AUTOMATIC DETECTION OF DATA MANIPULATION

IN POWER SYSTEMS

by

Leen Al Halabi

A Thesis Presented to the Faculty of the
American University of Sharjah
College of Engineering
in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science in
Mechatronics Engineering

Sharjah, United Arab Emirates

April 2019

Approval Signatures

We, the undersigned, approve the Master's Thesis of Leen Al Halabi.

Thesis Title: Automatic detection of data manipulation in power systems.

Signature **Date of Signature**
(dd/mm/yyyy)

Dr. Shayok Mukhopadhyay
Assistant Professor, Department of Electrical Engineering
Thesis Advisor

Dr. Ahmad Osman
Associate Professor, Department of Electrical Engineering
Thesis Co-Advisor

Dr. Mostafa Farouk Shaaban
Professor, Department of Electrical Engineering
Thesis Committee Member

Dr. Lotfi Romdhane
Professor, Department of Mechanical Engineering
Thesis Committee Member

Dr. Mohammad Jaradat
Director, Mechatronics Engineering Graduate Program

Dr. Lotfi Romdhane
Associate Dean for Graduate Affairs and Research
College of Engineering

Dr. Naif Darwish
Acting Dean, College of Engineering

Dr. Mohamed El-Tarhuni
Vice Provost for Graduate Studies

Acknowledgement

I would like to thank my advisors Dr. Shayok Mukhopadhyay and Dr. Ahmad Osman for providing knowledge, guidance, support, and motivation throughout my research stages. I’m deeply beholden for their great assistance, worthy discussion and suggestions. Truly, I’m grateful for their attention to details and vivid clarifications which inspired me to work harder to fulfill the requirements of this thesis. Their dedication in reviewing the work based on regular meetings, discussions, and provocative questions has greatly influenced the meaningful research of the thesis.

I would like to thank the professors of the Mechatronics Engineering program who taught me the master level courses. I really appreciate their advice and motivation.

Further, I would like to extend my gratitude to all those individuals who encouraged me for completing this work. Indeed, the successful completion of this work would not have been possible without their enthusiasm, pledge, inspiration and continuous support.

Many thanks are delivered to my parents, brother and fiancé who encouraged me and supported me throughout this journey. Indeed, a word of appreciation is not enough but I hope that it pays some of my recognition to the people behind my success, my family.

Dedication

To my family for their vital support.

To my professors for their encouragement and never-ending guidance.

Abstract

Securing and protecting a power grid system is critical because a power grid transmits and distributes power to millions of people across a country. One of the significant topics in this field is having a real time model that monitors and controls power system grids. A robust monitoring system can be built based on State Estimation (SE) techniques especially when dealing with non-linear structures such as power systems. In order to have a secured power network, the data acquired by the Supervisory Control And Data Acquisition (SCADA) systems has to be reliable and consistent. This is achieved by enforcing false data detection methods where a malicious interference can be detected. Such methods can differentiate between rubbish data and an intrusion attacking the network. Neural Networks (NNs) are considered one of the widely used techniques in detecting false data injections. This work develops a strategy for automatic detection of data manipulation in a power system network. The main contribution is to introduce a Neural Network (NN) based system that can detect any data manipulation whether bypassed by the state estimators or not. The model is capable of detecting the intrusion with a minimum of three random meters in a grid being manipulated. This provides power system operators the ability to take the required decision before a large-scale attack can occur.

Keywords: ***Power grid, Neural Network, State Estimation, False Data Injection, SCADA.***

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ANN	Artificial Neural Network
FDI	False Data Injection
NN	Neural Network
NR	Newton Raphson
SE	State Estimation
WLS	Weighted Least Squares

Chapter 1. Introduction

In this chapter, a short introduction about the power and energy management systems and the problems encountered in this field is provided. Then, the specific problem investigated in this study as well as the thesis contribution is presented. Finally, general organization of the thesis is illustrated.

1.1. Overview

Energy management is deliberated as the fastest developing segment in the power industry and as one of the biggest engineering success stories of the past decade. An essential goal behind this development is to monitor, coordinate and control the process of generation, transmission and distribution of electrical energy. Consequently, an optimized, reliable and accurate power network can be achieved [1]. Commonly, these functions are done by the control centre which is divided into three subsystems as described in Figure 1.1.

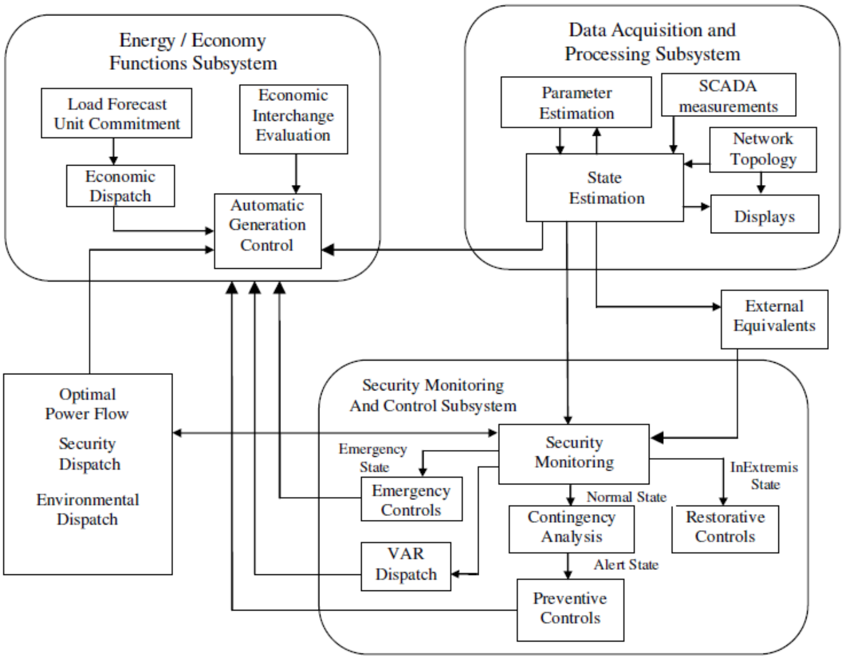

Figure 1.1: Functional diagram of modern energy management system [1].

To achieve the above mentioned objective, a Supervisory Control And Data Acquisition system (SCADA) has been employed in power networks since it can provide the required real time data at a reasonable complexity and cost. It offers the operator the ability to anticipate, study and control the optimal response against measured conditions. In fact, obtaining a clear picture of the states is obtained through State Estimation (SE) which is considered as the backbone of the Energy Management System (EMS). This technique is widely utilized in most power systems to estimate the state variables (bus voltages and phase angles). Basically, such technique is introduced to replace the need of having hardware monitoring devices at each bus. The state estimator is able to filter the raw data and determine the values of state variables to allow the operators to take decisions aimed at maintaining the security of the power grid. One of the commonly used algorithms in SE is Weighted Least Squares (WLS) [2]. The overall objective of the SE is to reduce the errors as well as improve the efficiency of the power system.

Furthermore, there are various types of measurement errors in power grids. These types are divided into random errors; which happens due to lack of precision of the measurement equipment, gross errors; which arise from outside interference, topological errors; which are due to configuration errors [1]. SE is only capable of filtering the data from random errors. There are other types of errors, such as an attack or data manipulation, which may affect the state estimator from providing an accurate estimate of the states. Hence, a bad data detection and identification step, is of vital importance in every SE procedure.

Neural Networks (NNs) have been implemented widely in false data detection when dealing with security assessment of power systems. It is a multilayer based solution that consists of input layers followed by hidden layers which are connected to output layers. In other words, NNs can be described as a huge number of neurons interconnected with each other to resolve a certain problem. NNs can be trained by feeding them with different scenarios and examples with the corresponding outputs. A NN is based on pattern recognition, where it can detect and identify patterns inherent in both linear and non-linear systems [3, 4, 5].

Recently, certain attacks have been developed in a way to bypass the conventional SE based bad data detection techniques employed in a power network [5, 6]. Hence, in this thesis, a NN is designed to detect if a False Data Injection (FDI) attack has occurred, even if the state estimator bypassed such data. This will ensure a continuous, reliable and smooth operation of the power network [6, 7].

1.2. Thesis Objectives

Detect any data manipulation through the designed NN whether such manipulated data is passed by the state estimator or not and hence, enhance the quality of the estimated data being monitored by the operators of the power systems.

1.3. Research Contribution

Each part of a mechatronic system needs focus to develop a better overall system architecture. Therefore, this work focuses on the decision-making aspects of a mechatronic system. Specifically, a NN based decision making system is developed for a power grid, which not only involves software-based monitoring but also acting upon the decision taken. However, due to the subject at hand involving power grids, it is not possible to carry out actual full-scale tests. Therefore, this work uses simulation results developed on a standard IEEE-30 bus power system model. The main contributions of this work are as follows:

- A NN based-system is developed to detect a certain type of an intrusion namely a random FDI attack in power grids.
- Detect an FDI attack even if only three random meters are attacked, and the magnitude of meter readings are manipulated by a minimum of 4% of their actual value.
- Verify the above detection scheme works successfully under different load scenarios.

1.4. Thesis Organization

The rest of the thesis is organized as follows: Chapter 2 provides background about power systems architecture and recent techniques used for SE. Moreover, works related to this research are discussed. The employed methods and algorithms are discussed in Chapter 3 along with the implementation of the preliminary architecture. Chapter 4 provides some preliminary simulation results based on the proposed work in the previous chapter. Finally, Chapter 5 concludes the work and outlines future work efforts.

Chapter 2. Background and Literature Review

In this chapter, the basic principles of the power network are discussed. Basically, the chapter is organized as follows; first it highlights the recent work achieved in regards of power grids, SE and NNs. Later, it provides an introduction on each of the previously mentioned topics that emphasizes their significance in providing a reliable power system.

2.1. Literature Review

The authors in [8], have proposed the idea of considering both the physical and system theory aspects when investigating the vulnerabilities of smart grids. In July 2010, a malware called “Stuxnet” was found targeting vulnerable SCADA systems [9]. Stuxnet reflects a clear example of a cyber-attack used to induce physical consequences. Authors have illustrated that the proposed approach is capable of modelling the malicious behaviours as either components’ failures, external inputs, or noises. In addition, it is found that utilizing such algorithm will help in analysing the attacks’ effects on the system, and hence, designing several detection algorithms or counter measures against attacks.

In recent years, SE [10, 11] theory paved the way to exploit the time varying structure of the system in its estimation. It was originally invented by Adrien-Marine Legendre and Carl Friedrich Gauss in 1805-1809. Fred Schweppe in [11] modeled the SE problem as the basis selection problem. Thus, the estimated state vector can be obtained by using well-known algorithms in the field of power electricity. Shweppe introduced how to estimate the state vector using the WLS algorithm [11].

In power grids, maintaining synchronism between each power sub-systems is one of the major factors that helps in increasing the reliability of the power system in meeting the future demand growth. The security assessment of any power system comprises vulnerabilities resulting in voltage insecurity, static insecurity and dynamic insecurity [12]. The instability can be as a form of an increase in the angular swings of some generators which can lead to a loss of synchronism with other generators [13]. Further, Artificial Neural Network (ANN) has been utilized widely in the field of security that helps estimating the steady state, transient state and dynamic stability/security status. This is due to its ability in solving non-linear pattern recognition problems.

In [14], a divergence-based feature selection algorithm has been introduced in combination with the ANN for online security monitoring of the power systems. The ANN works as a classifier in which it categorizes the operating data state into secure and insecure. This is achieved by passing the data into feature selection step. The feature selection step includes a pre-disturbance real and reactive power generation data of each generator. The main aim of this stage is to eliminate the insensitive features in prior to avoid exhaustive search of second stage. In the second stage, classes are classified based on the concept of divergence which is a measure of dissimilarity between the classes.

In [15], authors proposed an error correction method based on ANN which can detect the erroneous measurements and replace it with the corresponding correct values. False measurements were added to the estimated state vectors and fed into the ANN. Results show that the proportion of divergent SE operations is somehow reduced as well as the mean square error of the estimated system state.

In [16], a NN model has been proposed which is capable of detecting random FDI attack over IEEE-14, 30 and 300 bus systems. In fact, the system parameters were simulated using MATPOWER tool box. Results show that the proposed algorithm can detect the occurrence of an attack if four or more meters have been attacked.

2.2. Background

An overview of the investigated techniques is illustrated in this section. First, power systems are discussed in which power flow problem is stated followed by load flow solution. Later, the SE is presented in which WLS algorithm is utilized. Finally, the NN basics are explained as well as the algorithm used in building the NN.

2.2.1. Power systems. This section describes the architecture of power systems and introduces the most common method in solving the power flow problem. In addition, it highlights the importance of the power flow calculation in monitoring the power grid.

2.2.1.1. ***Power flow.*** The power flow problem is the computation of voltage magnitude and phase angle at each bus in a power system under balanced three-phase steady-state conditions. There are several methods used to solve the power flow problem such as Gauss-Seidel, Newton Raphson (NR), Fast-decoupled power flow [17, 18]. The starting point of a power flow problem is a single line diagram of the power system where input data can be extracted. The input data comprises bus data, transmission line data, and transformer data. Each bus is associated with four variables defined as below [18, 19].

- Voltage magnitude V_k.
- Phase angle δ_k .
- Net real power P_k.
- Reactive power Q_k supplied to the bus.

At each bus, two of the above variables are always known (as an input) and the others can be calculated by the power flow solution as shown in Figure 2.1. The power injection at each bus k is the difference between the power generated and the load power which is defined as below.

$$P_k = P_{Gk} - P_{Lk} \tag{1}$$

$$Q_k = Q_{Gk} - Q_{Lk} \tag{2}$$

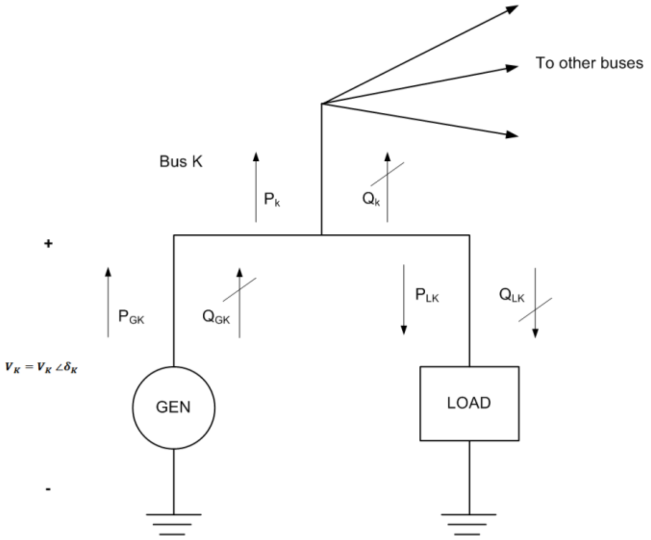

Figure 2.1: Bus variables V_k, d_k, P_k [19].

Furthermore, there are three different types of buses named as below:

- Swing bus (Slack bus): In each power system, there is only one bus defined as Slack bus (usually numbered as bus-1) which is considered as a reference bus characterized by an input of voltage of 1.0 per unit and zero phase angle. Thus, power flow calculates the P and Q for this bus.
- Load (PQ) bus: In this bus, P_k and Q_k are considered as inputs and hence, power flow computes V_k and δ_k.
- Voltage controlled bus (PV) bus: The inputs to this bus are P_k and V_k and consequently Q_k and δ_k are found by power flow. This bus is also known as Generator Bus.

The above types of buses are summarized in Table 2.1.

Table 2.1: Types of buses and its variables.

Type of bus	**Known variables**	**Unknown variables**
Load Bus (P- Q Bus)	P, Q	V, δ
Generator Bus (P - V Bus)	P, V	Q, δ
Slack Bus (V- δ Bus)	V, δ	P, Q

Moreover, the input data to each transmission line is represented as a π circuit series impedance $\bar{Z}'$ and shunt admittance Y'; the two buses to which the line is connected. In this thesis, all vectors are denoted with a bar above the variable. The bus admittance matrix Y is formulated from the nodal equations defined as below [18, 19].

$$\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & \dots & Y_{1N} \\ Y_{21} & Y_{22} & Y_{23} & \dots & Y_{2N} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{N1} & Y_{N2} & Y_{N3} & \dots & Y_{NN} \end{bmatrix} \begin{bmatrix} V_{10} \\ V_{20} \\ \vdots \\ V_{N0} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_N \end{bmatrix} \quad (3)$$

The above can be re-written as follows

$$Y\bar{V} = \bar{I} \quad (4)$$

Where Y is the bus admittance matrix, $\bar{V}$ is the N-bus voltages and $\bar{I}$ is the column vector of N-current sources. The Y-bus is molded as below:

- Diagonal elements: Y_{kk}= sum of admittances connected to bus k (k= 1, 2, 3, …,N).
- Off-diagonal elements: Y_{kn}= - (sum of admittances connected between buses k and n) ($k \neq n$).

The significance of writing nodal equations is to generate the admittance matrix easily and solve accordingly for the bus voltage vector when currents are known [19].

Further, in this thesis, IEEE-30 bus system is considered where six generators are distributed among the buses, one slack bus referenced as the first bus and the rest of the buses considered as load buses. The architecture of the above system is shown in Figure 2.2.

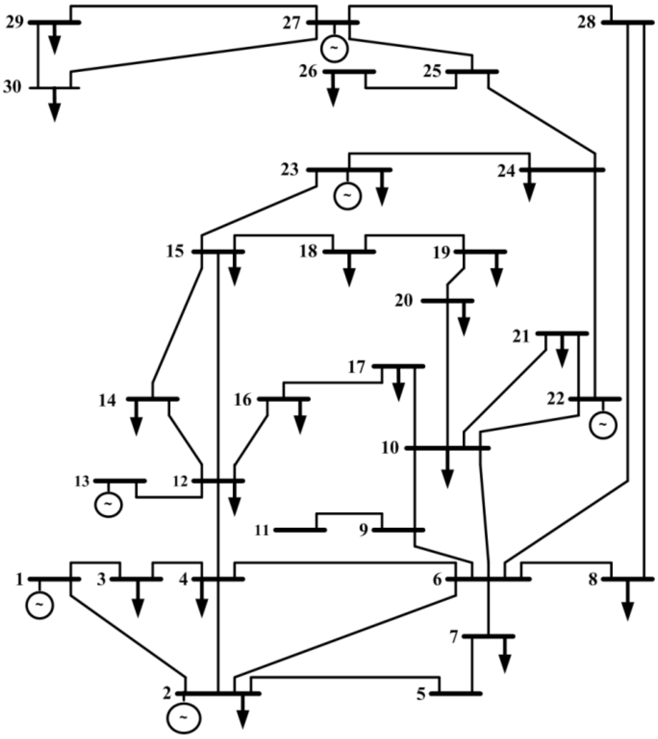

Figure 2.2: IEEE 30-bus system [19].

2.2.1.2. ***Newton Raphson (NR) method.*** This method is an iterative algorithm that solves a set of non-linear equations with an equal number of unknowns [20, 21]. It is widely used for solving load flow problems due to its ability to obtain the state variables (voltages and phase angles) of all the buses. The power equations are defined as following:

$$P_k(x) = V_k \sum_{n=1}^{N} Y_{kn} V_n \cos(\delta_k - \delta_n - \theta_{kn}) \tag{5}$$

$$Q_k(x) = V_k \sum_{n=1}^{N} Y_{kn} V_n \sin(\delta_k - \delta_n - \theta_{kn}) \tag{6}$$

Where k= 2, 3, ….., N. The first slack bus (k=1) is excluded since it is known. However, after the formation of the Y-bus, the Jacobian matrix is computed as following

$$J_{11} = \begin{bmatrix} L_{22} & \cdots & L_{2x} \\ \vdots & \ddots & \vdots \\ L_{x2} & \cdots & L_{xx} \end{bmatrix} \tag{7}$$

Where

$L_{ik} = \frac{\partial P_i}{\partial \delta_k}$, $L_{ii} = \frac{\partial P_i}{\partial \delta_i}$

$$J_{21} = \begin{bmatrix} M_{22} & \cdots & M_{2x} \\ \vdots & \ddots & \vdots \\ M_{x2} & \cdots & M_{xx} \end{bmatrix} \tag{8}$$

Where

$M_{ik} = \frac{\partial Q_i}{\partial \delta_k}$, $M_{ii} = \frac{\partial Q_i}{\partial \delta_i}$

$$J_{12} = \begin{bmatrix} N_{22} & \cdots & N_{2x} \\ \vdots & \ddots & \vdots \\ N_{x2} & \cdots & N_{xx} \end{bmatrix} \tag{9}$$

Where

$N_{ik} = |V_k| \frac{\partial P_i}{\partial |V_k|}$, $N_{ii} = |V_i| \frac{\partial P_i}{\partial |V_i|}$

$$J_{22} = \begin{bmatrix} Z_{22} & \cdots & Z_{2x} \\ \vdots & \ddots & \vdots \\ Z_{x2} & \cdots & Z_{xx} \end{bmatrix} \tag{10}$$

Where

$Z_{ik} = |V_i| \frac{\partial Q_i}{\partial |V_k|}$, $Z_{ii} = |V_i| \frac{\partial Q_i}{\partial |V_k|}$

The above equations are solved iteratively and formed into one vector that can be used later in the SE equation [22].

2.2.2. State Estimation (SE) overview. Yet, with the late integration of low carbon technology in power grids, an enhanced observability of the networks has become crucial. As a result, SE started receiving a significant attention by distribution system operators in performing security analysis and managing the networks. Indeed, SE is considered a key energy management system function that enables achieving a reliable operation of the power system. Its responsibility lies in providing a database of the real time state of the system which helps other energy management functions [22, 23].

The state of an electric power system holds both the bus voltage magnitudes and their respective phase angles. SE is an algorithm that utilizes statistical techniques to estimate the actual values of the unknown state variables. This was first introduced by Schweppe in [3], [5] at the initials of 1970's. Many methods have been proposed to compute the state vector of the power system in the most efficient approach due to its high computational complexity. Obtaining this state vector includes collecting the real time measurements data such as line flows, power injection values, and voltage measurements. This was done through SCADA systems and proposed SE algorithms. Nevertheless, SE is divided into two types [24, 25]:

- **Static SE**. The state vector in this type obtains the instant values of the measurement set and is repeated regularly at suitable time intervals. This sort of algorithm is commonly used in "quasi-static" power systems where power changes slowly but steadily [26, 27, 28].
- **Dynamic SE**. Such type is frequently used in dynamic power systems where power changes due to the change of driven loads. The power generation varies as the load changes which in turn affects the power flow and injections across the whole system. Thus, the use of dynamic SE arises to satisfy the need of monitoring and controlling the power grid at short intervals unlike static SE. Dynamic state estimators are capable of computing the next state vector at instant (t+k) knowing the current state vector at instant (t). This has been considered of prominent importance in power systems due to the ability in performing security analysis one step ahead and hence, allowing the operators more time to take the required action [26, 27, 28].

WLS method is considered one of the common techniques that SE uses and was invented since the 19th century. The basis of this technique is to minimize the sum of the squares of the error between the estimated and actual value of the state variable [28].

2.2.2.1. ***Weighted Least Squares (WLS) algorithm.*** The WLS method is the optimal technique to solve an over-determined system. An over-determined system is when the number of equations is more than the number of unknowns. The over-determined system can be best approximated through the least squares method. The least square method is a way to estimate the unknown parameters in an equation by minimizing the sum of the squared errors (deviation) between the data and the model." Weighted" means that a certain number is assigned to each result based on the reliability of the data [29].

2.2.2.2. ***Weighted Least Squares (WLS) mathematical modeling***. The data of the model is received as a vector $\bar{z}$, from the SCADA system, and it is of size m that can be written as:

$$\bar{z}^{meas} = \begin{bmatrix} z_1 \\ \vdots \\ z_m \end{bmatrix} \tag{11}$$

Then, a state vector $\bar{x}$ is considered and since the power system is a complex network then the SE will be a non-linear function denoted by $\bar{h}(x)$ as follows

$$h(x) = \begin{bmatrix} h_1(x_1 \ \dots \ x_n) \\ \vdots \\ h_m(x_1 \ \dots \ x_n) \end{bmatrix} \tag{12}$$

Where n is the total number of the system state variables. There is a true measurement values for the state variables. Thus, the calculated estimate from the non-linear function $h(x)$ will deviate from the actual value and this deviation is called error ($\bar{e}$). The error vector is represented in the below equation and one can note that the size of $\bar{e}$ is of the same size as $\bar{z}$. Also, it is worth mentioning that each error is independent of the other errors, with zero mean and independent covariance [30, 31].

$$\bar{e} = \begin{bmatrix} e_1 \\ \vdots \\ e_m \end{bmatrix} \tag{13}$$

Therefore, the true measurement $\bar{z}$ can be best represented by the following equation:

$$\bar{z}^{meas} = h(x) + \bar{e} \tag{14}$$

Where the $\bar{z}$ equation represents the state equation after derivation. And the error is added to compensate for the difference between the actual and calculated values [30]. In order to solve the SE problem using the WLS SE technique, an objective function should be minimized. In other words, the jacobian matrix shall be minimized as given below.

$$H(x) = min_x J(x) = \sum_{i=1}^{m} \frac{(z_i - h_i(x))^2}{R_{ii}} \tag{15}$$

Where:

x : the system state variables.

$J(x)$: the measurement residual function.

z_i : i^th measurement.

m : total number of measurements.

R : is called “the covariance matrix of measurement errors” [30].

As indicated from the above equation, $J(x)$ is a summation of the squares of the measurement errors, in addition to a weighted matrix known as the covariance matrix as described below.

$$Cov\ (\bar{e}) = R = \begin{bmatrix} R_{11} & 0 & 0 & 0 \\ 0 & R_{22} & \dots & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & R_{mm} \end{bmatrix} = \begin{bmatrix} {\sigma_1}^2 & 0 & \dots & 0 \\ 0 & {\sigma_2}^2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & 0 & {\sigma_{mm}}^2 \end{bmatrix} \tag{16}$$

Where ${\sigma_i}^2$ is the i^{th} measurement variance and m is the number of measurements. The objective function can be minimized by taking the partial derivatives of $J(x)$ with respect to the state variables x. Hence, $\frac{\partial J\, x}{\partial x}$ is denoted as $g(x)$ which is a common term in most power systems. Similarly, the measurement function is derived with respect to the state variable and is denoted by $H(x)$ as shown below [32, 33].

$$H = \begin{bmatrix} \frac{\partial P_{ii}}{\partial \theta} & \frac{\partial P_{ii}}{\partial V} \\ \frac{\partial P_{ij}}{\partial \theta} & \frac{\partial P_{ij}}{\partial V} \\ \frac{\partial Q_{ii}}{\partial \theta} & \frac{\partial Q_{ii}}{\partial V} \\ \frac{\partial Q_{ij}}{\partial \theta} & \frac{\partial Q_{ij}}{\partial V} \\ \frac{\partial V_{mag}}{\partial \theta} & \frac{\partial V_{mag}}{\partial V} \end{bmatrix} \tag{17}$$

Where:

$\frac{\partial P_{ii}}{\partial \theta}$: Real power injection measurements at bus *i* with respect to the phase angles.

$\frac{\partial P_{ii}}{\partial V}$: Real power injection measurements at bus *i* with respect to the voltage magnitudes.

$\frac{\partial Q_{ii}}{\partial \theta}$: Reactive power injection measurements at bus *i* with respect to the phase angles.

$\frac{\partial Q_{ii}}{\partial V}$: Reactive power injection measurements at bus *i* with respect to the voltage magnitudes.

$\frac{\partial P_{ij}}{\partial \theta}$: Real power flow measurements from bus *i* to bus *j* with respect to the phase angles.

$\frac{\partial P_{ij}}{\partial V}$: Real power flow measurements from bus *i* to bus *j* with respect to the voltage magnitudes.

$\frac{\partial Q_{ij}}{\partial \theta}$: Reactive power flow measurements from bus *i* to bus *j* with respect to the phase angles.

$\frac{\partial Q_{ij}}{\partial V}$: Reactive power flow measurements from bus *i* to bus *j* with respect to the voltage magnitudes.

$\frac{\partial I_{mag}}{\partial \theta}$: Current magnitude measurements from bus *i* to bus *j* with respect to the phase angles.

$\frac{\partial I_{mag}}{\partial V}$: Current magnitude measurements from bus *i* to bus *j* with respect to the voltage magnitudes.

$\frac{\partial V_{mag}}{\partial V}$: Voltage magnitude measurements with respect to their corresponding voltage magnitude at bus i or bus j. Such that:

$\frac{\partial V_i}{\partial V_i} = 1, \frac{\partial V_i}{\partial V_j} = 0, \frac{\partial V_i}{\partial \theta_i} = 0,$ and $\frac{\partial V_i}{\partial \theta_j} = 0$

$H(x)$ is called the measurement Jacobian matrix, which is equal to $\frac{\partial J(x)}{\partial x}$ [33]. The objective function in terms of $g(x)$ and $H(x)$ is as follow:

$$g\,(x) = \frac{\partial J(x)}{\partial x} = -\frac{\partial h(x)}{\partial x}^{T} R^{-1}\left(z - h\,(x)\right) = -H(x)^{T} R^{-1}\left(z - h\,(x)\right) = 0 \qquad (18)$$

Where T is the transpose of the matrix. It is important to highlight that computing $H(x)$ is one of the most significant steps when solving the WLS algorithm. In fact, most of the data acquired from the SCADA systems are integrated in a non-linear form. As a result, calculating the Jacobian matrix is more complicated compared with linear systems [34]. Furthermore, as per Taylor series expansion in [33], the higher order of the above equation can be neglected and hence, the equation can be iteratively resolved by NR method. Below equation represents the iterative solution for the state variable, x:

$$x^{k+1} = x^k + [H(x^k)^T R^{-1} H(x^k)]^{-1} H(x^k)^T R^{-1} (z - h\,(x^k)) \tag{19}$$

$$x^{est} = [H(x)^T R^{-1} H(x)]^{-1} H(x)^T R^{-1} (z - h(\,x)) \tag{20}$$

The matrix $G(x_k)$ is sparse, definite and positive noting that the system is fully observable [33]. Given that $G(x_k)$ is sparse, the inverse of it is a full matrix which is not likely to have on the right side according to the below equation

$$x^{k+1} = x^k + G(x^k)^{-1}\, g(x^k) \tag{21}$$

Where:

$G(x^k)$: $H(x^k)^T R^{-1} H(x^k)$ = known as the gain matrix.

$g(x^k)$: $H(x^k)^T R^{-1}(\,z - h\,(x^k))$ = the non-linear function.

k : iteration number.

(x^k) : solution vector for the kth iteration.

Consequently, some adjustments and manipulations are done in which x_k is moved to the left hand side of the equation, which yields the difference between the two consecutive iterations, $\Delta x = (x_{k+1}) - (x_k)$. Later, all of the equation is multiplied by $G(x_k)$ to discard it from the right side. The new equation will become as following

$$G(x^k)(\Delta\, x) = g(x^k) \tag{22}$$

$$G(x^k)(\Delta\, x) = H(x^k)^T R^{-1} (z - h\,(x^k)) \tag{23}$$

At the start of the process, k is set to zero and the initial value of the state variable x is set to 1.0 per unit for the voltage and zero for the phase angle. After that, $g(x_k)$ and $G(x_k)$ are computed to find the next state variable x_1. This procedure is repeated until the difference between $x(k+1)$ and $x(k)$ is below a defined threshold ε [35, 36].

2.2.3. Neural Network (NN) overview. NN has been used extensively in complex systems to extract certain patterns and detect trends that cannot be observed by humans or other computers. The basis of this technique relies on feeding your designed network with information to train it and enhance its performance. One of the main advantages of NN is its Adaptive learning; the ability of doing certain tasks based on the given data for training. It is also important to highlight that a new representation of the information fed during the training will be done by the NN [37, 38].

In comparison to conventional computers, NN utilizes an algorithmic approach in solving the problems. It processes the information in a similar way to the human brain. The network is defined as a huge number of highly interconnected processing elements (called neurons) working in parallel to resolve a particular problem. NN is trained by examples and not a set of instructions [39, 40, 41]. The selection of the examples must be done carefully so the NN can function successfully.

2.2.3.1. *Network layers.* The NN consists of three layers; inputs layer which is connected to hidden layers connected to outputs layer as shown in Figure 2.3.

Each layer consists of several units represented as mentioned earlier "neurons". The input layer represents the raw data fed into the network for training. Each unit in the hidden layer is responsible to take an action based on the action taken by the input units, as well as the weights of the corresponding connections between the input and hidden neurons. Similarly, the output units depend on hidden units' activities and the connections between the hidden and output neurons [42, 43].

As mentioned earlier, the NN can build its own representation of the input data which makes it an attractive topic to most of the developers. However, a hidden unit is active based on the weights between the input and hidden units. Thus, any modification to the weights change the representation of the hidden unit accordingly [44, 45, 46].

The network's response in pattern recognition can be classified into two general paradigms; associative mapping and regularity detection [47]. In the first paradigm, a particular pattern is learnt and produced by the network on the input units when a new pattern is applied on the set of input neurons.

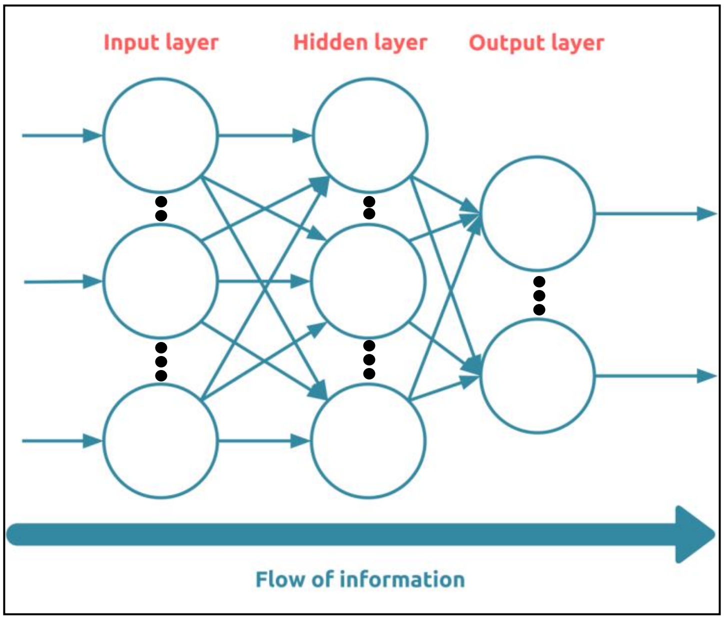

Figure 2.3: NN layers architecture.

Generally, the associative mapping uses different techniques such as auto- association and hetro-association. An auto-association happens whenever the states of both input and output units match. This type technique is utilized in order to complete the pattern. In other words, it is used whenever the pattern is distorted or incomplete. In the second technique, pairs of patterns are stored in the network in which it builds an association between two sets of patterns. However, hetero-association can be divided into two algorithms; nearest-neighbor in which the output pattern is dependent on the stored input pattern being the closest to the pattern generated. The second algorithm is interpolative where the output pattern depends on interpolating the stored patterns corresponding to the pattern presented.

The second paradigm is the regularity detection in which units learn to react against certain properties of the input patterns. Unlike the associative mapping where the network saves the connections of the patterns, in regularity detection the output of each unit has a certain meaning. This type can be very useful for feature discovery field [48]. Every NN holds knowledge which is kept in the weights of the connections between the layers. Adjusting these values in the network as an experience function will produce a learning rule for modifying the values of the weights. Further, the information stored in the weights are represented by matrix W of a NN.

Following the way learning is performed, NNs can be distinguished into two categories [49, 50]:

- **Fixed networks** where the weights are fixed, i.e. $\frac{dW}{dt}$=0. In such networks, the weights chosen in advance to solve a problem.
- **Adaptive networks** which are flexible to change the weights, i.e. $\frac{dW}{dt}$ ≠0.

All learning methods used to train the NN can be classified into two major categories; supervised learning which includes an external instructor in order to teach each output the required response against a certain input. Paradigms of supervised learning include reinforcement learning, error-correction learning and stochastic learning. The second method is the unsupervised learning which utilizes local information only. It is known as self-organizing network; in which it organizes the data by itself to be presented to the network. In addition, it detects the collective properties of the data [50].

The NN can be trained offline if the learning and operation phases are different. Usually, supervised learning is executed offline, while unsupervised learning is executed online.

Moreover, to implement the above mentioned learning algorithms, the error derivative for the weight (EW) has to be calculated in order to change the weight by an amount that is proportional to the rate at which the error changes as the weight is changed. To calculate the EW, the weight has to be disrupted slightly so the error changes can be observed. However, this method is inefficient since it requires a separate disruption for each weight.

Another way to calculate the EW is to use the Back-Propagation (BP) algorithm which is described below, and has become nowadays one of the most important tools for training NNs.

2.2.3.2. ***The Back-Propagation (BP) algorithm.*** NN are usually trained by computing the error between the desired and calculated output iteratively for each unit. This is done by calculating the EW of each unit until the error is minimal. This means that the NN has to compute the change of the error whenever the error of each weight is changed. Indeed, the BP algorithm is considered the most widely used method for computing the change of error [51].

In order to calculate the EW, the NN shall first compute the EA; which is the rate as the error changes when the activity level of a unit is varied. However, for the output units, the EA is computed by taking the difference between the actual and desired output. For the hidden unit before the output layer, it is important to first identify all the weights between the hidden unit and the output units. Once recognized, those weights by the EAs of the corresponding outputs shall be multiplied and added to the products. The summation will be equal to the EA for the chosen hidden unit. Moreover, once all the EAs of the hidden units before the output layer, the EAs of the units in other layers shall be computed moving from one layer to another in an opposite direction to the way activities propagate throughout the network. As a result, this technique is named as back propagation. The EW for each incoming connection can be computed easily once the EA of a unit is calculated. EW is defined as the product of the change of error into the activity through the corresponding connection [52]. This is clearer if we take an example as shown in Figure 2.4.

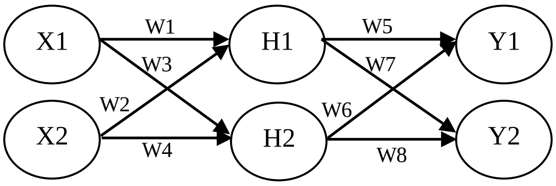

Figure 2.4: NN layers architecture example.

Where X_1 and X_2 are the input signals, $H_1$and H_2 are the hidden layers and the outputs represented by $Y_1$and Y_2. The hidden layer H_1 is calculated as following

$$H_1 = (x_1 \times w_1) + (x_2 \times w_2) \tag{24}$$

After that, the activation function (sigmoid denoted by S [51]) is used

$$S(x) = \frac{1}{1 + e^{-x}} \tag{25}$$

in which the output of H_1 will become

$$out\ H_1 = \frac{1}{1 + e^{-H_1}} \tag{26}$$

The same procedure is applied on H_2 until the output of H_2 is obtained. Later, the outputs can be computed as per below

$$Y_1 = (out\ H_1 \times w_5) + (out\ H_2 \times w_6) \tag{27}$$

$$out\ Y_1 = \frac{1}{1 + e^{-Y_1}} \tag{28}$$

The same way is used to obtain Y_2. Then, the output values are compared with the target values by calculating the total error as below

$$E_{total} = \sum \frac{1}{2}(target - output\ )^2 \tag{29}$$

Consequently, the weights will be updated backwards starting by $w_5, w_6, w_7\ and\ w_8$. The total error calculated will be differentiated with respect to each weight. Below is done for w_5

$$w_5 = w_5 - \eta \times \frac{\partial E_{total}}{\partial w_5} \tag{30}$$

In which η is the learning rate and $\frac{\partial E_{total}}{\partial w_5}$ computed as below

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial out\ Y_1} \times \frac{\partial out\ Y_1}{\partial Y_1} \times \frac{\partial Y_1}{\partial w_5} \tag{31}$$

This is done for each weight and then the same forward calculations are done again. The NN will stop computing until the total error is below the defined tolerance value [52].

Chapter 3. False Data Injection (FDI) System

This chapter begins with an overview about the problem formulation behind this thesis. After that, the system architecture is explained thoroughly by presenting the process of the proposed design with a detailed flowchart. Later, the use of SE and its role in this work is illustrated. Finally, NN design is explored carefully.

3.1. Problem Formulation

Smart grid systems are considered one of the main evolving segments in the power industry. Indeed, smart grids are structured in a way that requires sophisticated and decentralized monitoring control schemes to guarantee a stable functionality of the operation. Yet, allocating monitoring devices at all the buses of the power system is highly costly. As a result, SE is introduced where different techniques can be utilized. Moreover, the communication between the monitoring devices and the SCADA system is vulnerable to external attacks and hence, adding a cyber-layer to the infrastructure is a key element in the power network. The main aim behind this work is to introduce a trustworthy NN that is capable of providing an automatic detection of data manipulation whether bypassed by the state estimator or not over the whole power network. Accordingly, operator will be informed when there is an intrusion over the power network.

3.2. System Model

Several meters have been placed at some buses as shown in Figure 3.1. These meters are used to obtain the data of the buses including real and reactive power injections as well as real and reactive power flowing between the buses. These data are listed in Appendix B. The bus data is utilized to find the bus state variables (phase angles) through the SE. This technique can be used later in validating the phase angles found by SE. In this algorithm, Q- V equations are totally neglected, and all the bus voltages are constants at 1.0 per unit. Thus, the line power flow equation between bus j and k with X reactance is simplified to

$$P_{jk} = \frac{\delta_j - \delta_k}{X_{jk}} \tag{32}$$

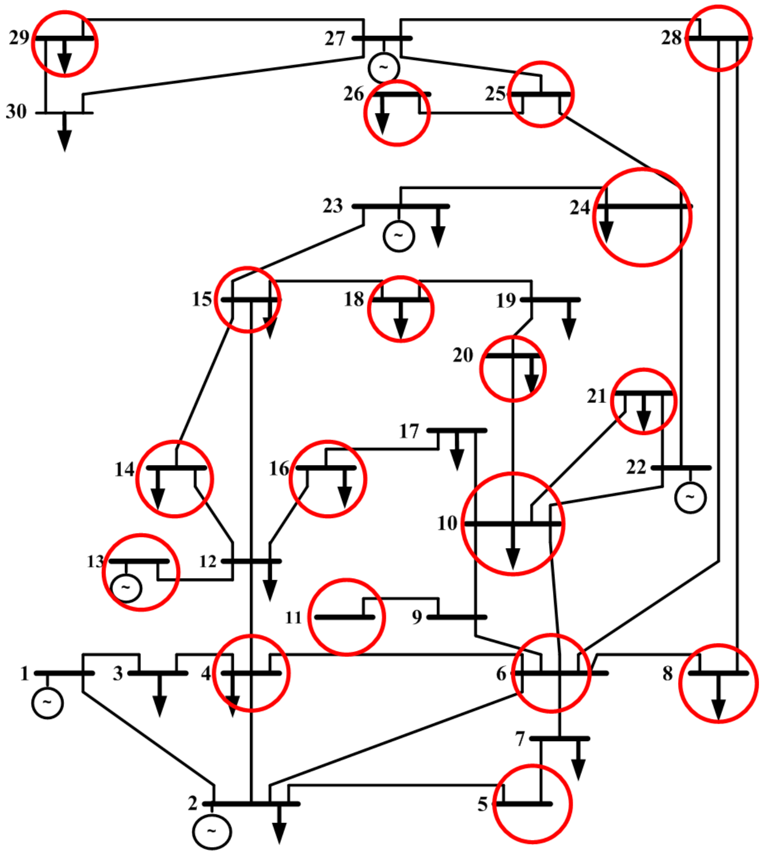

Figure 3.1: IEEE-30 bus system with located meters.

So, the real power equations reduce to a linear problem as

$$-\bar{B}\bar{\delta} = \bar{P} \tag{33}$$

Where $\bar{B}$ is an imaginary component of Y_{bus} calculated neglecting line resistance, $\bar{\delta}$ is the phase angles vector and $\bar{P}$ is a vector of real power injections (with assumed positive generation). Later, SE is performed to estimate the state variables of the buses. After that, the phase angles are compared with the ones obtained in the load power flow (NR) to validate the correctness of the SE technique. Finally, the obtained data is passed through a NN model to detect and ensure no data manipulation has occurred. The flowchart in Figure 3.2 illustrates the process proposed in this thesis.

3.3. State Estimation (SE) Algorithm

The SE is done through WLS method. The raw data attained from the bus system are used to estimate the voltages and phase angles of all the buses. This method outstands other techniques in minimizing the error as much as possible.

3.4. False Data Injection (FDI) Attack

FDI attacks can be conducted on different parts of the power grid such as transmission systems, distribution systems, advanced metering infrastructure, etc [30].

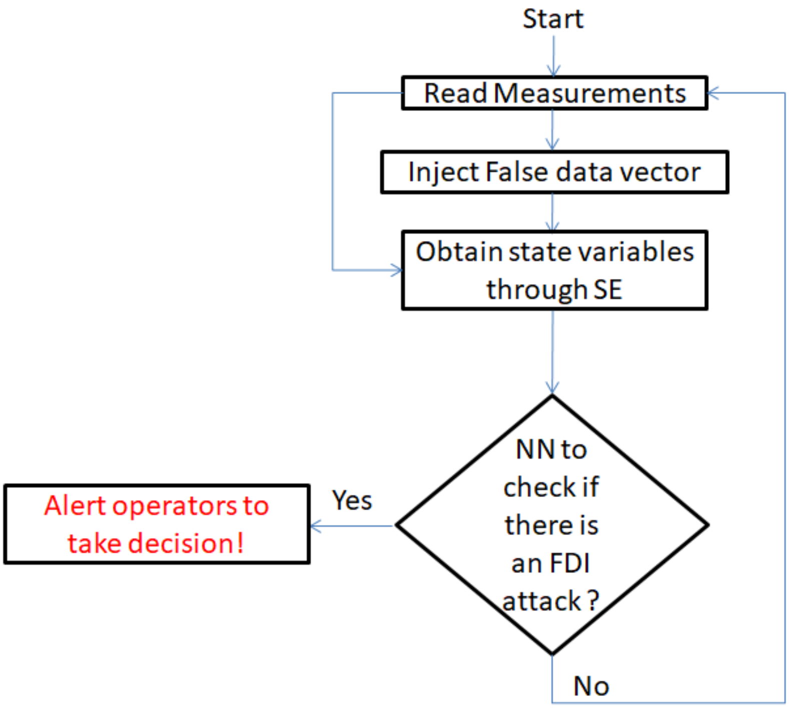
Start
Read Measurements
Inject False data vector
Obtain state variables through SE
NN to check if there is an FDI attack ?
Yes
Alert operators to take decision!
No

Figure 3.2: Process flowchart.

In this thesis, a random FDI attacks on the static SE in the AC power transmission system are considered. In fact, the attacker is assumed to have a limited access to a certain number of meters in which their physical data (voltages, currents, phase angles) can be manipulated. This is achieved by corrupting the measurement vector communicated by the SCADA and hence launch an FDI attack. However, this requires a pre-knowledge of the needed elements of the measurement matrix H. It is assumed that the attacker has the ability to learn these system information prior to devising the attack, either because the attacker is a trusted insider or has hacked into the system databases.

The process for constructing a vector for an FDI attack is as follows. Any non-zero arbitrary vector can be chosen by the attacker to be as attack vector $\bar{a}$ in order to corrupt the measurement vector

$$\bar{z}_a = \bar{z} + \bar{a} \tag{34}$$

Where $\bar{z}$ is the original measurement vector. This vector holds all the power injections and flow measurements that are used to estimate the state variables (voltages and phase angles) of the plant buses. Once this vector is corrupted then the estimated state variables will be affected accordingly. In usual state estimators, traditional bad data detection methods are able to detect an attack if the residual $\bar{r} = ||\bar{z}_a - H\hat{\hat{x}}||$ is more than a certain predefined threshold τ (usually chosen to be 3 [30]). Nevertheless, if the attacker uses $\overline{H}\bar{c}$ as the attack vector $\bar{a}$ (i.e., $\bar{a} = H\bar{c}$); where $\bar{c}$ is a non-zero error vector of length n injected by the attacker; then, the L_2-norm of the measurement residual of $\bar{z}_a$ is equal to that of $\bar{z}$, as shown below.

$$\begin{aligned} ||\bar{z}_a - H\hat{\hat{x}}|| &= ||\bar{z} + \bar{a} - H(\hat{\hat{x}} + \bar{c})|| \\ &= ||\bar{z} - H\hat{\hat{x}} + (\bar{a} - H\bar{c})|| \\ &= ||\bar{z} - H\hat{\hat{x}}|| \leq \tau \end{aligned} \tag{35}$$

In other words, if the attacker chooses $\bar{a}$ as a linear combination of the column vectors of H, then, $\bar{z}_a$ can pass the detection as long as $\bar{z}$ can pass the detection [30]. Hence, the resulted state variables vector $\hat{\hat{x}}_{bad}$ can be expressed as $\hat{\hat{x}} + \bar{c}$**;** where $\hat{\hat{x}}$ is the healthy state variables vector. Using this method will allow the corrupted vector to be processed without being detected by the state estimators.

In this thesis, the above system model is constructed in which a NN model is proposed to detect an FDI attack that it is bypassed by the traditional state estimators. It is important to note that the attack vector is being generated randomly and the attacked meters are being selected randomly as well. The simulated results are shown in the simulation results section.

3.5. Neural Network (NN) Model

The NN tool in MATLAB is used in which is based on the BP algorithm. The estimated input data vectors are passed into the NN to check if the estimated data are the same as the target values. A certain threshold has been set in which if the change in data exceeds that threshold, an intrusion is detected. The NN acts as a cyber layer to protect and ensure a reliable data being displayed to the operators. The NN model consists of ten hidden layers with one input vector and one output vector.

Chapter 4. Power System Simulation

In this chapter, the previously mentioned IEEE-30 bus system is used in estimating the bus state variables. Implementation of this architecture along with the employed algorithms for real-time data monitoring is presented. MATLAB simulator is used to validate the proposed design.

4.1. Power Flow by Newton Raphson (NR)

To begin with, a power flow analysis using NR method has been performed to find out the phase angles of the buses so it can be used later on in validating the SE algorithm. The 30-buses system data are fed into the load power flow calculation in which there are six generator buses, one slack bus and the rest of the buses are considered as load buses. Furthermore, the line data parameters are selected as per Appendix A. These data are used to calculate the nodal admittance matrix known as the Y-bus matrix.

Once the admittance matrix is calculated, the Susceptance (B) matrix can be attained since it is the imaginary part of the admittance as mentioned earlier in the report. Nevertheless, the slack bus is removed before calculating the real power of lines which is used accordingly in determining the phase angles as shown Figure 4.1.

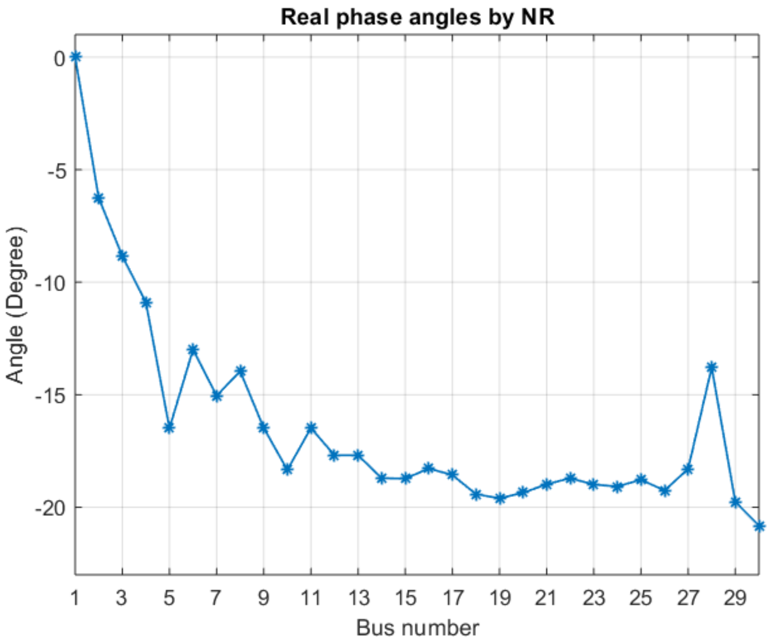

Figure 4.1: Phase angles of the IEEE-30 bus system using NR.

4.2. State Estimation (SE) Validation

In this design, SE is performed using the WLS technique to acquire the state variables; voltages and phase angles. The previous IEEE-30 bus system is utilized considering both real and reactive powers as per Appendix B.

The same line data presented in Appendix A is used in calculating the Y-bus matrix. The admittance matrix is used in calculating power flow equations to estimate the state variables. The resulting phase angles are compared with the NR results in order to assess the reliability of the SE algorithm in estimating the correct state variables. Both results are plotted as shown in Figure 4.2.

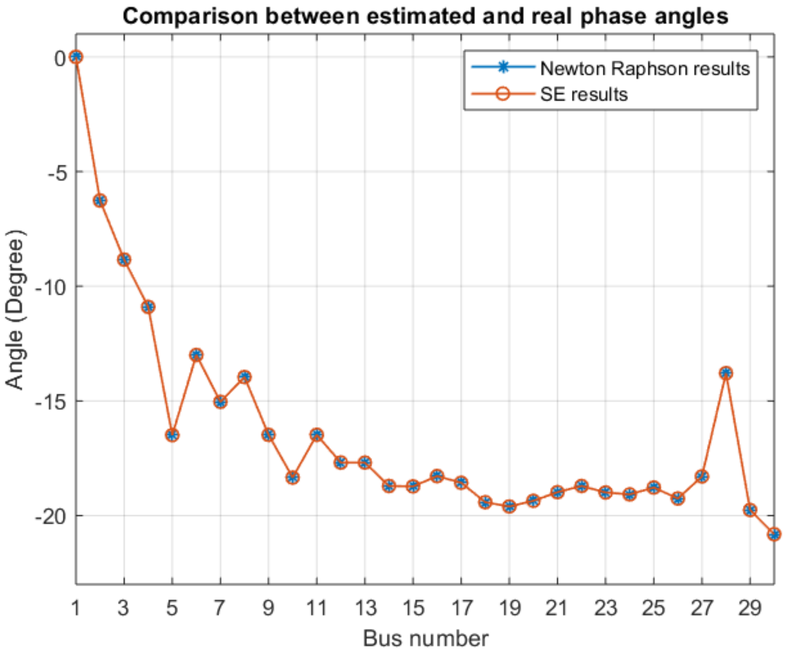

Figure 4.2: Comparison between estimated and real phase angles.

It is clearly shown in Figure 4.2 that the estimated phase angles are exactly matching the real values for a single sample measured at each of the 30 buses. After validating the SE technique, the attack vector is built in order to be injected to the measurement vector.

4.3. Attack Vector

The attack vector can be simulated as an offset added to the measurement vector. For example, if the attacker has an access to meters 12, 76, and 84 in the

IEEE-30 bus system shown in Figure 3.1 then, the attack vector $\bar{a}$ is shown in Table 4.1.

Table 4.1: Attack vector.

Meter #	Offset value
1 ⋮	0 ⋮
12	0.021194
⋮	0 ⋮
76	0.023478
⋮	0 ⋮
84	0.0353159
93	⋮ 0

Note that the empty elements are zeros. Hence, the corrupted measurement vector $\bar{z}_a = \bar{z} + \bar{a}$, is fully shown in Appendix C, however, the corrupted meters are highlighted in Table 4.2 below.

Table 4.2: Healthy measurement vector vs. corrupted measurement vector.

Type of measurement	Meter #	Healthy measurement value	Corrupted measurement value
⋮	1 ⋮	⋮	⋮
Real power injection	12	-0.032	-0.011
⋮	⋮	⋮	⋮
Reactive power flow	76	-0.159	-0.136
⋮	⋮	⋮	⋮
Reactive power flow	84	-0.097	-0.062
⋮	⋮ 93	⋮	⋮

Comparing both measurement vectors, it is observed that the measurement values of the attacked meters only (highlighted) have been changed. In this thesis, the meters reachable by the attacker are randomly selected in each trial since this may vary each time depending on the knowledge the attacker has with respect of the power plant topology. Similarly, the offset values of the selected meters are randomly generated within the range of (0 to 1) in each trial. The same is applied with the IEEE-30 bus system.

4.4. Neural Network (NN) Training

In order to train the NN, several tests have been conducted in which five sets of measurements are taken at different times of the day. In each test, SE has been applied to estimate the corresponding voltages and phase angles. Later, different meters are attacked with different attack vectors to be fed to the NN for FDI attack detection. A total of 30 random attack vectors are injected into each measurement set of values. All the tests are listed in Appendix D.

4.4.1. Test 1. In this test, SE is first conducted and the results of the state variables compared to the real ones are shown in Figure 4.3. It is observed that the estimated state variables are almost the same as the real variables obtained by NR method.

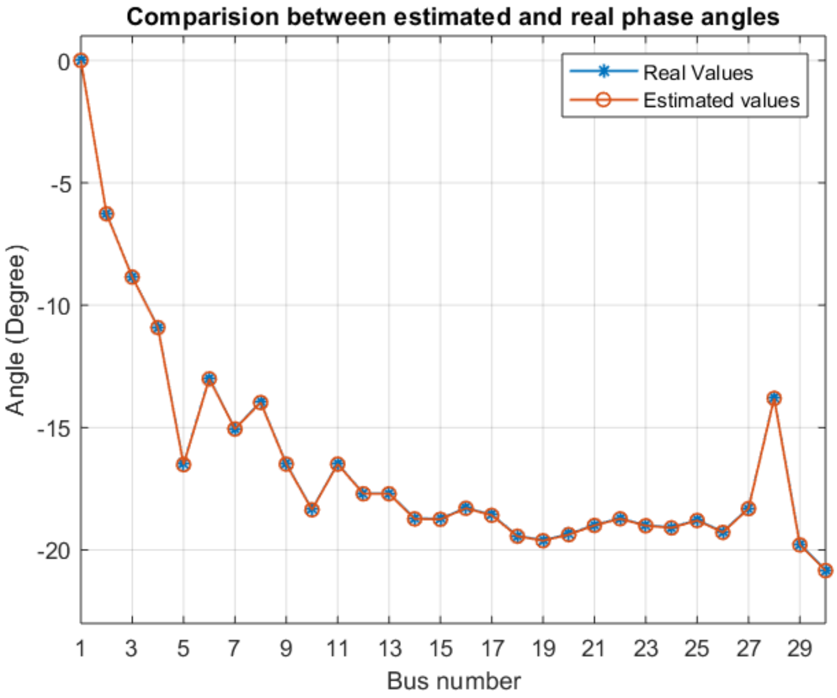

Figure 4.3: Comparison between estimated and real phase angles (test 1).

4.4.2. Test 2. In this test, different meters for real and reactive power injections have been changed as indicated in Appendix D (Test 2). However, SE is first conducted and the results of the state variables compared to the real ones are shown in Figure 4.4.

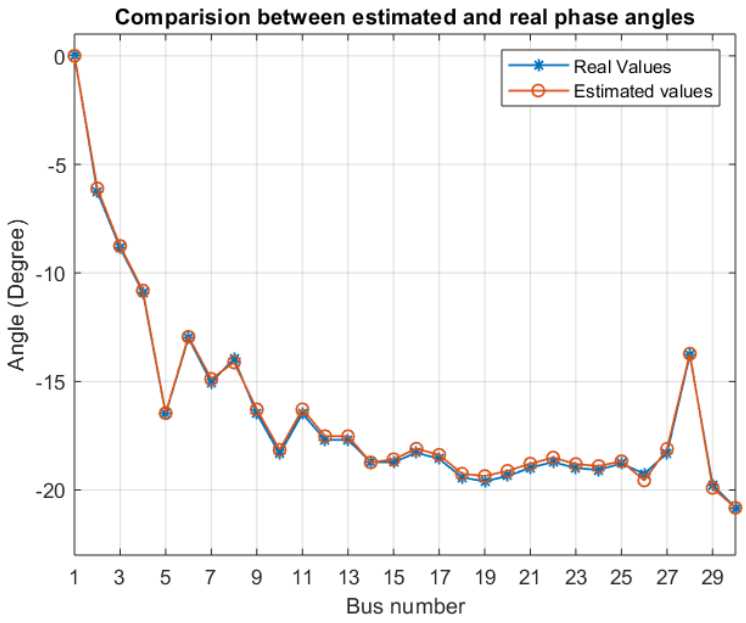

Figure 4.4: Comparison between estimated and real phase angles (test 2).

It is observed from Figure 4.4 that the estimated state variables are almost the same as the real variables obtained by NR method.

4.4.3. Test 3. In this test, different meters for real power injections, reactive power injections, real power flow and reactive power flow measurements have been changed as indicated in Appendix D (Test 3). However, SE is first conducted and the results of the state variables compared to the real ones are shown in Figure 4.5. It is observed that the estimated state variables are almost the same as the real variables obtained by NR method.

4.4.4. Test 4. In this test, different meters for real power injections, reactive-power injections, real power flow and reactive power flow measurements have been changed as indicated in Appendix D (Test 4). However, SE is first conducted and the results of the state variables compared to the real ones are shown in Figure 4.6. From Figure 4.6, it is noticed that the estimated state variables are approximately the same as the real variables obtained by NR method. However, there is a difference of one degree between some of the phase angles as highlighted. However, there are several reasons behind why the numbers are so different from the original values.

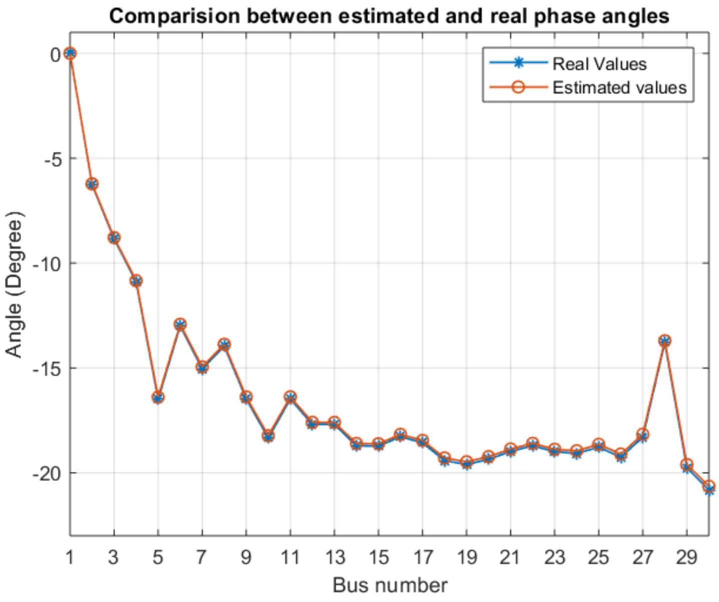

Figure 4.5: Comparison between estimated and real phase angles (test 3).

One of the reasons is obviously the measurement. Sometimes, the measurements themselves are deviated from the “True” value by almost 10% [32]. The second reason is the error variance of the measuring devices. In this case, σ was taken to be 0.01. Therefore, the main objective function behind the WLS method is to minimize the Jacobian matrix, given in equation 15.

4.4.5. Test 5. In this test, SE is again conducted and the results of the state variables (phase angles) compared to the real ones are shown in Figure 4.7. It is observed that the estimated phase angles are almost the same as the real angles obtained by NR method. However, there is 0.13 degree difference in some of the phase angles. As mentioned earlier, this difference could be due to several reasons such as variances in the error of a meter’s measurement. Each of the above SE results are attacked with 30 random attack vectors that ranges between 0 -20% of the measurement values and fed into the NN.

A total number of 250 samples representing voltages and phase angles have been fed into the NN in which it includes 175 corrupted and 75 healthy samples. In the proposed NN model, twenty hidden layers have been chosen since it provides the required optimum output in a shortest timely manner compared to higher number of layers.

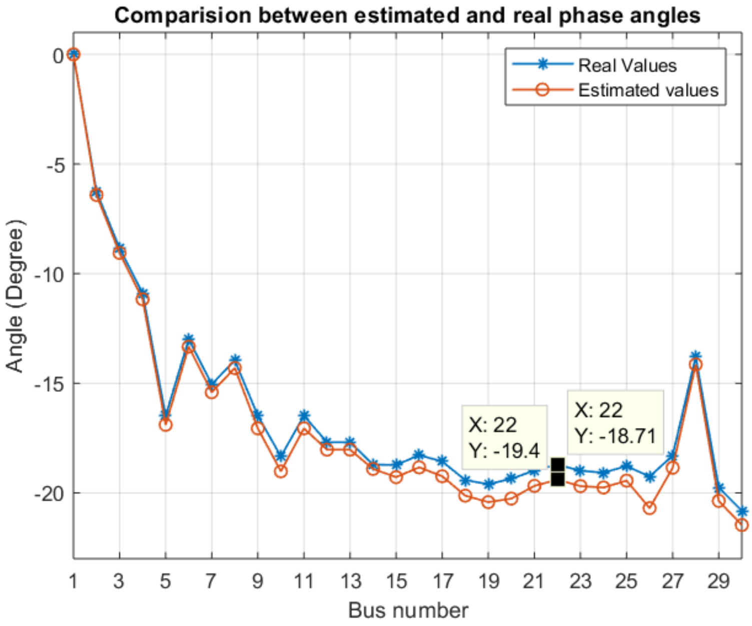

Figure 4.6: Comparison between estimated and real phase angles (test 4).

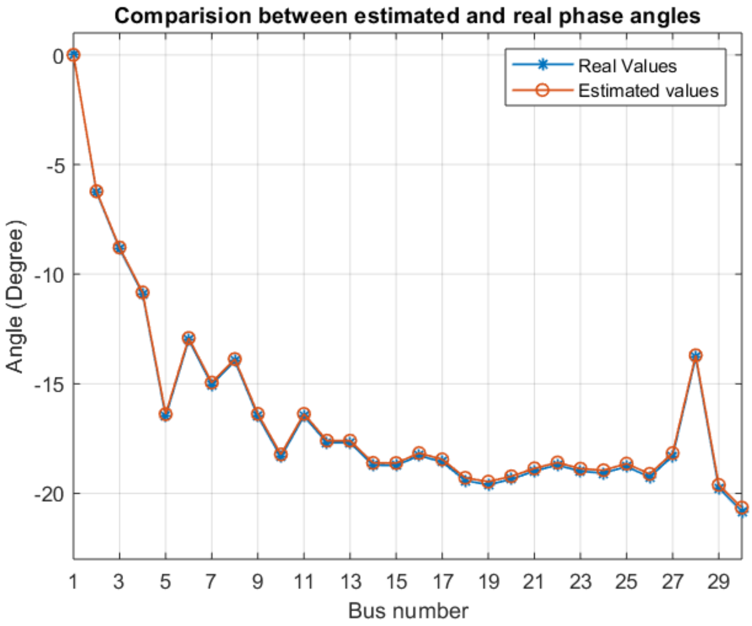

Figure 4.7: Comparison between estimated and real phase angles (test 5).

Figure 4.8 illustrates the NN architecture taken by MATLAB. It is clearly shown in Figure 4.8 that the input vector consists of 59 state variables representing the voltages and phase angles of the IEEE-30 bus system excluding the slack bus. Following that, 20 hidden layers are chosen to be connected to the output layer. Finally, the output vector consists of two categories as True and False.

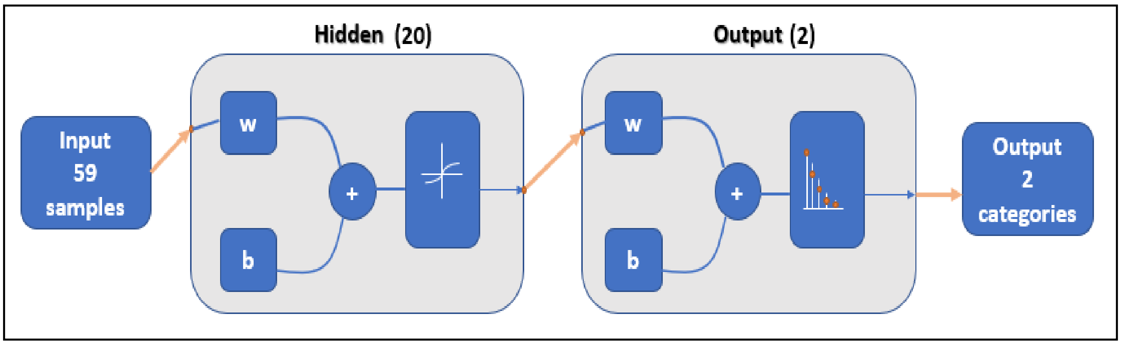

Figure 4.8: NN architecture.

The confusion matrix is plotted to observe the behaviour of the NN design as shown in Figure 4.9.

A closer look at Figure 4.9 indicates that 165 samples have been identified and detected correctly. However, 9 samples are detected as healthy samples while they are corrupted measurements. Further, the NN model has been validated in which 2 samples only were detected wrong out of 38 samples. In the test stage, two samples are only detected wrongly while the other 36 samples are detected successfully.

4.5. Tested System Model

Two tests have been conducted on the 30-bus system in which the number of corrupted meters is changed. It has been shown in [10] that the minimum number of attacked meters needed to detect an FDI intrusion, is four. However, this work illustrates that the designed NN can detect FDI attacks when four meters are attacked, or even if only three meters are attacked.

4.5.1. Four meters test. Measurements from a random set of four meters have been corrupted with random offset values. For example, the corrupted meters are (62, 17, 12, 90) with added offset values of [0.070, .0166, .0153, 0.169] respectively. The SE is performed with the given measurements and the obtained state variables are shown in Appendix D. However, a major change is highlighted in Table 4.3 below. It is observed from Appendix D that the main impact is on the phase angles in which it deviates with a maximum value of 17.65 degrees as highlighted in Table 4.3. The resulting vector is passed through the NN and the result is shown Table 4.4.

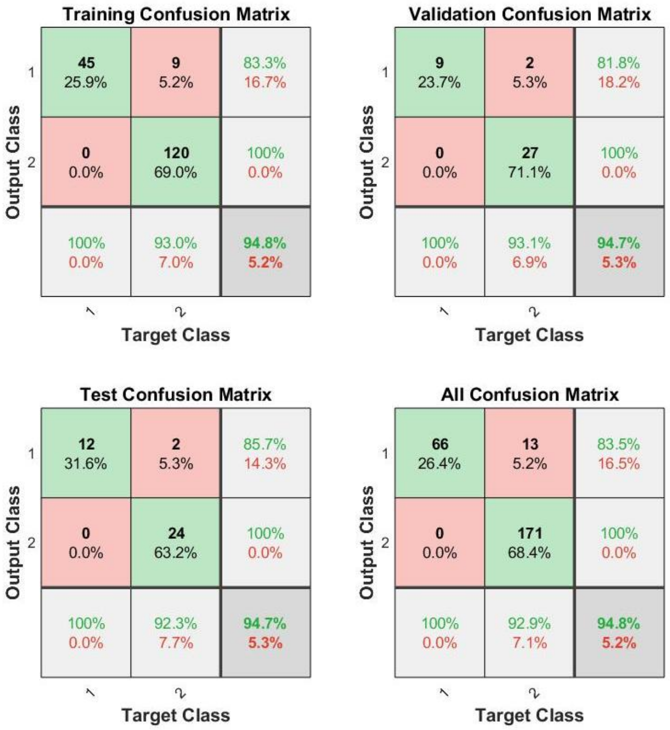

Figure 4.9: Confusion matrix.

Table 4.3: Measurement vectors of healthy and corrupted vectors (four meters).

Bus	Corrupted measurement vector		True measurement vector		Deviation of Voltage from the true value	Deviation of phase angle from the true value
	Voltage (p.u)	Phase angle (degree)	Voltage (p.u)	Phase angle (degree)		
26	1.11198	1.604819	0.9071	-19.2564	0.204881	17.6516

Table 4.4: NN result (four meters).

NN categories	Confidence rate for 4 meters test
True	5.6 %
False	94.4 %

4.5.2. Three meters test. Measurements from three meters have been corrupted with random offset values. The meters (12, 76, 84) are corrupted with an attack vector of [0.235, 0.353, 0.182] respectively. The obtained SE results are shown in Appendix E. However, the major change highlighted in Table 4.5.

Table 4.5: Healthy and corrupted measurement vectors (three meters with residual higher than 3).

Bus	Corrupted measurement vector		True measurement vector		Deviation of Voltage from the true value	Deviation of phase angle from the true value
	Voltage (p.u)	Phase angle (degree)	Voltage (p.u)	Phase angle (degree)		
18	1.043305	-14.1118	0.935246	-19.4165	0.108059	5.304693

It is observed from Table 4.5 that the phase angles deviate with a maximum value of 5.3 degrees. As per [31], if the residual is bigger than a certain threshold σ (usually 3 degrees) then, it is detected by normal state estimators. Further, another three meters test has been conducted in which meters (9, 59, 85) are corrupted with an attack vector of [0.234, 0.253, 0.132] respectively. However, SE is not capable of detecting the corruption since the residual is below than 3. The results of the estimated state variables are shown in Appendix F. However, the major change is highlighted in Table 4.6. It is observed from Table 4.6 that the maximum difference is 2.3 degrees which is usually not detected by SE. Yet, the corrupted measurement vector is fed into the NN and the results are shown in Table 4.7.

Table 4.6: Healthy and corrupted measurement vectors (three meters with residual less than 3).

Bus	Corrupted measurement vector		True measurement vector		Deviation of Voltage from the true value	Deviation of phase angle from the true value
	Voltage (p.u)	Phase angle (degree)	Voltage (p.u)	Phase angle (degree)		
14	0.982818	-16.3776	0.955965	-18.7107	0.026852	2.33318

Table 4.7: NN Result (three meters).

NN categories	Confidence rate for 3 meters test.
True	5.3 %
False	94 %

From Table 4.7, NN classifies the tested sample as a corrupted measurement with 94% confidence. This means that the NN is detecting the corruptions successfully.

4.6. Neural Network (NN) Detection Capability

The NN model has been tested with descending attack values to test and validate its detection capability. As mentioned earlier, the considered change of the attack values ranges from 0-20%. Hence, several tests have been simulated in which the range of change has been varied considering both three and four meters tests.

4.6.1. Three meters test. In this test, five different cases are presented. In the first case, three meters are attacked with a vector of 20% deviation from the true measurement values that represent reactive power injection, real and reactive power flow. Further, this test is implemented on the five sets of measurements presented earlier in section 4.4. Table 4.8 summarizes the average confidence rates of the NN results against the five measurement sets.

Table 4.8: NN Result with 20% change.

NN categories	Confidence rate of 3 meters test for all tests.
True	1.2 %
False	98.8 %

It is observed from Table 4.8 that NN classifies the tested sample as a corrupted measurement with 98.8% confidence. This means the NN can detect the attack successfully. In the second case, the attack vector values have been generated with a 10% deviation from the measurement values that represent real power injection, reactive power injection and real power flow to check if the NN can still detect an attack or not. Table 4.9 summarizes the average confidence rates of the NN against the five sets of measurements.

Table 4.9: NN Result with 10% change.

NN categories	Confidence rate of 3 meters test for all tests.
True	2.4 %
False	97.6 %

It is observed from Table 4.9 that the NN classifies that the average confidence of false data is 97.6 % which means the NN can detect the attack. In the second case, the attack vector values have been generated with a 5% deviation from the measurements values to check if the NN can still detect an attack or not. Table 4.10 summarizes the average confidence rates of the NN against the five sets of measurements. It is clearly shown from Table 4.10 that the NN can detect an attack successfully with 55.6 % confidence. Further, the deviation percentage of the attack vector from the true measurements has been decreased to 4% and the corresponding results of the NN are shown in Table 4.11.

Table 4.10: NN Result with 5% change.

NN categories	Confidence rate of 3 meters test for all tests.
True	44.4 %
False	55.6 %

Table 4.11: NN Result with 4% change.

NN categories	Confidence rate of 3 meters test for all tests.
True	49 %
False	51 %

Table 4.12: NN Result with 3% change.

NN categories	Confidence rate of 3 meters test for all tests.
True	63 %
False	37 %

It is observed from Table 4.11 that the NN results are very close from each other, however, the NN can still detect the attack. Moreover, the deviation of the attack vector against the measurement values has been decreased further to 3 % as illustrated in Table 4.12. The NN confidence of true data is 63 % which is a wrong detection. Indeed, the NN fails to detect any attack lower than 4% deviation from the true measurement values.

4.6.2. Four meters test. In this test, two different cases are presented. In the first case, four meters are attacked with 3% change in which the attacked measurements represent real power injection, reactive power injection and real power flow. Further, this test is implemented on the five sets of measurements presented earlier. Table 4.13 summarizes the average rates of the NN against the five sets of measurements.

Table 4.13: NN Result with 3% change.

NN categories	Confidence rate of 4 meters test for all tests.
True	32 %
False	68%

It is observed from Table 4.13 that the average confidence of false data is 68% which means the NN can detect the attack. In the second case, the deviation percentage of the attack vector from the true measurements has been decreased to 2% to check if the NN can still detect the attack. Table 4.14 summarizes the average rates of the NN results against the five sets of measurements. As illustrated in Table 4.14, the NN fails to detect an attack since the false confidence rate is only 33%.

Table 4.14: NN Result with 2% change.

NN categories	Confidence rate of 4 meters test for all tests.
True	67 %
False	33 %

Furthermore, a random generation of the attack vector is conducted in which it changes randomly between 0 - 20% of the measurement values. This has been implemented on the five sets of measurements shown earlier. Table 4.15 summarizes the results of the NN against these tests.

Table 4.15: Summarized NN results against the five measurement sets.

	How many times attacked	**Success rate**	**Failure rate**
Test 1	40	97%	3%
Test 2	40	92%	8%
Test 3	30	90%	10%
Test 4	30	91%	9%
Test 5	30	90%	10%
Total		**92 %**	**8 %**

Chapter 5. Conclusion and Future Work

In this thesis, a Neural Network (NN) based strategy is developed to detect data manipulation i.e. external attacks on bus state variables; voltages and phase angles, when bypassed by the traditional state estimators (SEs). The NN developed for detecting data manipulation is built in Matlab, and it is fed with historical training measurements of all buses states variables from different times of the day, to train the NN about the normal operational states of the power system buses. External false data injection (FDI) attacks, are simulated by manipulating random meters' measurements of real and reactive power injections, as well as real and reactive power flows in the IEEE 30-bus system.

The NN developed, successfully detects the FDI attacks when SE overlooks the manipulation, i.e. the threshold of the residual between the true and estimated values are satisfied, even if certain states have reasonably significant deviations from their true value. Simulation results show that the proposed NN based technique is able to detect data manipulation, i.e. FDI attacks even if as low as three meters have been manipulated. It is shown that successful detection of the FDI attack by the NN occurs, if any three measurements across three meters deviate by at least 4% of their true values.

As a future work, testing the power systems data with different NN structures shall be developed to enhance its detection covering different types of system topologies. Further, critical meters in power system shall be studied and investigated to identify the vulnerabilities in the power grid system. In addition, a field test of the designed NN model can be implemented to ensure its robustness and enhance its performance accordingly. Moreover, an additional direction for future work is to try a different type of intrusion with the developed NN to improve its detection capabilities.

References

[1] Y. Wu, A. Onwuachumba and M. Musavi, "Bad Data Detection and Identification Using Neural Network-Based Reduced Model State Estimator," 2013 *IEEE Green Technologies Conference (GreenTech)*, Denver, CO, USA, 2013, pp. 183-189.

[2] Power System State Estimation. [Online]. Available: https://www.kth.se /social/up load/Lecture_15 StateEstimation.pdf [Accessed: 02-January-2019].

[3] Y. Mo et al., "Cyber–Physical Security of a Smart Grid Infrastructure," *IEEE Proceedings - Generation, Transmission and Distribution*, vol. 100, no. 1, pp. 195-209, Jan. 2012.

[4] S. Mangalwedekar and S. K. Surve, "Measurement sets in power system state estimator in presence of false data injection attack," *2015 IEEE International Advance Computing Conference (IACC)*, Banglore, India, 2015, pp. 855-860.

[5] C. Liu, J. Wu, C. Long and D. Kundur, "Reactance Perturbation for Detecting and Identifying FDI Attacks in Power System State Estimation," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 4, pp. 763-776, Aug. 2018.

[6] Z. Hu, Y. Wang, X. Tian, X. Yang, D. Meng and R. Fan, "False data injection attacks identification for smart grids," *2015 Third International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE)*, Beirut, Lebanon, 2015, pp. 139-143.

[7] G. Chaojun, P. Jirutitijaroen and M. Motani, "Detecting False Data Injection Attacks in AC State Estimation," in *IEEE Transactions on Smart Grid*, vol. 6, no. 5, pp. 2476-2483, Sept. 2015.

[8] A. Nourian and S. Madnick, "A Systems Theoretic Approach to the Security Threats in Cyber Physical Systems Applied to Stuxnet," in *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 1, pp. 2-13, 1 Jan.-Feb. 2018.

[9] S. Poudel, Z. Ni, X. Zhong and H. He, "Comparative studies of power grid security with network connectivity and power flow information using unsupervised learning," *2016 International Joint Conference on Neural Networks (IJCNN)*, Vancouver, BC, Canada, 2016, pp. 2730-2737.

[10] Schweppe F, Wildes J, and Rom D. “Power system static state estimation: Parts I, II, and III”. Denver, Colorado, USA, *Power Industry Computer Conference (PICA)*, June, 1969.

[11] J. K. Mandal, A. K. Sinha and L. Roy, "Incorporating nonlinearities of measurement function in power system dynamic state estimation," in *IEEE Proceedings - Generation, Transmission and Distribution*, vol. 142, no. 3, pp. 289-296, May 1995.

[12] K. Verma and K. R. Niazi, "Determination of vulnerable machines for online transient security assessment in smart grid using artificial neural network," 2011 *Annual IEEE India Conference*, Hyderabad, India, 2011, pp. 1-5.

[13] Y. Yang, T. Littler, S. Sezer, K. McLaughlin, and H. Wang, “Impact of cyber-security issues on smart grid,” in *Innovative Smart Grid Technologies (ISGT Europe), 2011 2nd IEEE PES International Conference and Exhibition on*, Bucharest, Romania IEEE, 2011, pp. 1–7.

[14] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, “A survey on cyber securityfor smart grid communications,” *IEEE Communications Surveys &Tutorials*, 2012.

[15] Y. Liu, P. Ning, and M. K. Reiter, “False data injection attacks againststate estimation in electric power grids,” *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, p. 13, 2018.

[16] A. Ashok, M. Govindarasu, and J. Wang, “Cyber-physical attack resilient wide-area monitoring, protection, and control for the power grid,” *IEEE Proceedings - Generation, Transmission and Distribution*, 2017.

[17] Load Flow Studies. [Online]. Available: https://nptel.ac.in/courses/ Webcourse-contents/IIT-KANPUR/power-system/chapter_4/4_10.html [Accessed: 23-January-2019].

[18] J.Glover, M. S Sarma, and T. J Overbye. “Power System Analysis and Design”, 5th ed. Stamford: Cengage Learning, 2017, p. 345-362.

[19] K. R. Niazi, C. M. Arora and S. L. Surana, "Power system security evaluation using ANN: feature selection using divergence," *Proceedings of the International Joint Conference on Neural Networks, 2003.*, Portland, OR, 2003, pp. 2094-2099 vol.3.

[20] M. Cramer, P. Goergens and A. Schnettler, "Bad data detection and handling in distribution grid state estimation using artificial neural networks," 2015 *IEEE Eindhoven PowerTech*, Eindhoven, Netherlands, 2015, pp. 1-6.

[21] Load Flow Studies. [Online]. Available: https://cnls.lanl.gov/~ chertkov/SmarterGrids/Talks/Milano.pdf [Accessed: 10-Feburary-2019].

[22] A. Monticelli, Electric power system state estimation, *IEEE Proceedings - Generation, Transmission and Distribution*, vol. 88, no. 2, p. 262–282, Feb 2000.

[23] Y.-F. Huang, S. Werner, J. Huang, N. Kashyap, and V. Gupta, State estimation in electric power grids: Meeting new challenges presented by the requirements of the future grid, *Signal Processing Magazine, IEEE,* vol. 29, no. 5, p. 33–43, 2012.

[24] M. Baran and A. Kelley, “A branch-current-based state estimation method for distribution systems,” *Power Sys., IEEE Trans*., vol. 10, no.1, pp. 483–491, Feb 1995.

[25] A. Meliopoulos and F. Zhang, “Multiphase power flow and state estimation for power distribution systems,” *Power Sys., IEEE Trans*.vol. 11, no. 2, pp. 939–946, May 1996.

[26] S. Bi and Y. J. Zhang, “Graphical methods for defense against false-data injection attacks on power system state estimation,” *IEEE Transactions on Smart Grid*, vol. 5, no. 3, pp. 1216–1227, 2014.

[27] R. Niu and L. Huie, "System state estimation in the presence of false information injection," *2012 IEEE Statistical Signal Processing Workshop (SSP)*, Ann Arbor, MI, 2012, pp. 385-388.

[28] D. Falco and U. Bezerra, “Power system operating state forecasting for security analysis applications,” *Int. Journal of Elec. Power& Energy Sys*., vol. 13, no. 6, pp. 330 – 336, 1991.

[29] Weighted-Least-Square (WLS) in State Estimation. [Online]. Available: https://www.gridpack.org/wiki/images/c/cd/SE.pdf [Accessed: 04-March-2019].

[30] H.-J. Koglin, T. Neisius, G. Beiler, and K. Schmitt, “Bad data detection and identification,” *Int. Journal of Elec. Power & Energy Sys*., vol.12, no. 2, pp. 94 – 103, 1990.

[31] S. Bi and Y. J. Zhang, “Graphical methods for defense against false-data injection attacks on power system state estimation,” *IEEE Transactions on Smart Grid*, vol. 5, no. 3, pp. 1216–1227, 2014.

[32] A. Leite da Silva, M. Do CouttoFilho, and J. F. De Queiroz, “State forecasting in electric power systems,” *Gen., Tran. & Dist., IEE Proc*,. vol. 130, no. 5, pp. 237–244, September 1983.

[33] A. Ghosh, D. Lubkeman, and R. Jones, “Load modeling for distribution circuit state estimation,” *Power Delivery, IEEE Trans*., vol. 12, no.2, pp. 999–1005, Apr 1997.

[34] S. Julier and J. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, Mar 2004.

[35] E. Falak, N. Nguyen, R. Zheng and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," 2018 *IEEE Global Communications Conference (GLOBECOM),* Atlanta, GA, USA 2018, pp. 808-813.

[36] S. Mangalwedekar, S. K. Surve and H. A. Mangalvedekar, "False Data Injection Attacks and detection scenarios in the power system," *2015 Annual IEEE India Conference (INDICON)*, New Delhi, India, 2015, pp. 1-6.

[37] J. Zhu and X. Wei, "Defending false data injection attacks against power system state estimation: A stealthiness corruption-oriented method," *2016 IEEE International Conference on Power System Technology (POWERCON)*, Wollongong, NSW, Australia, 2016, pp. 1-5.

[38] R. J. R. Kumar and B. Sikdar, "Efficient detection of false data injection attacks on AC state estimation in smart grids," *2017 IEEE Conference on Communications and Network Security (CNS)*, Las Vegas, NV, USA, 2017, pp. 411-415.

[39] Q. Deng and J. Sun, "False Data Injection Attack Detection in a Power Grid Using RNN," *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*, Washington, DC, USA, 2018, pp. 5983-5988.

[40] M. Ashrafuzzaman *et al*., "Detecting Stealthy False Data Injection Attacks in Power Grids Using Deep Learning," *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, Limassol, Cyprus, 2018, pp. 219-225.

[41] Q. Yang, J. Yang, W. Yu, D. An, N. Zhang and W. Zhao, "On False Data-Injection Attacks against Power System State Estimation: Modeling and Countermeasures," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 3, pp. 717-729, March 2014.

[42] Y. Zhou and Z. Miao, "Cyber attacks, detection and protection in smart grid state estimation," *2016 North American Power Symposium (NAPS)*, Denver, CO, USA, 2016, pp. 1-6.

[43] A. Larsson, A. Germond and B. Zhang, "Application of Neural Networks to the Identification of Steady State Equivalents of External Power Systems," *2006 International Conference on Power System Technology*, Chongqing, China, 2006, pp. 1-6.

[44] Yu Jilai and Liu Zhuo, "Artificial neural networks based steady state equivalents of power systems," *Proceedings of the First International Forum*

on Applications of Neural Networks to Power Systems, Seattle, WA, USA, 1991, pp. 174-177.

[45] T. Nakagawa, Y. Hayashi and S. Iwamoto, "Neural network application to state estimation computation," *Proceedings of the First International Forum on Applications of Neural Networks to Power Systems*, Seattle, WA, USA, 1991, pp. 188-192.

[46] H. Mori, "An artificial neural net based method for power system state estimation," *Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan)*, Nagoya, Japan, 1993, pp. 1533-1536 vol.2.

[47] Y. Zhou and Z. Miao, "Cyber attacks, detection and protection in smart grid state estimation," *2016 North American Power Symposium (NAPS)*, Denver, CO, USA, 2016, pp. 1-6.

[48] A. Ayad, H. E. Z. Farag, A. Youssef and E. F. El-Saadany, "Detection of false data injection attacks in smart grids using Recurrent Neural Networks," *2018 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, Washington, DC, USA, 2018, pp. 1-5.

[49] How Electricity Is Delivered To Consumers. [Online]. Available: https://www.eia.gov/energyexplained/index.php?page=electricity_delivery [Accessed: 14-February-2019].

[50] N. H. Abbasy, "Neural network aided design for metering system of power system state estimation," *Proceedings of IEEE. AFRICON '96*, Stellenbosch, South Africa, 1996, pp. 607-610 vol.2.

[51] T. Tian, M. Zhu and B. Zhang, "An artificial neural network-based expert system for network topological error identification," *Proceedings of ICNN'95 - International Conference on Neural Networks*, Perth, WA, Australia, 1995, pp. 882-886 vol.2.

[52] M. Ferdowsi, B. Zargar, F. Ponci and A. Monti, "Design considerations for artificial neural network-based estimators in monitoring of distribution systems," *2014 IEEE International Workshop on Applied Measurements for Power Systems Proceedings (AMPS)*, Aachen, Germany, 2014, pp. 1-6.

Appendix A

Line Data parameters

From bus	To bus	R (pu)	X (pu)	B/2 (pu)
1	2	0.0192	0.0575	0.0264
1	3	0.0452	0.1652	0.0204
2	4	0.057	0.1737	0.0184
3	4	0.0132	0.0379	0.0042
2	5	0.0472	0.1983	0.0209
2	6	0.0581	0.1763	0.0187
4	6	0.0119	0.0414	0.0045
5	7	0.046	0.116	0.0102
6	7	0.0267	0.082	0.0085
6	8	0.012	0.042	0.0045
6	9	0	0.208	0
6	10	0	0.556	0
9	11	0	0.208	0
9	10	0	0.11	0
4	12	0	0.256	0
12	13	0	0.14	0
12	14	0.1231	0.2559	0
12	15	0.0662	0.1304	0
12	16	0.0945	0.1987	0
14	15	0.221	0.1997	0
16	17	0.0824	0.1923	0
15	18	0.1073	0.2185	0
18	19	0.0639	0.1292	0
19	20	0.034	0.068	0
10	20	0.0936	0.209	0
10	17	0.0324	0.0845	0
10	21	0.0348	0.0749	0
10	22	0.0727	0.1499	0
21	23	0.0116	0.0236	0
15	23	0.1	0.202	0
22	24	0.115	0.179	0
23	24	0.132	0.27	0
24	25	0.1885	0.3292	0
25	26	0.2544	0.38	0
25	27	0.1093	0.2087	0
28	27	0	0.396	0
27	29	0.2198	0.4153	0
27	30	0.3202	0.6027	0
29	30	0.2399	0.4533	0
8	28	0.0636	0.2	0.0214
6	28	0.0169	0.0599	0.065

Appendix B

IEEE-30 bus system data

Bus	Vsp (pu)	Theta	PGi (pu)	QGi (pu)	PLi (pu)	QLi (pu)	Qmin (pu)	Qmax (pu)
1	1.06	0	0	0	0	0	0	0
2	1.043	0	0.622	0.5	0.217	0	-0.4	0.50
3	1	0	1.2	0	0.024	0	0	0
4	1.06	0	0	0	0.076	0	0	0
5	1.01	0	0.344	0.37	0.942	0	-0.4	0.40
6	1	0	0	0	0	0	0	0
7	1	0	0	0	0.228	0	0	0
8	1.01	0	0.242	0.373	0.3	0	-0.1	0.40
9	1	0	0	0	0	0	0	0
10	1	0	0	0	0.058	0	0	0
11	1.082	0	0.216	0.162	0	0	-0.06	0.24
12	1	0	0	0	0.112	0	0	0
13	1.071	0	0.28	0.106	0	0	-0.06	0.24
14	1	0	0	0	0.062	0	0	0
15	1	0	0	0	0.082	0	0	0
16	1	0	0	0	0.035	0	0	0
17	1	0	0	0	0.09	0	0	0
18	1	0	0	0	0.032	0	0	0
19	1	0	0	0	0.095	0	0	0
20	1	0	0	0	0.022	0	0	0
21	1	0	0	0	0.175	0	0	0
22	1	0	0	0	0	0	0	0
23	1	0	0	0	0.032	0	0	0
24	1	0	0	0	0.087	0	0	0
25	1	0	0	0	0	0	0	0
26	1	0	0	0	0.035	0	0	0
27	1	0	0	0	0	0	0	0
28	1	0	0	0	0	0	0	0
29	1	0	0	0	0.024	0	0	0
30	1	0	0	0	0.106	0	0	0

Appendix C

Healthy measurement against Corrupted measurement

Type of measurement	Meter #	Healthy measurement value	Corrupted measurement value
Voltage	1	1.06	1.06
Real power injection	2	-0.076	-0.076
	3	-0.942	-0.942
	4	0	0
	5	-0.3	-0.3
	6	-0.058	-0.058
	7	0	0
	8	0	0
	9	-0.0621	-0.0621
	10	-0.0819	-0.0819
	11	-0.035	-0.035
	12	-0.032	0.789194
	13	-0.022	-0.022
	14	-0.175	-0.175
	15	-0.087	-0.087
	16	0	0
	17	-0.035	-0.035
	18	0	0
	19	-0.024	-0.024
Reactive power injection	20	-0.016	-0.016
	21	0.1538	0.1538
	22	0	0
	23	0.2096	0.2096
	24	-0.02	-0.02
	25	0.2066	0.2066
	26	0.1483	0.1483
	27	-0.016	-0.016
	28	-0.025	-0.025
	29	-0.018	-0.018
	30	-0.009	-0.009
	31	-0.007	-0.007
	32	-0.112	-0.112
	33	-0.067	-0.067
	34	0	0
	35	-0.023	-0.023
	36	0	0
	37	-0.009	-0.009
Real power	38	0.4377	0.4377

Type of measurement	Meter #	Healthy measurement value	Corrupted measurement value
flow	39	0.8213	0.8213
	40	-0.8487	-0.8487
	41	-0.816	-0.816
	42	0.7161	0.7161
	43	-0.1475	-0.1475
	44	-0.5836	-0.5836
	45	-0.3772	-0.3772
	46	-0.2705	-0.2705
	47	-0.1539	-0.1539
	48	-0.2705	-0.2705
	49	0.081	0.081
	50	-0.1786	-0.1786
	51	0.0623	0.0623
	52	0.0423	0.0423
	53	-0.048	-0.048
	54	-0.0623	-0.0623
	55	-0.0844	-0.0844
	56	-0.176	-0.176
	57	0	0
	58	-0.0554	-0.0554
	59	0.0177	0.0177
	60	-0.0549	-0.0549
	61	-0.05	-0.05
	62	-0.1832	-0.1832
	63	-0.1883	-0.1883
	64	0.037	0.037
	65	-0.0693	-0.0693
Reactive power flow	66	0.0451	0.0451
	67	0.0402	0.0402
	68	0.0577	0.0577
	69	0.0687	0.0687
	70	-0.2202	-0.2202
	71	0.1027	0.1027
	72	0.0624	0.0624
	73	0.0089	0.0089
	74	0.1463	0.1463
	75	0.0244	0.0244
	76	-0.1599	0.07488
	77	0.0327	0.0327
	78	-0.0977	-0.0977
	79	0.031	0.031
	80	0.042	0.042
	81	-0.0167	-0.0167

Type of measurement	Meter #	Healthy measurement value	Corrupted measurement value
	82	-0.0133	-0.0133
	83	-0.0205	-0.0205
	84	-0.0973	0.255859
	85	0	0
	86	-0.0361	-0.0361
	87	0.0248	0.0248
	88	-0.0353	-0.0353
	89	-0.0309	-0.0309
	90	-0.0219	-0.0219
	91	0.0399	0.0399
	92	0.0061	0.0061
	93	-0.0136	-0.0136

Appendix D

Measurement vectors of Healthy and corrupted vectors (four meters)

Bus	Corrupted measurement vector		True measurement vector		Deviation of Voltage from the true value	Deviation of phase angle from the true value
	Voltage (p.u)	Phase angle (degree)	Voltage (p.u)	Phase angle (degree)		
1	0.873902	0	0.986606	0	0.112704	0
2	0.844335	-8.24539	0.9701	-6.26262	0.125765	1.982766
3	0.82357	-11.5015	0.947483	-8.84065	0.123913	2.660891
4	0.814226	-14.1981	0.938487	-10.9005	0.124261	3.297572
5	0.803442	-21.8339	0.933599	-16.4916	0.130157	5.34234
6	0.815616	-16.9554	0.939607	-12.9955	0.123991	3.959875
7	0.799766	-19.8782	0.928822	-15.042	0.129056	4.83628
8	0.819791	-18.3141	0.944996	-13.9586	0.125205	4.355491
9	0.845324	-21.8173	0.966768	-16.4787	0.121445	5.338559
10	0.82357	-24.1806	0.947249	-18.3417	0.123679	5.838875
11	0.893421	-21.8173	1.009343	-16.4787	0.115922	5.338559
12	0.862609	-21.8297	0.974636	-17.689	0.112026	4.140771
13	0.886041	-21.8297	0.995492	-17.689	0.109451	4.140771
14	0.842421	-23.0443	0.955965	-18.7107	0.113544	4.333531
15	0.835461	-23.0321	0.949162	-18.727	0.113701	4.305103
16	0.835715	-23.5301	0.955621	-18.2771	0.119906	5.252961
17	0.820452	-24.4449	0.944137	-18.5685	0.123685	5.876316
18	0.831758	-22.1989	0.935246	-19.4165	0.103487	2.782452
19	0.798509	-26.8442	0.930664	-19.6032	0.132154	7.240949
20	0.803335	-26.3359	0.933951	-19.3551	0.130617	6.980816
21	0.807877	-25.0035	0.93283	-18.9791	0.124953	6.024428
22	0.817024	-24.1978	0.937229	-18.7082	0.120205	5.48957
23	0.808687	-25.0099	0.933186	-18.9927	0.1245	6.017221
24	0.825112	-21.9623	0.923167	-19.0758	0.098055	2.886473
25	0.975079	-9.6089	0.927094	-18.7756	0.047985	9.166694
26	1.11198	1.604819	0.9071	-19.2564	0.204881	17.6516
27	1.013506	-9.13707	0.939566	-18.2935	0.07394	9.156404
28	0.828465	-17.3431	0.939914	-13.7889	0.111449	3.554199
29	0.993448	-10.3909	0.917726	-19.7575	0.075722	9.366559
30	0.981847	-11.2908	0.90514	-20.8141	0.076707	9.52326

Appendix E

Measurement vectors of Healthy and corrupted vectors (three meters)

Bus	Corrupted measurement vector		True measurement vector		Deviation of Voltage from the true value	Deviation of phase angle from the true value
	Voltage (p.u)	Phase angle (degree)	Voltage (p.u)	Phase angle (degree)		
1	1.042114	0	0.986606	0	0.055508	0
2	1.026545	-5.58124	0.9701	-6.26262	0.056445	0.681375
3	1.00702	-7.87334	0.947483	-8.84065	0.059538	0.967304
4	0.999957	-9.67244	0.938487	-10.9005	0.061469	1.228072
5	0.993093	-14.6603	0.933599	-16.4916	0.059494	1.831272
6	0.999502	-11.5646	0.939607	-12.9955	0.059895	1.430945
7	0.988671	-13.3674	0.928822	-15.042	0.059849	1.674537
8	1.004517	-12.4126	0.944996	-13.9586	0.059521	1.546009
9	1.016964	-14.8496	0.966768	-16.4787	0.050196	1.629128
10	1.024053	-16.5591	0.947249	-18.3417	0.076804	1.782548
11	1.057596	-14.8496	1.009343	-16.4787	0.048253	1.629128
12	1.065636	-14.45	0.974636	-17.689	0.091001	3.239003
13	1.084776	-14.45	0.995492	-17.689	0.089284	3.239003
14	1.050275	-15.1858	0.955965	-18.7107	0.09431	3.524892
15	1.047386	-15.3467	0.949162	-18.727	0.098224	3.380281
16	1.036407	-15.8773	0.955621	-18.2771	0.080786	2.399806
17	1.02029	-16.6517	0.944137	-18.5685	0.076152	1.916822
18	1.043305	-14.1118	0.935246	-19.4165	0.108059	5.304693
19	1.000258	-18.4492	0.930664	-19.6032	0.069594	1.154024
20	1.0046	-18.0852	0.933951	-19.3551	0.070649	1.269865
21	1.033033	-17.6128	0.93283	-18.9791	0.100204	1.366331
22	1.017185	-16.8913	0.937229	-18.7082	0.079956	1.816881
23	1.040053	-17.7842	0.933186	-18.9927	0.106867	1.208489
24	1.010767	-17.4028	0.923167	-19.0758	0.0876	1.672968
25	0.9974	-16.782	0.927094	-18.7756	0.070305	1.9936
26	0.972575	-17.03	0.9071	-19.2564	0.065475	2.226456
27	1.004761	-16.2882	0.939566	-18.2935	0.065196	2.00531
28	1.000203	-12.2739	0.939914	-13.7889	0.060289	1.515008
29	0.984508	-17.5645	0.917726	-19.7575	0.066782	2.192974
30	0.972799	-18.4811	0.90514	-20.8141	0.067659	2.332996

Appendix F

Measurement vectors of Healthy and corrupted vectors (three meters)

Bus	Corrupted measurement vector		True measurement vector		Deviation of Voltage from the true value	Deviation of phase angle from the true value
	Voltage (p.u)	Phase angle (degree)	Voltage (p.u)	Phase angle (degree)		
1	0.988265	0	0.986606	0	0.001659	0
2	0.971319	-6.26148	0.9701	-6.26262	0.001219	0.001144
3	0.949276	-8.8087	0.947483	-8.84065	0.001793	0.031949
4	0.940701	-10.8426	0.938487	-10.9005	0.002214	0.057878
5	0.935133	-16.4485	0.933599	-16.4916	0.001534	0.043036
6	0.941368	-12.9546	0.939607	-12.9955	0.001761	0.040887
7	0.930484	-14.9976	0.928822	-15.042	0.001662	0.044382
8	0.946642	-13.9206	0.944996	-13.9586	0.001647	0.037986
9	0.969615	-16.3255	0.966768	-16.4787	0.002847	0.15317
10	0.950409	-18.1718	0.947249	-18.3417	0.00316	0.169902
11	1.012075	-16.3255	1.009343	-16.4787	0.002732	0.15317
12	0.982574	-16.9635	0.974636	-17.689	0.007938	0.725475
13	1.003268	-16.9635	0.995492	-17.689	0.007776	0.725475
14	0.982818	-16.3776	0.955965	-18.7107	0.026852	2.33318
15	0.956741	-17.8601	0.949162	-18.727	0.007579	0.866856
16	0.960522	-17.8837	0.955621	-18.2771	0.004901	0.393397
17	0.947626	-18.3625	0.944137	-18.5685	0.003489	0.206044
18	0.94051	-18.9148	0.935246	-19.4165	0.005264	0.501658
19	0.9345	-19.3512	0.930664	-19.6032	0.003836	0.252041
20	0.937691	-19.1158	0.933951	-19.3551	0.003739	0.239323
21	0.937994	-18.6064	0.93283	-18.9791	0.005164	0.372737
22	0.935628	-19.0668	0.937229	-18.7082	0.001602	0.358609
23	0.939179	-18.5348	0.933186	-18.9927	0.005992	0.457859
24	0.919004	-19.7886	0.923167	-19.0758	0.004163	0.712776
25	0.918914	-19.7847	0.927094	-18.7756	0.00818	1.009146
26	0.893944	-20.673	0.9071	-19.2564	0.013155	1.416564
27	0.933115	-19.0692	0.939566	-18.2935	0.006451	0.775741
28	0.940774	-13.7944	0.939914	-13.7889	0.00086	0.005513
29	0.911105	-20.5539	0.917726	-19.7575	0.006622	0.796427
30	0.898424	-21.626	0.90514	-20.8141	0.006716	0.811927

Vita

Leen Al Halabi was born in Damascus, Syria. She received her primary and secondary education in Khorfakkan, UAE. She received her B.Sc. degree in Electronics Engineering from Khalifa University in 2016. During her bachelors study, she co-authored two papers which were presented in international conferences.

In January 2016, she joined the Mechatronics Engineering master's program in the American University of Sharjah as a graduate teaching assistant. Her research interests are in the area of power system’s security and Neural Networks.