.7 and Fig. 5.8 show the performance of our proposed detector during a sudden load change. The proposed detector (5.16) with a threshold $\tau = 500$ is used. The sudden load change condition is simulated by cutting the active power injection of bus 4 by 1 p.u. As shown in Fig. 5.7, the proposed detector does not detect the sudden load changes and thus no measurements are replaced as we expected. As a result, the active power at bus 4 is accurately estimated. It should be noted that the load change at bus 4 does not abruptly change the active power. This change happens rather gradually during the time period $60 \leq k \leq 85$ and this allows the predictor to adapt to the new changes in the load with gradually decreasing prediction error. For example, the predicted value for the active power of bus 4 at time $k = 60$ is -10.23 p.u (predicted at time $k = 59$) while the true value is -10.06 p.u. which is equivalent to a 0.17 prediction error. However the prediction error becomes as low as 0.025 at $k = 85$ when the predictor is fully adapted to the load changes. Similarly, in Fig. 5.8 the voltage magnitude (top) and phase angle (bottom) at bus 3 are estimated with a high state estimation accuracy and an adaptive predictor with low prediction error. This demonstrates the capability of a dynamic state estimator to adapt to dynamic load changes.

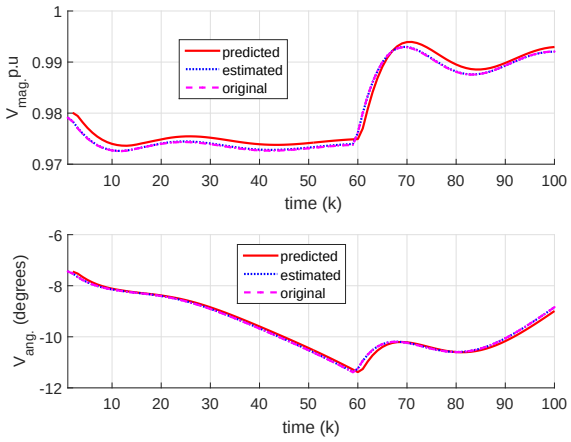

Figure 5.8: The voltage magnitude (top) and phase angle (bottom) at bus 3 vs time k with a load change at $k = 60$.

5.7 Conclusion

The problem of detection and removal of false data injected into smart grids have been studied in this chapter. By applying a hypothesis test on measurement innovation vector, we have developed a detection algorithm that accurately detected and removed false data before they were processed by the state estimator. Unlike conventional algorithms that rely on measurement correlation to discriminate false data from sudden changes in the system, the designed algorithm can detect any false data including those injected into correlated measurements. Simulation results have shown that the proposed algorithm, detected and removed attacks and thus enhanced the state estimation accuracy.

5.8 Appendix

5.8.1 Proof of (5.8)

The point $\mathbf{x}$, which minimizes (5.7) can be obtained by calculating the first derivative of $J(\mathbf{x})$ and setting it to zero. Define the first derivative of $J(\mathbf{x})$ as

$$\mathbf{g}(\mathbf{x}) = \frac{\partial \mathbf{J}(\mathbf{x})}{\partial \mathbf{x}} = -\frac{\partial \mathbf{h^T}(\mathbf{x})}{\partial \mathbf{x}} \mathbf{R^{-1}} \left[\mathbf{z} - \mathbf{h}(\mathbf{x})\right] + \mathbf{M^{-1}} \left(\mathbf{x} - \tilde{\mathbf{x}}\right). \tag{5.17}$$

The minimum point $\hat{\mathbf{x}}$ of $J(\mathbf{x})$ is calculated by solving

$$\mathbf{g}(\hat{\mathbf{x}}) = \mathbf{0}. \tag{5.18}$$

Given the non-linearity of (5.17), (5.18) is solved by iterative methods such as the Newton-Raphson method [1].

The Taylor series expansion of $\mathbf{g}(\mathbf{x})$ for $\mathbf{x} = \mathbf{x}^{(0)} + \Delta_{\mathbf{x}}$ is

$$\mathbf{g}(\mathbf{x}) = \mathbf{g}(\mathbf{x^{(0)}}) + \frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}} \mid_{\mathbf{x}=\mathbf{x}^{(0)}} \Delta_{\mathbf{x}}, \tag{5.19}$$

where $\mathbf{x}^{(0)}$ is the initial point and

$$\frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{g}^{'}(\mathbf{x}) = \left[\mathbf{H}^T(\mathbf{x})\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}) + \mathbf{M}^{-1}\right]^{-1}. \tag{5.20}$$

According to the Newton-Raphson method [1], by setting (5.19) to zero, the increment

$\Delta_{\mathbf{x}}$ is obtained as

$$\Delta_{\mathbf{x}} = -\left[\mathbf{g}^{'}(\mathbf{x}^{(0)})\right]^{-1}\mathbf{g}(\mathbf{x}^{(\mathbf{0})}). \tag{5.21}$$

Thus

$$\mathbf{x} = \mathbf{x}^{(0)} - \mathbf{\Sigma}^{(0)}\mathbf{g}(\mathbf{x}^{(\mathbf{0})}), \tag{5.22}$$

where

$$\mathbf{\Sigma}^{(0)} = \left[\mathbf{g}^{'}(\mathbf{x}^{(0)})\right]^{-1} = \left[\mathbf{H}^{T}(\mathbf{x}^{(0)})\mathbf{R}^{-1}\mathbf{H}(\mathbf{x}^{(0)}) + \mathbf{M}^{-1}\right]^{-1}. \tag{5.23}$$

By combining (5.17), (5.22), and (5.23) at the $(i+1)$-th iteration with an initial point $\mathbf{x}^{(i)} = \mathbf{x}^{(i+1)} - \Delta_{\mathbf{x}}$, the $i+1$-th point becomes

$$\begin{aligned}\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \mathbf{\Sigma}^{(i)}\{&\mathbf{H}^{T}(\mathbf{x}^{(i)})\mathbf{R}^{-1}[\mathbf{z} + \mathbf{h}(\mathbf{x}^{(i)})] \\ &-\mathbf{M}^{-1}(\mathbf{x}^{(i)} - \tilde{\mathbf{x}})\},\end{aligned} \tag{5.24}$$

where $\mathbf{H}(\mathbf{x}) = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}$. And this completes the proof.

5.8.2 Proof of (5.12)

Write the Taylor series expansion of $\mathbf{h}(\mathbf{x}_{k+1})$ around a linearization point $\tilde{\mathbf{x}}_{k+1}$ as

$$\mathbf{h}(\mathbf{x}_{k+1}) = \mathbf{h}(\tilde{\mathbf{x}}_{k+1}) + \mathbf{H}(\tilde{\mathbf{x}}_{k+1})(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}), \tag{5.25}$$

where $\mathbf{H}(\tilde{\mathbf{x}}_{k+1}) = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}|_{\mathbf{x}=\tilde{\mathbf{x}}_{k+1}}$.

The higher order terms of (5.25) are omitted by assumption that the difference $(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1})$ is very small.

Combining (5.1), (5.11), and (5.25) gives

$$\mathbf{v}_{k+1} = \mathbf{H}(\tilde{\mathbf{x}}_{k+1})(\mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}) + \mathbf{e}_{k+1}. \tag{5.26}$$

Following (5.26), the covariance $\mathbf{S}_{k+1}$ is given by

$$\begin{aligned}
\mathbf{S}_{k+1} &= \mathbb{E}\left[\mathbf{v}_{k+1}\mathbf{v}_{k+1}^T \mid \mathbf{x}_k = \hat{\mathbf{x}}_k\right] \\
&= \mathbb{E}\left[\mathbf{H}(\tilde{\mathbf{x}}_{k+1})\Delta_{\mathbf{x}}\Delta_{\mathbf{x}}^T\mathbf{H}^T(\tilde{\mathbf{x}}_{k+1}) \mid \mathbf{x}_k = \hat{\mathbf{x}}_k\right] + \mathbb{E}\left[\mathbf{e}_{k+1}\mathbf{e}_{k+1}^T\right] \\
&= \mathbf{H}(\tilde{\mathbf{x}}_{k+1})\mathbb{E}\left[\Delta_{\mathbf{x}}\Delta_{\mathbf{x}}^T \mid \mathbf{x}_k = \hat{\mathbf{x}}_k\right]\mathbf{H}^T(\tilde{\mathbf{x}}_{k+1}) + \mathbf{R}_{k+1} \\
&= \mathbf{H}(\tilde{\mathbf{x}}_{k+1})\mathbf{M}_{k+1}\mathbf{H}^T(\tilde{\mathbf{x}}_{k+1}) + \mathbf{R}_{k+1},
\end{aligned} \tag{5.27}$$

where $\Delta_{\mathbf{x}} = \mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}$. This completes our proof.

5.8.3 Proof of Theorem (5.1)

Write the covariance matrix of the innovation vector as $\mathbf{S}_{k+1} = \mathbf{U}^T\mathbf{D}\mathbf{U}$, where $\mathbf{D}$ is a diagonal matrix with the eigenvalues of $\mathbf{S}_{k+1}$ on its main diagonal and $\mathbf{U}$ is the corresponding orthonormal eigenvector matrix. The whitening transformation of the innovation vector $\mathbf{v}_{k+1}$ is $\bar{\mathbf{v}}_{k+1} = \mathbf{W}_{k+1}\mathbf{v}_{k+1}$, where $\mathbf{W}_{k+1} = \mathbf{D}^{-\frac{1}{2}}\mathbf{U}$.

Following the Gaussian distribution of $\mathbf{v}_{k+1}$ given in (5.15), the hypothesis test on $\bar{\mathbf{v}}_{k+1}$ is

$$\begin{aligned} \mathcal{H}_0 : \bar{\mathbf{v}}_{k+1} = \mathbf{W}_{k+1}\mathbf{v}_{k+1} \quad &\sim \mathcal{N}(\mathbf{0}, \mathbf{I_m}) \\ \mathcal{H}_1 : \bar{\mathbf{v}}_{k+1} = \mathbf{W}_{k+1}\mathbf{v}_{k+1} \quad &\sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{I}_m), \end{aligned} \tag{5.28}$$

where $\mathbf{I}_m$ is a $m \times m$ identity matrix and $\boldsymbol{\mu} = \mathbf{W}_{k+1}\mathbf{a}$.

The LLR for the hypothesis test in (5.28) is

$$\begin{aligned} \log L(\bar{\mathbf{v}}_{k+1}, \mathbf{a}) &= \log \frac{\Pr(\bar{\mathbf{v}}_{k+1}|\mathbf{a}, \mathcal{H}_1)}{\Pr(\bar{\mathbf{v}}_{k+1}|\mathcal{H}_0)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau \\ &= 2\bar{\mathbf{v}}_{k+1}^T\mathbf{W}_{k+1}\mathbf{a} - \mathbf{a}^T\mathbf{W}_{k+1}^T\mathbf{W}_{k+1}\mathbf{a} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau. \end{aligned} \tag{5.29}$$

Since $\mathbf{a}$ is unknown, we can perform GLRT, where $\mathbf{a}$ can be estimated by maximizing the LLR (5.29). Setting $\partial\frac{\log L(\bar{\mathbf{v}}_{k+1}, \mathbf{a})}{\partial \mathbf{a}} = 0$ yields

$$\bar{\mathbf{a}} = \mathbf{W}_{k+1}^{-1}\bar{\mathbf{v}}_{k+1}, \tag{5.30}$$

where $\bar{\mathbf{a}}$ is the maximum likelihood (ML) estimate of $\mathbf{a}$.

Replacing $\mathbf{a}$ in (5.29) with $\bar{\mathbf{a}}$ in (5.30) results in

$$\begin{aligned}\log L(\bar{\mathbf{v}}_{k+1}, \mathbf{a}) &= \bar{\mathbf{v}}_{k+1}^T \bar{\mathbf{v}}_{k+1} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau \\ &= \sum_{i=1}^{m} \bar{\mathbf{v}}_{k+1}^2(i) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau. \end{aligned} \tag{5.31}$$

This completes our proof.

5.9 References

[1] Saba Akram and Qurrat ul Ann. Newton raphson method. *International Journal of Scientific & Engineering Research*, 6(7), 2015.

[2] AM Leite Da Silva, MB Do Coutto Filho, and JF De Queiroz. State forecasting in electric power systems. In *IEE Proceedings C (Generation, Transmission and Distribution)*, volume 130, pages 237–244. IET, 1983.

[3] Milton Brown Do Coutto Filho and Julio Cesar Stacchini de Souza. Forecasting-aided state estimationpart i: Panorama. *IEEE Transactions on Power Systems*, 24(4):1667–1677, 2009.

[4] Handschin E, Schweppe F C, Kohlas J, and Fiechter A. Bad data analysis for power system state estimation. *IEEE Trans. Power Apparatus and Systems*, 94(2):329–337, Mar 1975.

[5] Amit Jain and NR Shivakumar. Power system tracking and dynamic state estimation. In *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*, pages 1–8. IEEE, 2009.

[6] O. Kosut, L. Jia, R. J. Thomas, and L. Tong. Malicious data attacks on the smart grid. *IEEE Trans. on Smart Grid*, 2(4):645–658, Dec 2011.

[7] Hong Li and Weiguo Li. Estimation and forecasting of dynamic state estimation in power systems. In *Sustainable Power Generation and Supply, 2009. SUPERGEN'09. International Conference on*, pages 1–6. IEEE, 2009.

[8] Jingwen Liang, Oliver Kosut, and Lalitha Sankar. Cyber attacks on ac state estimation: Unobservability and physical consequences. In *PES General Meeting— Conference & Exposition, 2014 IEEE*, pages 1–5. IEEE, 2014.

[9] Jeu-Min Lin and Heng Yau Pan. A static state estimation approach including bad data detection and identification in power systems. In *IEEE Power Eng. Soc. General Meeting*, pages 1–7, June 2007.

[10] Yao Liu, Peng Ning, and Michael K Reiter. False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security*, 14(1):13, 2011.

[11] Patrick McDaniel and Stephen McLaughlin. Security and privacy challenges in the smart grid. *IEEE Security & Privacy*, 7(3), 2009.

[12] K Nishiya, J Hasegawa, and T Koike. Dynamic state estimation including anomaly detection and identification for power systems. In *IEE Proceedings C (Generation, Transmission and Distribution)*, volume 129, pages 192–198. IET, 1982.

[13] Md Ashfaqur Rahman and Hamed Mohsenian-Rad. False data injection attacks against nonlinear state estimation in smart power grids. In *Power and Energy Society General Meeting (PES), 2013 IEEE*, pages 1–5. IEEE, 2013.

[14] Kuang-Rong Shih and Shyh-Jier Huang. Application of a robust algorithm for dynamic state estimation of a power system. *IEEE Transactions on Power Systems*, 17(1):141–147, 2002.

[15] Dong Wei, Yan Lu, Mohsen Jafari, Paul M Skare, and Kenneth Rohde. Protecting smart grid automation systems against cyberattacks. *IEEE Trans. Smart Grid*, 2(4):782–795, Dec 2011.

[16] Zong-Han. Yu and Chin Wen-Long. Blind false data injection attack using pca approximation method in smart grid. *IEEE Trans. Smart Grid*, 6(3):1219–1226, May 2015.

Chapter 6

Conclusions

This chapter briefly explains the main findings of this dissertation and the improvements made over other research projects that focus on intrusion detection and state estimation in power grids. To close the chapter, a list of some possible directions for the future research is provided.

6.1 Contributions

This dissertation designed algorithms to enhance the robustness of power grid state estimation system by improving its intrusion detection capabilities and reducing its state estimation errors. The designs were presented in four main chapters, namely: optimum PMU placement for power system state estimation, optimum PMU placement for bad data detection in power systems, low latency detection of sparse false data injections in smart grids, and dynamic state estimation and false data detection in power systems. The main findings of each of the mentioned chapters are summarized as follows.

First, optimum PMU placement algorithms for power system state estimation were designed. The objective of these algorithms was to find the best PMU locations that maximized the state estimation accuracy. The design metric was the MSE (mean squared error) of the state variables. Two low complexity algorithms were developed to balance the encountered tradeoff between performance and complexity. The simulation results showed that the MSE

performance of the designed low complexity algorithms approached that of the optimum but less computationally efficient algorithm based on exhaustive search. All proposed algorithms achieved significant gains over common algorithms designed based on the concept of critical measurements.

Second, optimum PMU placement algorithms for malicious attack or intrusion detection in power systems were designed. The objective of these algorithms was to find the best PMU locations that maximize the probability of detecting the attacks. Similarly to the preceding chapter, the algorithms tackled the encountered tradeoff between performance and complexity. The simulation results showed that the developed algorithm can outperform the common algorithms, based on the concept of critical measurements, in terms of the attack detection probability.

Third, quickest detection of false data injected to smart grids was studied in this dissertation. we developed a new OMP-CUSUM algorithm for low latency detection of false data injections can efficiently identify the meters under attack, and minimize the detection delay of bad data injections. furthermore, an optimum attack vector was developed to test and validate the performance of the proposed algorithm. The simulation results indicated that the designed OMP-CUSUM algorithm accurately and reliably detected intrusions with short delays. Particularly, the designed OMP-CUSUM algorithm had considerably lower complexities as compared to its conventional counterparts based on exhaustive search.

Fourth, dynamic state estimation and false data detection in power systems was studied in this dissertation. A detector was designed to detect and remove false data in dynamic systems, where false data can be confused with infrequent but normal sudden changes in the systems. Unlike conventional algorithms, which cannot detect false data injected into

correlated measurements, the designed detector accurately detected and removed attacks on the system including those injected into correlated measurements.

6.2 Future Work

The presented low latency intrusion detection algorithms uses a linear static model for the system measurements. Such a model assumes that the system is operating in the steady state and states and power measurements are quasi-static over time. Therefore, the state-power measurement relationship is approximately linear.

A nonlinear dynamic model such as that presented in Chapter 6 is practically more realistic, albeit its complexity. For the future work, we intend to further extend our low latency intrusion detection research to nonlinear dynamic models.

Appendix

Vita

Israel Akingeneye received his B.S. degree in Electrical Engineering from University of Arkansas, Fayetteville, USA in May, 2013. He then started his direct PhD program in Electrical Engineering at University of Arkansas, Fayetteville, USA in August, 2013. Israel is currently a PhD candidate in the Department of Electrical Engineering, University of Arkansas, Fayetteville, USA. He is a student member of IEEE. His research interests include Wireless Communications, Wireless Sensor Networks, Statistical Signal Processing, and Cyber Resilience of Smart Grids.