# STATE ESTIMATION
# OF
# POWER SYSTEMS

By

SHAZLINA BINTI MOHAMAD SHUIB

FINAL REPORT

Submitted to the Electrical & Electronics Engineering Programme
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

# CERTIFICATION OF APPROVAL

## STATE ESTIMATION
## OF
## POWER SYSTEMS

by

Shazlina Binti Mohamad Shuib

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

Dr. Ramiah Jegatheesan,
Project Supervisor

DR. RAMIAH JEGATHEESAN
B.E. (Hons), M.Sc. (Engg.), Ph.D.
Professor, EE Programme
UNIVERSITI TEKNOLOGI PETRONAS
Bandar Seri Iskandar,
31750 Tronoh, Perak, Malaysia.

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

June 2007

# CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

Shazlina Binti Mohamad Shuib

# ABSTRACT

State Estimation of Power System has same function as Power Flow Analysis in determining system status which is very important in operating the power systems securely. Detail understanding on the differences between State Estimation and Power Flow Analysis has to be taken into consideration. The focus will be on testing the Power System State Estimation either by using existing software or developed software. The chosen option is to produce software using MATLAB program that can test the State Estimation in Power System network and in obtaining consistent system states which are magnitude, $|V|$, and phase angles, $\delta$, of bus voltages. Weight Least Squares Method is used to calculate estimates of the state variables (unknown data) by using measured data. When bad measurements are detected, the estimated states are not reliable anymore. So, the data have to be identified and discarded by statistical tests. The presence of bad data is assumed due to improper connection with reversed positive and negative leads at the meter equipment. The limitations that Power Flow Analysis has can be removed by state estimation based on the weighted least squares method and also the method of detecting and eliminating the bad data.

# ACKNOWLEDGEMENTS

First and foremost, the author would like to praise Allah the Almighty for giving the blessing and strength to be able to complete her final year project successfully.

The author's deepest appreciation especially goes to her supervisor, Dr. Ramiah Jegatheesan for his immeasurable support and enthusiasm in giving guidance. The author's utmost gratitude also goes to Mr. Nursyarizal Mohd Nor for the priceless assistance and advice throughout the completion of this Final Year Project. Special thanks to the FYP coordinators for preparing a very systematic schedule to complete the project and also to Ms Siti Hawa for her sincere assistance and guidance in complying with the procedures for submission of all reports.

Most of all, a sincere appreciation is especially dedicated to the author's beloved family, her parents, Haji Mohamad Shuib Abdul Ghani and Hajjah Fuziah Murat for their encouragement and tolerance during her final year project period. Last but not least, the author is also grateful to her best friend, Khalid Ibrahim, who has always been supportive and understanding for the whole time the project was being undertaken. Finally, the author's heartfelt gratitude goes to everyone who has shared their ideas, experiences and encouragement.

Thank you.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS

1. WLS : Weighted Least-Squares
2. PET : Power Education Toolbox
3. rad. : radian
4. p.u. : per unit
5. Nm : Number of Measurement
6. Ns : Number of State Variable
7. PSAT : Power System Analysis Toolbox

# CHAPTER 1

# INTRODUCTION

As the project entitled *'State Estimation of Power System'*, the focus of this Final Year Project would obviously lie on the concept of the topic. Moreover, understandings on the fundamental of Power Flow Solutions are also being taken into consideration. In the first subtopic, Background of Study, there will be an elaboration on the basic concept of the State Estimation of Power System. Then, a brief discussion on the difference between State Estimation and Power Flow Analysis will be explained in the second section, Problem Statement. Later, there will be a complete list of goals and aims of the project, in the subtopic of Objectives and Scope of Study.

## 1.1 Background of Study

The status of power system that is required for a number of application programs such as economic dispatch and load frequency control as well as security assessment can be determined by state estimation. This method uses large amount of measured data to estimate the system status. So, error or unavailability of the measurements might occur which will lead to malfunction in the outputs. Hence, the results obtained can be accepted as estimated of true values if no bad data are present. However, if there is any bad data present, this must be detected and removed from the estimation calculations. The State Estimation use large amount of measured data to estimate the system state of the voltage magnitude, $|V|$ and phase angle, $\delta$ at bus voltages rather than Power Flow Analysis.

Moreover, monitoring the generation and transmission of very complex power system networks has made the task of operating the system more difficult. There is a condition where in avoiding major system failures, more extensive supervisory control and data acquisition are installed by the electric utilities throughout the network with the purpose of supporting computer-based systems at the energy control center. The data created is for the used of the application programs such as to ensure the economic system operation and others to asses how secure the system would be if equipment failures and transmission-line outages were to occur. [1]

The method of the State Estimation can provide a real-time estimate of the system state and can determine of how good the estimate is before it is used for real-time power-flow calculations or on-line system security assessment. The estimates of the states have to be tested to statistical testing before being produced as the best possible estimates and accepted as satisfactory. Consequently, if there is error detected in the state estimation it has to be filtered out. [1]

## 1.2 Problem Statement

For the Security System reason, either the occurrence of equipment failures or transmission-line outages, a security assessment has to be done in determining a reliable estimate of the existing state of the system. State variables or unknown quantities that have to be estimated are the voltage magnitudes and also the phase angles of all the buses but except for the slack bus. Due to this, the number of measurements cannot be limited to only quantities required for the conventional Power Flow Solution. However, if one of the inputs or measurements which are the P and Q injections at load buses and also P and |V| values at voltage-controlled buses, is unavailable, the conventional Power Flow Solution cannot be gained. In addition, if one or more errors occurred at the inputs or measurements, the results of the Power Flow Solutions will become insufficient.

Thus, in removing the limitations, the State Estimation can be done by using the Weighted Least-Squares calculations, where the numbers of inputs or measurements are larger than the number of unknown quantities or state variables. Furthermore, there is an additional to the measurements which are very easily measured quantities that can be used in the State Estimation but cannot be used in conventional Power Flow Solutions. They are the measurements at P and Q line flows in the transmission lines of the system. Subsequently, the unknown quantities or state variables are estimated and if there is any bad data present, it can be detected and discarded. [1]

The project has been chosen after considering the relevancy and significance of the project. This project has been proposed in order to give opportunities to students to develop and simulate a software themselves and test whether it can work as required which is in the practical Power Systems. There are certain reasons in initiating this study about the State Estimation of Power System which are:

- Data contains inaccuracies or errors since physical measurements
- Power Flow Analysis cannot be obtained if there is error or one or more inputs are unavailable
- Measured data do not always tally with the calculated values when using existing software (PET Software) then converting to MATLAB software

The main purpose of simulating the State Estimation either by using existing softwares or developed software, which is by using the MATLAB program, is to avoid redundancy in the data gathering systems. This is due to the complexity and larger data base that causes the number of actual measurements is greater than the number of data inputs. This leads to more equations to be solved than the unknown state variables. Hence, data filtration is very important and required before the raw data are used in state estimation computation and direct use of raw measurement is not advisable.

During the early stage, most of the learning is done on the difference between State Estimation and Power Flow Analysis. Both methods have the same purpose which is to find the bus voltages of a Power Flow System. Consequently, the task given is about understanding both basic concepts, in order to achieve the main objective, which is to produce the software for the Power System State Estimation.

For the second semester, more attempts have been made in finding the most suitable software that can be used in solving the State Estimation of Power System. In the previous semester, based on the methodology, the next procedure should have been to develop MATLAB software for the State Estimation of Power System. However, for certain reasons, the procedure has to be changed where the software will not be developed, but to apply the existing software on power system. The reason is that the existing software is much more easily understood and applicable when compared to producing a completely new MATLAB programming. Furthermore, there are several softwares in the industry that can be used in applying the State Estimation.

Hence, in order to find the most relevant and user-friendly software, several steps had been taken. The preferred softwares are SKM Power Tools and Power Education Toolbox where both can be applied in finding the results of State Estimation of Power System. Power Education Toolbox Software is found that to be very user-friendly compared to SKM Power Tools. So, more attempts have been conducted in applying the software on the networks. Additionally, extra efforts and more concentration have been given in learning how to use this software.

The main purpose of proceeding with this project till this level is to ensure that the software used must be able to produce the final results consistently. Furthermore, after simulating the State Estimation using the Weighted Least Squares for several numbers of buses in the power system network, the outcomes will be the magnitude and phase angle of the voltage at each bus. In addition, the final results must also tally with the measured values taken earlier before the simulation can take place.

The measurements taken are either from the $P_i$, $|V|_i$ values at the voltage-controlled buses or from the $P_i$, $Q_i$ injections at load buses. There are also measured quantities taken from the line real and reactive power flows of the power system networks which are $P_{ij}$ and $Q_{ij}$. The final results of the voltage values at each bus can be compared with the measurements taken for voltage magnitude at each bus by using voltmeter if there were any. While, the final voltage quantities between buses obtained can be calculated in ensuring that the measured power flow values at the line flows must be the same as the calculated values. The concept of the State Estimation of Power System studied earlier is then tested by using developed software.

However, after several trials on the PET Software, the earlier idea on the reliability of using the existing software which is PET software is untrue. This is because during the testing on the 2-bus, 3-bus, 4-bus and 14-bus system, the software could not obtain the results consistently. This is due to the number of iterations that keep on accumulating each time the State Estimation on the bus system is run. Consequently, the main objectives have been changed to develop a new software using MATLAB for the State Estimation of Power System which must produce consistent results that are the magnitude, |V|, and phase angles, δ, of the bus voltages. In addition, the results obtained, must tally with the measurements in the provided input data, which are P and Q injections at load buses, P and |V| values at voltage-controlled buses and also P and Q line flows in the transmission lines of the system.

## 1.3 Objectives & Scope of Study

In any project, objectives have to be clearly identified so that targets can be set and achieved by the end of the given period. Thus, the objectives of the project will be:-

- ✓ To apply the existing software or developed software for Power System State Estimation.
- ✓ To understand the software that has to be used in this project
- ✓ To prove that Power system parameters calculation using state estimation method is better than the conventional power flow analysis, implemented in MATLAB

Moving on to the scope of study, several aspects have to be focused so that the project can be completed appropriately. The aspects will be on;

- ✓ Understanding the differences between State Estimation and Power Flow Analysis.
- ✓ Doing research to learn the basic concept of State Estimation of Power System.
  - Method of Least Squares
  - Test for Bad Data

- ✓ The general practice in power-flow studies is identification of three types of buses in the network:
  - Load Buses, P-Q Buses
  - Voltage-Controlled Buses, P-V Buses
  - Slack Buses

- ✓ The method used in Power Flow Solutions
    - Newton-Raphson Method

- ✓ The application of the Newton-Raphson Power Flow Solution.
- ✓ The Power Flow Analysis in System Design and Operation.
- ✓ Learning about MATLAB Programming in order to prepare the coding for the software of State Estimation of Power System.
- ✓ Exploring and learning about other Programming level that might be used for this project. For example, Visual Basic.

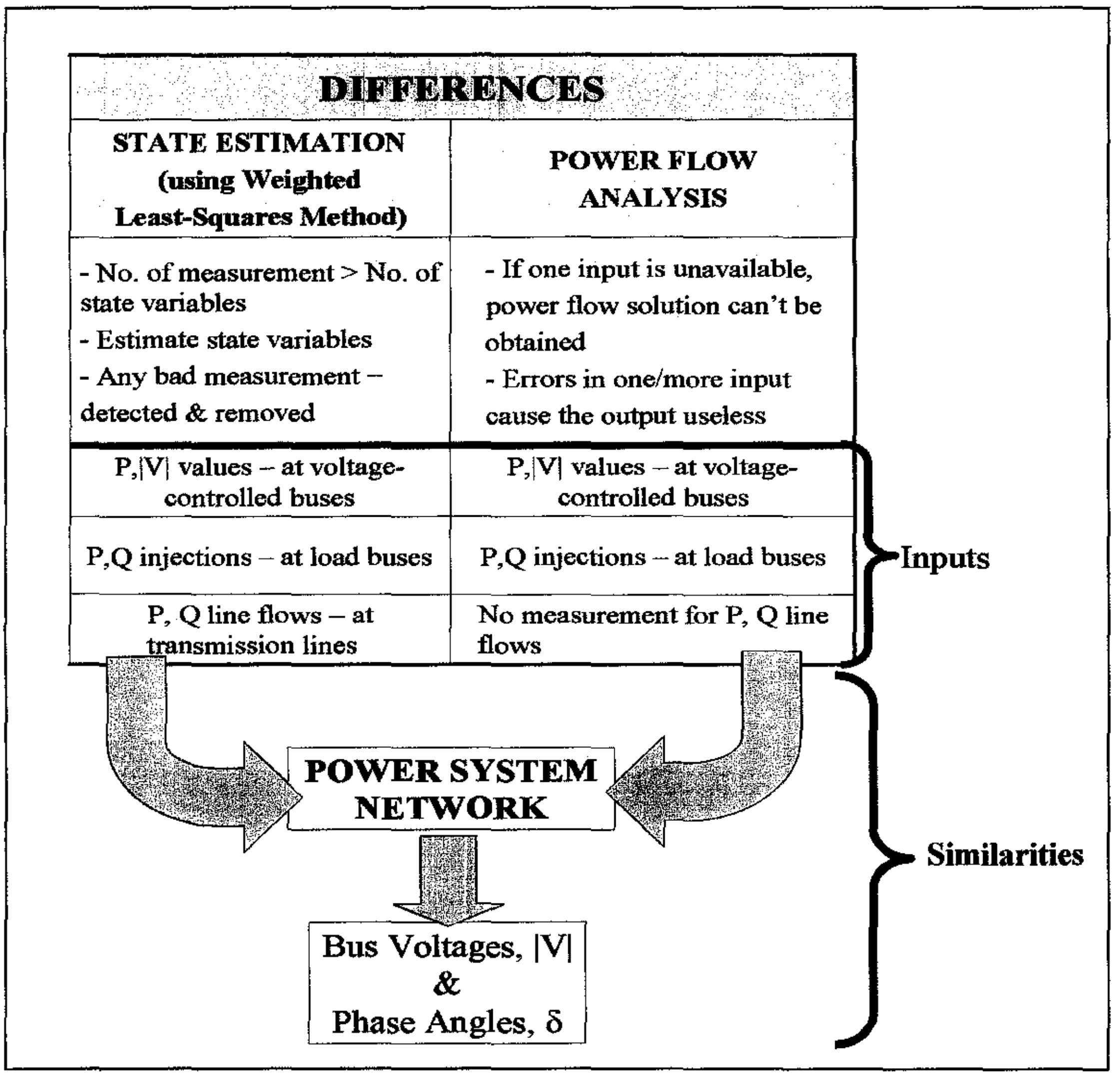
All of these studies are done within the duration given. At the end of the Final Year Project II period, the deliverable should be a software that can be used in order to perform the State Estimation of Power System by using Weight Least Squares method and also in testing and discarding of the bad data.

# CHAPTER 2

# LITERATURE REVIEW AND THEORY

Literature review is very important in building a strong foundation before the project can be continued further. In the early stage which is during the Final Year Project 1, a few important topics have been studied in order to gain knowledge on issues related to the project.

## 2.1 The Similarities and Differences between Power Flow Analysis and State Estimation



| DIFFERENCES | |
|---|---|
| **STATE ESTIMATION (using Weighted Least-Squares Method)** | **POWER FLOW ANALYSIS** |
| - No. of measurement > No. of state variables<br>- Estimate state variables<br>- Any bad measurement – detected & removed | - If one input is unavailable, power flow solution can't be obtained<br>- Errors in one/more input cause the output useless |
| P,\|V\| values – at voltage-controlled buses | P,\|V\| values – at voltage-controlled buses |
| P,Q injections – at load buses | P,Q injections – at load buses |
| P, Q line flows – at transmission lines | No measurement for P, Q line flows |

Figure 1.1 Similarities and differences between State Estimation and Power Flow Analysis

## 2.2 The Power Flow Analysis

For the Power Flow Solutions, the equations of functions P and Q are as shown below:

$$\boxed{P_i = \sum_{n=1}^{N} |V_i||V_n||Y_{in}|\cos(\theta_{in} + \delta_n - \delta_i)}$$

$$\boxed{Q_i = -\sum_{n=1}^{N} |V_i||V_n||Y_{in}|\sin(\theta_{in} + \delta_n - \delta_i)}$$

As can be seen, they are non-linear functions of the unknown quantities or state variables which are the magnitude, $|V|$, and phase angles, $\delta$, of the bus voltages. Consequently, these equations are used in the State Estimation of Power System for this Final Year Project topic. This is due to the calculation involves is the ac Power System Network which will produce non-linear equations. Hence, power flow calculation such as the Newton-Raphson Method which usually employs iterative techniques is chosen in linearizing the non-linear equations. This method is based on bus admittance equations. This method will be described briefly in Chapter 4, Results and Discussion which is applied in the 2-bus and 3-bus system.

Here, the differences between State Estimation and Power Flow Analysis can be seen when the Load Flow Analysis which employs the Newton Raphson Method in solving the power flow problem. From the observation of the calculation for Newton-Raphson Method, the following is the solution procedure for the method of Power Flow Analysis. Firstly, the line data and bus data are read in order to construct the bus admittance matrix. Then, setting $k = 0$, a starting solution is assumed. Usually, a flat start is assumed in which all the unknown phase angles are taken as zero and unknown voltage magnitudes are taken as 1.0 p.u. Next, the mismatch powers, for example the error vector, are computed. If the elements of error vector were less than the specified tolerance, the problem is solved. Then, proceed to the step where the line flows, transmission loss and slack bus power are calculated.

Otherwise, proceed to the step where the elements of sub-matrices H, N, M and L are computed and the equation below is solved as follows:-

$$\boxed{\begin{bmatrix} H & N \\ M & L \end{bmatrix}_k \begin{bmatrix} \Delta\delta \\ \dfrac{\Delta|V|}{|V|} \end{bmatrix} = \begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix}_k \quad \text{for} \quad \begin{bmatrix} \Delta\delta \\ \dfrac{\Delta|V|}{|V|} \end{bmatrix}}$$

Then, the solution is updated as below:

$$\boxed{\begin{bmatrix} \delta \\ |V| \end{bmatrix}_{k+1} = \begin{bmatrix} \delta \\ |V| \end{bmatrix}_k + \begin{bmatrix} \Delta\delta \\ \Delta|V| \end{bmatrix}}$$

Furthermore, the new value of $k$ is set to be one value higher, with $k + 1$. Consequently, the step where the calculation of the mismatch powers is repeated all over again until the elements of error vector are specified to be less than the tolerance in order to ensure that the problem is solved. Finally, when the results are obtained the line flow transmission loss and slack bus power can be calculated. The outcomes of this method are similar to the end result of the State Estimation which is in determining the magnitude, $|V|$, and phase angles, $\delta$, of the bus voltages.

Electric utility such as Tenaga Nasional Berhad (TNB), conducts power flow analysis quite often with the purpose of updating the status of the power system for different load conditions. This is due to ensure the economic system operation and also for the system security assessment. In addition, the network data and bus power data are supplied as input data. Then, the ac power system generates a set of non-linear equations. Furthermore, the bus voltages are calculated for the given network configuration and bus power injections.

## 2.3 The Method of Least Squares

In the State Estimation, the Weighted Least-Squares method is used since the number of inputs or measurements are larger than the number of unknown quantities or state variables. Additionally, this method is used as it is the best method to fit the inputs or measurements as it relates two or more quantities. Thus, the procedure in obtaining the most possible estimates of the unknown quantities or state variables must be taken into consideration. For easy understanding, this method is applied to a set of dc circuit that has measurements which is shown in Section 3.1. Then, in applying the State Estimation procedures which consists of Weighted Least-Squares method and testing as well as removing the bad data, it can be shown in the Power System Network in Section 4. From the results obtained, that the estimated values are those which minimize the weighted sum of the squares of the measurement errors.

## 2.4 Existing Softwares

After going through the final presentation for FYP I, a suggestion has been made by one of the evaluators where the State Estimation of Power System which initially has to be tested on a developed software using MATLAB program should be replaced with any existing softwares such as SKM Power Tools or Power Education Toolbox. It is due to the softwares are believed to be reliable and user friendly. The softwares are studied for the purpose of proving their reliability. In addition the study is performed in order to familiarize with their functions.

There is a function in simulating the State Estimation of Power System in the SKM Power Tools Software. However, a problem occurred in locating the measurements in the power system network, which could not be solved. Even though consultation has been done to the software's distributor, the problem still could not be solved. So, after referring to the Supervisor, Dr. Ramiah Jegatheesan, the learning process of the software is continued with another software which is the Power Education Toolbox (PET) Software which is supposed to be a very interesting and its usage easily understood. This software is also believed to be a very user-friendly software which is used as a teaching tool for power engineering.

Though, there is certain circumstance that could not be avoided that has stopped the process of applying this software. Initially, the simulation runs smoothly for the first run. However, the problem arise when it comes to the next run, where the final results which are the magnitude, $|V|$, and phase angles, $\delta$, of the bus voltages could not be achieved consistently. The results keep generating different results when each time the simulation is run. So, in order to complete the project on a timely basis and again after consultation with the Supervisor, Dr. Ramiah Jegatheesan, the testing of the State Estimation of Power System on the existing softwares has to be replaced with the initial method which is to be tested on a developed software by using MATLAB program. Finally, after developing the coding for the State Estimation of Power System using MATLAB program, it is proven that testing the State Estimation through developed software can produce consistent results compared to testing the State Estimation on existing softwares. In addition, the results tallied with the provided inputs or measurements and also the bad data can be detected as well as discarded.

# CHAPTER 3
# METHODOLOGY / PROJECT WORK

All steps and procedures to be taken have to be thoroughly clarified, in order to ensure the smoothness of the project completion work. This methodology section will briefly show the pre-determined track.

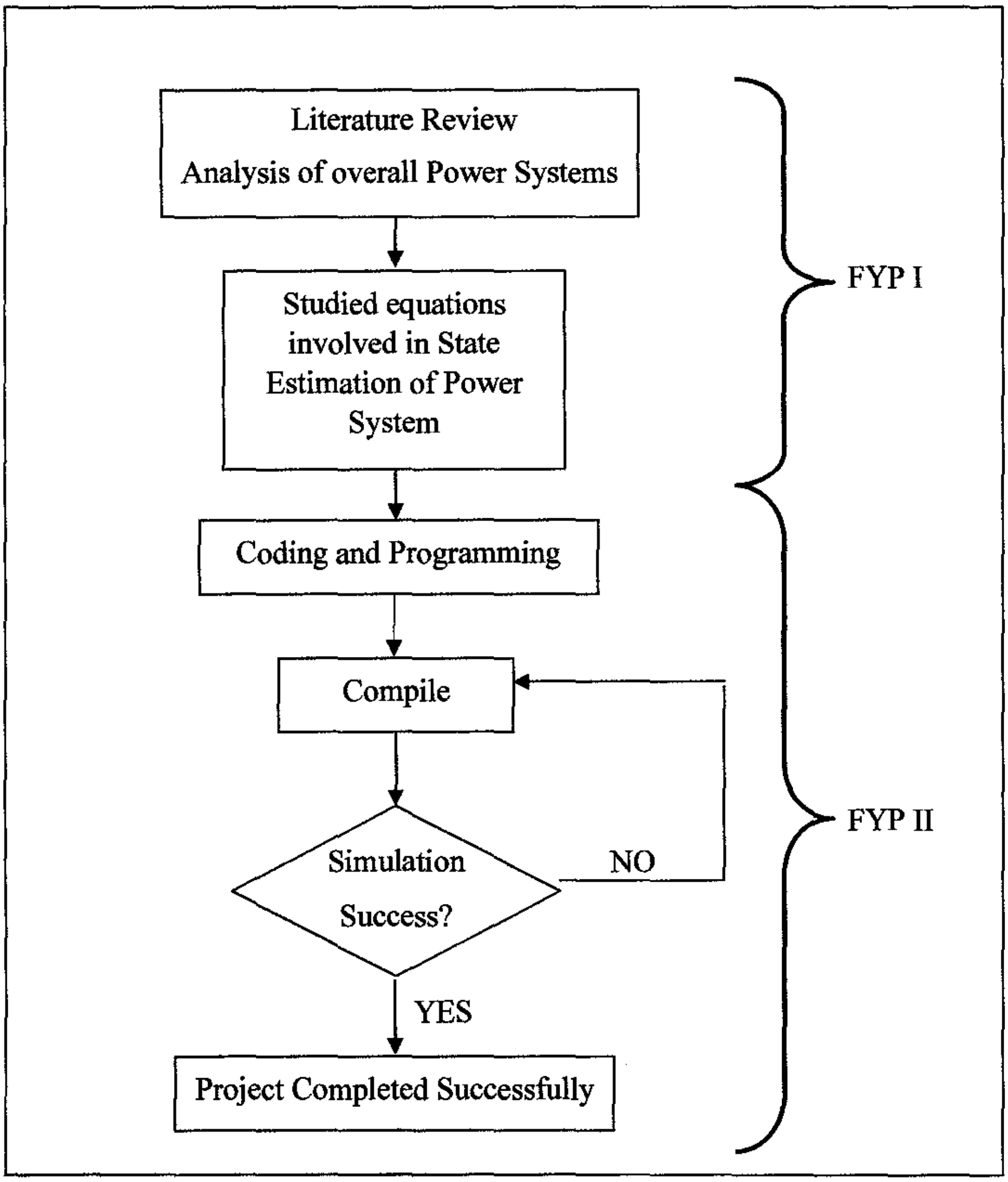


Figure 3.1 Flow Chart of the Methodology

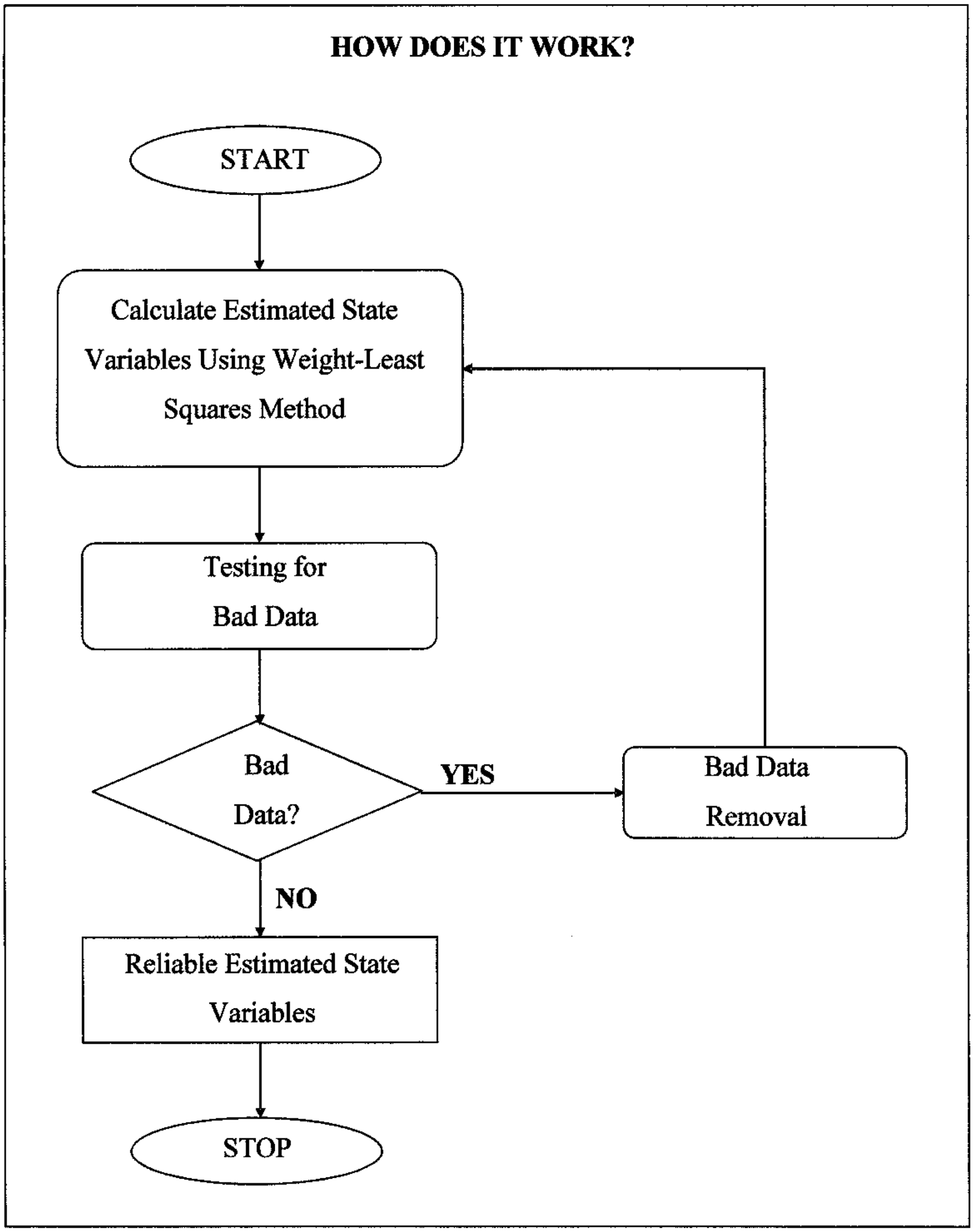
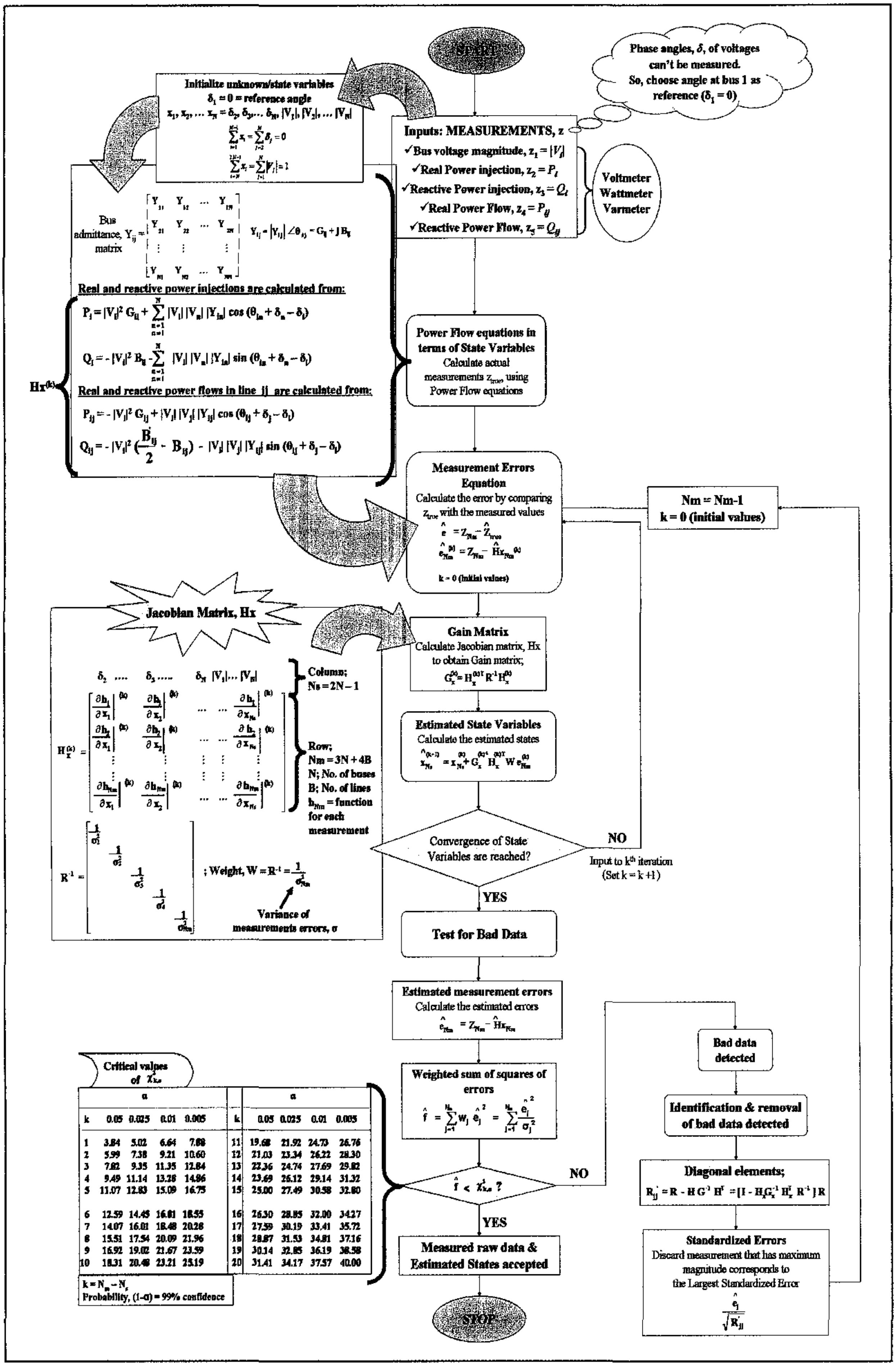


Figure 3.2 Flow Chart of the State Estimation Procedure Generally



|  | α |  |  |  | | α |  |  |  |
|---|---|---|---|---|---|---|---|---|---|
| k | 0.05 | 0.025 | 0.01 | 0.005 | k | 0.05 | 0.025 | 0.01 | 0.005 |
| 1 | 3.84 | 5.02 | 6.64 | 7.88 | 11 | 19.68 | 21.92 | 24.73 | 26.76 |
| 2 | 5.99 | 7.38 | 9.21 | 10.60 | 12 | 21.03 | 23.34 | 26.22 | 28.30 |
| 3 | 7.82 | 9.35 | 11.35 | 12.84 | 13 | 22.36 | 24.74 | 27.69 | 29.82 |
| 4 | 9.49 | 11.14 | 13.28 | 14.86 | 14 | 23.69 | 26.12 | 29.14 | 31.32 |
| 5 | 11.07 | 12.83 | 15.09 | 16.75 | 15 | 25.00 | 27.49 | 30.58 | 32.80 |
| 6 | 12.59 | 14.45 | 16.81 | 18.55 | 16 | 26.30 | 28.85 | 32.00 | 34.27 |
| 7 | 14.07 | 16.01 | 18.48 | 20.28 | 17 | 27.59 | 30.19 | 33.41 | 35.72 |
| 8 | 15.51 | 17.54 | 20.09 | 21.96 | 18 | 28.87 | 31.53 | 34.81 | 37.16 |
| 9 | 16.92 | 19.02 | 21.67 | 23.59 | 19 | 30.14 | 32.85 | 36.19 | 38.58 |
| 10 | 18.31 | 20.48 | 23.21 | 25.19 | 20 | 31.41 | 34.17 | 37.57 | 40.00 |

$k = N_m - N_s$
Probability, $(1-\alpha)$ = 99% confidence

Figure 3.3 Complete Procedure of State Estimation of Power System

For fully understanding in the Weight Least Squares method, a brief discussion is prepared in section 3.1 where the WLS method is being applied on a simple dc circuit. In the verification of obtaining the estimated states, the equations used are slightly differ from the equations used in determining the estimated state in ac power system. This is due to the circuit equations are linear. While, a complete explanation on the procedure of State Estimation is in Chapter 4, where it is discussed on an ac power system. Furthermore, the measurement equations for ac power system are non-linear and iterative solutions are required as in Newton Raphson Power flow procedure.

## 3.1 Illustrative Example of Weight Least Squares Method

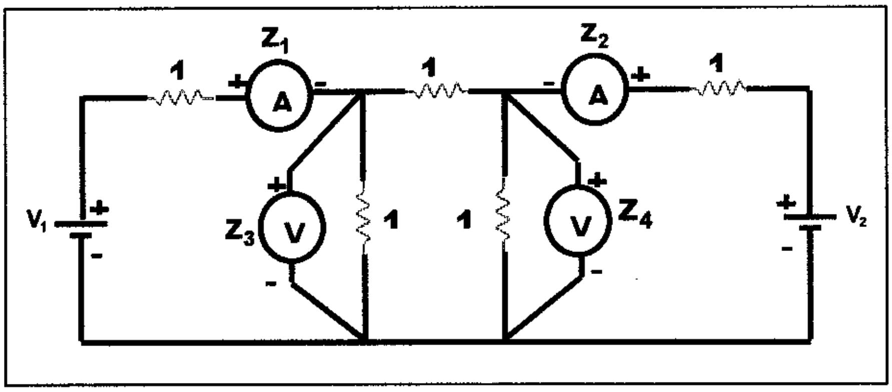


Figure 3.4 Simple DC Circuit

In understanding the WLS Method, it can be shown in a dc circuit referred in the Figure 3.4. The measurement set consists of:

- $z_1$ } Ammeters
- $z_2$
- $z_3$ } Voltmeter
- $z_4$

The circuit which has five resistances of 1Ω each need to estimate the two unknown voltage sources, $V_1$ and $V_2$. In determining the measurements, by applying the circuit analysis equation, it creates the equations for the true values of the measured quantities, $z_i$ in terms of the network parameters and the true but unknown voltage sources, $x_1 = V_1$ and $x_2 = V_2$. Then, it is also observed that error terms are added to the measurement equations to represent the error occurs in the meter readings. [1]

Referring to the Figure 3.4, the equations obtained are:

$$\underbrace{\begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{matrix}}_{\text{Measurements}} \quad \begin{matrix} = \\ = \\ = \\ = \end{matrix} \quad \underbrace{\begin{matrix} 5/8\,x_1 & - & 1/8\,x_2 \\ -1/8\,x_1 & + & 5/8\,x_2 \\ 3/8\,x_1 & + & 1/8\,x_2 \\ 1/8\,x_1 & + & 3/8\,x_2 \end{matrix}}_{\substack{\text{True values from} \\ \text{system model}}} \quad \begin{matrix} + \\ + \\ + \\ + \end{matrix} \quad \underbrace{\begin{matrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{matrix}}_{\text{Errors}}$$

Labeling the numerical coefficients of the equations as referred above with $h$, the new equations below are obtained, where $z_{j,\,true}$ is the true value of the actual measured quantity $z_j$.

$$\boxed{\begin{array}{ccccccccc} z_1 & = & h_{11}x_1 & + & h_{12}x_2 & + \; e_1 & = & z_{1,\,true} & + \; e_1 \\ z_2 & = & h_{21}x_1 & + & h_{22}x_2 & + \; e_2 & = & z_{2,\,true} & + \; e_2 \\ z_3 & = & h_{31}x_1 & + & h_{32}x_2 & + \; e_3 & = & z_{3,\,true} & + \; e_3 \\ z_4 & = & h_{41}x_1 & + & h_{42}x_2 & + \; e_4 & = & z_{4,\,true} & + \; e_4 \end{array}}$$

Now, the equations are rearranged into the matrix form:

$$\boxed{\begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} - \begin{bmatrix} z_{1,\,true} \\ z_{2,\,true} \\ z_{3,\,true} \\ z_{4,\,true} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} - \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{31} & h_{32} \\ h_{41} & h_{42} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}$$

Then to simplify the equations, the above equation can be written in a compact form which is shown as below:

$$\boxed{\mathbf{e} = \mathbf{z} - \mathbf{z}_{\mathbf{true}} = \mathbf{z} - \mathbf{Hx}}$$

The equation represents the errors between the actual measurements, z and the true (but unknown) values $\mathbf{z}_{\mathbf{true}} = \mathbf{Hx}$ of the measured quantities. The true values of $x_1$ and $x_2$ cannot be determined but the estimated values can be calculated which are:

$$\boxed{\hat{x}_1 \text{ and } \hat{x}_2}$$

These estimates are substituted in the equation that gives estimated values of errors in the form:

$$\boxed{\hat{\mathbf{e}} = \mathbf{z} - \mathbf{H}\hat{\mathbf{x}}}$$

Where, $\hat{\mathbf{e}}$ = Estimated errors

$\mathbf{z}$ = Actual measurements

$\mathbf{H}\hat{\mathbf{x}}$ = Estimates of true value, $z_j$

Once estimated state vector $\hat{x}$ is known, estimated measurements are calculated from $\hat{\mathbf{z}} = \mathbf{H}\hat{\mathbf{x}}$ and the estimated errors are determined as $\hat{\mathbf{e}} = \mathbf{z} - \hat{\mathbf{z}}$. In order to ensure a good accuracy of the measurements from meters, each term in the sum of squares is multiplied by an appropriate weighting factor, $w$, to find the state variables functions. [1]

$$\boxed{f = \sum_{j=1}^{4} w_j e_j^2 = w_1 e_1^2 + w_2 e_2^2 + w_3 e_3^2 + w_4 e_4^2}$$

Then, the estimated state variables as those values $\hat{x}_1$ and $\hat{x}_2$ which cause the function, $f$, to take its minimum value. By differentiating the equations of $f$ with respect to $x_1$ and $x_2$, below, we can get the minimum value which the equation equates to zero. Then, the equations have to be evaluated from the estimated of the unknown true values. [1]

$$\boxed{\frac{\partial f}{\partial x_1} = 2\left[ w_1 e_1 \frac{\partial e_1}{\partial x_1} + w_2 e_2 \frac{\partial e_2}{\partial x_1} + w_3 e_3 \frac{\partial e_3}{\partial x_1} + w_4 e_4 \frac{\partial e_4}{\partial x_1} \right] \Bigg|_{\hat{\mathbf{x}}} = 0}$$

$$\boxed{\frac{\partial f}{\partial x_2} = 2\left[ w_1 e_1 \frac{\partial e_1}{\partial x_2} + w_2 e_2 \frac{\partial e_2}{\partial x_2} + w_3 e_3 \frac{\partial e_3}{\partial x_2} + w_4 e_4 \frac{\partial e_4}{\partial x_{21}} \right] \Bigg|_{\hat{x}} = 0}$$

$$\begin{bmatrix} \frac{\partial e_1}{\partial x_1} & \frac{\partial e_2}{\partial x_1} & \frac{\partial e_3}{\partial x_1} & \frac{\partial e_4}{\partial x_1} \\ \frac{\partial e_1}{\partial x_2} & \frac{\partial e_2}{\partial x_2} & \frac{\partial e_3}{\partial x_2} & \frac{\partial e_4}{\partial x_2} \end{bmatrix} \begin{bmatrix} w_1 & & & \\ & w_2 & & \\ & & w_3 & \\ & & & w_4 \end{bmatrix} \begin{bmatrix} \hat{e}_1 \\ \hat{e}_2 \\ \hat{e}_3 \\ \hat{e}_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

In the matrix form as shown before, $w$ is the diagonal matrix of weighting factors. While, the partial derivatives are constant values which is denoted by **H**. As a result, the equation as referred below is obtained. [1]

$$\begin{bmatrix} h_{11} & h_{21} & h_{31} & h_{41} \\ h_{12} & h_{22} & h_{32} & h_{42} \end{bmatrix} \begin{bmatrix} w_1 & & & \\ & w_2 & & \\ & & w_3 & \\ & & & w_4 \end{bmatrix} \begin{bmatrix} \hat{e}_1 \\ \hat{e}_2 \\ \hat{e}_3 \\ \hat{e}_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

For simplification, a compact form is used as shown below:

$$\mathbf{H}^T \mathbf{W} \hat{\mathbf{e}} = \mathbf{0} \quad ; \quad \hat{\mathbf{e}} = \mathbf{z} - \hat{\mathbf{z}} \; ; \quad \hat{\mathbf{e}} = \mathbf{z} - \mathbf{H} \hat{\mathbf{x}}$$

$$\mathbf{H}^T \mathbf{W} (\mathbf{z} - \hat{\mathbf{z}}) = \mathbf{0}$$

$$\mathbf{H}^T \mathbf{W} (\mathbf{z} - \mathbf{H} \hat{\mathbf{x}}) = \mathbf{0}$$

$$\mathbf{H}^T \mathbf{W} \mathbf{H} \hat{\mathbf{x}} = \mathbf{H}^\mathrm{T} \mathbf{W} \mathbf{z}$$

Thus, solving the equation;

$$\hat{\mathbf{x}} = (\mathbf{H}^T \mathbf{W} \mathbf{H})^{-1} \mathbf{H}^\mathrm{T} \mathbf{W} \mathbf{z} \, ;$$

where the matrix of $\mathbf{H}^T \mathbf{W} \mathbf{H}$ is referred as the Gain Matrix, G.

Therefore,

$$\hat{\mathbf{x}} = \mathbf{G}^{-1} \mathbf{H}^T \mathbf{W} \mathbf{z}$$

The WLS method is expected to produce state estimates, $\hat{x}_i$, which are close to the true values $x_i$ of the state variables. In addition, the estimated states are given by the equation, $\hat{\mathbf{X}} = \mathbf{G}^{-1} \ \mathbf{H}^\mathrm{T} \ \mathbf{W} \ \mathbf{Z}$ because the circuit equations are linear compared to the Power System Network that has the nonlinear equations. This will be discussed further in Chapter 4. An expression for the differences $(\hat{x}_i - x_i)$ is found by substituting for $\mathbf{z} = \mathbf{Hx} + \mathbf{e}$ in the previous equation in order to obtain:

$$\begin{aligned}\hat{\mathbf{x}} &= \mathbf{G}^{-1}\,\mathbf{H}^{T}\,\mathbf{W}\,(\mathbf{Hx} + \mathbf{e})\\ &= \mathbf{G}^{-1}\,(\mathbf{H}^{T}\,\mathbf{WH})\,\mathbf{x} + \mathbf{G}^{-1}\,\mathbf{H}^{T}\,\mathbf{W}\,\mathbf{e}\\ &= \mathbf{G}^{-1}\,\mathbf{G}\,\mathbf{x} + \mathbf{G}^{-1}\,\mathbf{H}^{T}\,\mathbf{W}\,\mathbf{e}\\ &= \mathbf{x} + \mathbf{G}^{-1}\,\mathbf{H}^{T}\,\mathbf{W}\,\mathbf{e}\end{aligned}$$

## 3.2 Detection and Removal of Bad Data

In a Power System network, the state estimates calculated by the WLS method can be accepted if the measurements are accurate. If a measurement contains bad data, it should be detected and identified so that it can be removed from the calculations. Chi-square distribution, $\hat{f}$ provides a test to detect the bad data in the measurement set, which is as follows: [1]

The chi-square distribution, $\hat{f}$ provides a test to detect the bad data in the measurement set. The procedure to test for the bad data is as follows:

i) By using the measured quantities, $z_j$, the estimated state variables, $\hat{x}_i$ of the system are determined by using WLS method from the equation below:

$$\boxed{\hat{\mathbf{x}} = \mathbf{x} + \mathbf{G}^{-1}\,\mathbf{H}^{T}\,\mathbf{W}\,\mathbf{e}}$$

ii) Then, substitute the estimated values, $\hat{x}_i$ into the equation $\hat{\mathbf{z}} = \mathbf{H}\hat{\mathbf{x}}$ to calculate the estimated values $\hat{z}_j$ of the measurements and the estimated errors can be obtained as:

$$\boxed{\hat{e}_j = z_j - \hat{z}_j}$$

iii) Evaluate the sum of weighted squares estimated errors from equation below by using variances, $\sigma^2$;

$$\boxed{\hat{f} = \sum_{j=1}^{N_m} w_j \hat{e}_j^{\,2} = \sum_{j=1}^{N_m} \frac{\hat{e}_j^{\,2}}{\sigma_j^2}}$$

iv) The condition of the chi-square value to be less than the critical value of the chi-square distribution must be satisfied in order that the measured data and the state estimates can be accepted as being accurate.

$$\boxed{\hat{f} \langle \chi^2_{k,a}}$$

For the number of degrees of freedom; *k* and a probability α, the critical chi-square value $\chi^2_{k,\alpha}$ can be referred from Table 4.1. Where k is obtained from;

$$\boxed{k = (N_m - N_s)}$$

If it is satisfied, then the measured raw data and the state estimates are accepted as being accurate.

Table 3.1 Critical Values of $\chi^2_{k,a}$

| | α | | | | | | α | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **k** | **0.05** | **0.025** | **0.01** | **0.005** | | **k** | **0.05** | **0.025** | **0.01** | **0.005** |
| 1 | 3.84 | 5.02 | 6.64 | 7.88 | | 11 | 19.68 | 21.92 | 24.73 | 26.76 |
| 2 | 5.99 | 7.38 | 9.21 | 10.60 | | 12 | 21.03 | 23.34 | 26.22 | 28.30 |
| 3 | 7.82 | 9.35 | 11.35 | 12.84 | | 13 | 22.36 | 24.74 | 27.69 | 29.82 |
| 4 | 9.49 | 11.14 | 13.28 | 14.86 | | 14 | 23.69 | 26.12 | 29.14 | 31.32 |
| 5 | 11.07 | 12.83 | 15.09 | 16.75 | | 15 | 25.00 | 27.49 | 30.58 | 32.80 |
| 6 | 12.59 | 14.45 | 16.81 | 18.55 | | 16 | 26.30 | 28.85 | 32.00 | 34.27 |
| 7 | 14.07 | 16.01 | 18.48 | 20.28 | | 17 | 27.59 | 30.19 | 33.41 | 35.72 |
| 8 | 15.51 | 17.54 | 20.09 | 21.96 | | 18 | 28.87 | 31.53 | 34.81 | 37.16 |
| 9 | 16.92 | 19.02 | 21.67 | 23.59 | | 19 | 30.14 | 32.85 | 36.19 | 38.58 |
| 10 | 18.31 | 20.48 | 23.21 | 25.19 | | 20 | 31.41 | 34.17 | 37.57 | 40.00 |

v) However, if the condition is not met, there is possibility of the presence of bad measurements. So, in identifying the bad data, standardized error estimates are computed by using diagonal elements, $R'_{jj}$.

$$\boxed{\mathbf{R}_{jj}' = \mathbf{R} - \mathbf{H}\,\mathbf{G}^{-1}\,\mathbf{H}^{T} = [\mathbf{I} - \mathbf{H}\,\mathbf{G}^{-1}\,\mathbf{H}^{T}\,\mathbf{R}^{-1}]\,\boldsymbol{R}}$$

Then, the bad data is identified as the measurement that has the highest magnitude or known as the largest standardized error, which is attained from the standardized error in equation below. So, the bad data is removed from the measurement set.

$$\boxed{\frac{z_j - \hat{z}_j}{\sqrt{R'_{jj}}} \quad i.e. \quad \frac{\hat{e}_j}{\sqrt{R'_{jj}}}}$$

Consequently, the State Estimation calculation is repeated with a new Nm, which is reduced by one, as well as recalculating the sum of weighted squared errors, $\hat{f}$. If the new value of $\hat{f}$ satisfies the chi-square test of inequality, then the discarded measurement has been successfully identified as the bad data.

# CHAPTER 4

# RESULTS AND DISCUSSION

## 4.1 Test and Analysis on 2-bus Power System Network

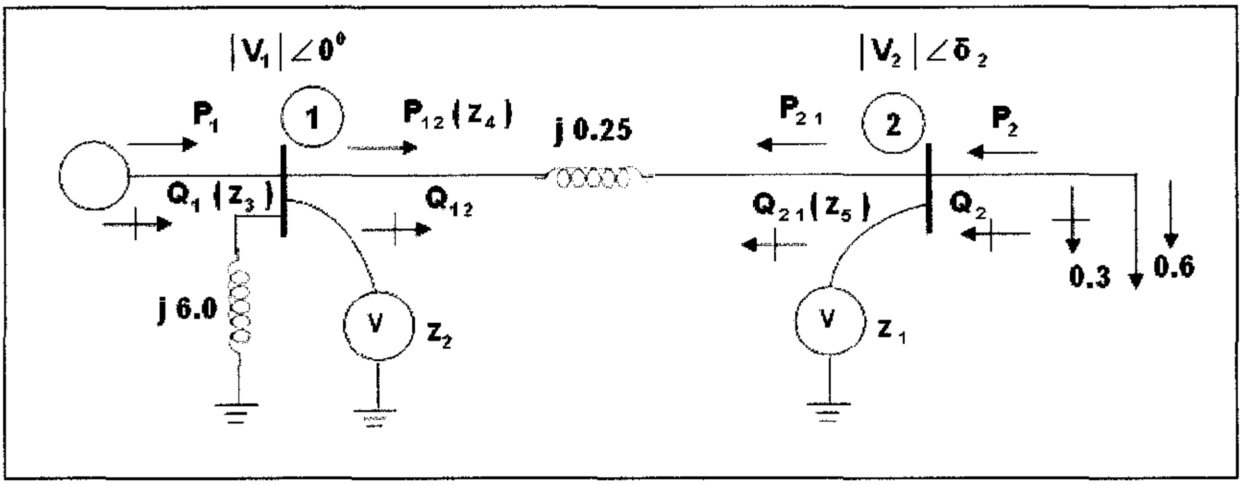


Figure 4.1 One Line Diagram of a Power System

Referring to the Figure 4.1 above, a simple 2-bus system is installed with two voltmeters, two varmeters and one wattmeter in order to measure the following five quantities (Refer Table 4.4). In performing the State Estimation for this 2-bus system, these input data in the table below (Refer Table 4.1, 4.2 and 4.3) are used in order to obtain the estimated state variables.

Table 4.1 Bus Data for 2-Bus System (in p.u.)

| **Bus No.** | **Type** | **Load** | | $\lvert V_i\rvert$ | $\delta_i$ |
|---|---|---|---|---|---|
| | | $P_i$ | $Q_i$ | | |
| 1 | Slack Bus | 0 | 0 | 1.0 | 0 |
| 2 | P-Q Bus | 0.6 | 0.3 | – | – |

Table 4.2 Line Data for 2-Bus System (in p.u.)

| **Line No.** | **Between Buses** | **Line Impedances** |
|---|---|---|
| 1 | 1-2 | 0 + j0.25 |

Table 4.3 Shunt Capacitor Data for 2-Bus System (in p.u.)

| **Bus No.** | **Admittance** |
|---|---|
| 1 | j6.0 |

Table 4.4 Measured Quantities on the Physical System Before Bad Data is Eliminated (in p.u.)

| **Bus & Line No.** | **Metering Equipment** | **Symbols** | **No. of measurement, $z_{Nm}$** | **Measured Quantities** | **Variances, $\sigma^2_{Nm}$** |
|---|---|---|---|---|---|
| Bus 2, voltage magnitude | voltmeter | $\lvert V_2 \rvert$ | $z_1$ | 0.92 | $(0.01)^2$ |
| Bus 1, voltage magnitude | voltmeter | $\lvert V_1 \rvert$ | $z_2$ | 1.02 | $(0.01)^2$ |
| Bus 1, reactive power injection | varmeter | $Q_1$ | $z_3$ | 0.605 | $(0.02)^2$ |
| Line $P_{12}$, flow from bus 1 to bus 2 | wattmeter | $P_{12}$ | $z_4$ | 0.598 | $(0.015)^2$ |
| Line $Q_{21}$, flow from bus 2 to bus 1 | varmeter | $Q_{21}$ | $z_5$ | 0.305 | $(0.02)^2$ |

### *4.1.1 Results for Mathematical Approach*

**BEFORE DETECTION & REMOVAL OF BAD DATA**

Number of measurement, Nm = 5

No of iterations = 5

Table 4.5 Estimated State Variables Before Detection & Removal of Bad Data

| $\hat{x}_{Ns}$ | **Flat-start Values** | **Estimated State Variables** |
|---|---|---|
| $\hat{x}_1 = \delta_2,$ | 0 | – 0.1762 rad |
| $\hat{x}_2 = \lvert V_2 \rvert$ | 1.0 | 0.9578 p.u. |
| $\hat{x}_3 = \lvert V_1 \rvert$ | 1.0 | 0.9843 p.u. |

## DETECTION & REMOVAL OF BAD DATA

Weighted Sum of Squares, $\hat{f} = 544.8358 > \chi^2_{2,0.01} = 9.21$

Consequently, in identifying which measurement contains the bad data, the standardized errors are computed on all the measurements.

Table 4.6 Standardized Errors Before Detection & Removal of Bad Data

| $z_{Nm}$ | **Standardized Errors** |
|---|---|
| $z_1$ | − 5.71026 |
| $z_2$ | 5.03667 |
| $z_3$ | 20.50931 |
| $z_4$ | − 22.65953 |
| $z_5$ | 23.34099 |

The largest standardized error as observed from the Table 4.6 is the measurement number 5. Therefore, the bad data is detected at reactive power flow, $Q_{21} = z_5$ and it is removed from the measurement set in order to repeat the State Estimation Calculations with the flat-start values again.

## AFTER DETECTION & REMOVAL OF BAD DATA

Number of measurement, $N_m = 4$

No of iterations = 3

Table 4.7 Measured Quantities on the Physical System After Bad Data is Eliminated (in p.u.)

| **Bus & Line No.** | **Metering Equipment** | **Symbols** | **No. of measurement,** $z_{Nm}$ | **Measured Quantities** | **Variances,** $\sigma^2_{Nm}$ |
|---|---|---|---|---|---|
| Bus 2, voltage magnitude | voltmeter | $\lvert V_2 \rvert$ | $z_1$ | 0.92 | $(0.01)^2$ |
| Bus 1, voltage magnitude | voltmeter | $\lvert V_1 \rvert$ | $z_2$ | 1.02 | $(0.01)^2$ |
| Bus 1, reactive power injection | varmeter | $Q_1$ | $z_3$ | 0.605 | $(0.02)^2$ |
| Line $P_{12}$, flow from bus 1 to bus 2 | wattmeter | $P_{12}$ | $z_4$ | 0.598 | $(0.015)^2$ |

Table 4.8 Estimated State Variables After Detection & Removal of Bad Data

| $\hat{x}_{Ns}$ | **Flat-start Values** | **Estimated State Variables** |
|---|---|---|
| $\hat{x}_1 = \delta_2,$ | 0 | – 0.1600 rad |
| $\hat{x}_2 = \lvert V_2 \rvert$ | 1.0 | 0.9223 p.u. |
| $\hat{x}_3 = \lvert V_1 \rvert$ | 1.0 | 1.0174 p.u. |

## DETECTION OF BAD DATA

Weighted Sum of Squares, $\hat{f} = 0.1328 < \chi^2_{1,0.01} = 6.64$

As a result, the estimated state variables are accepted with 99% confidence. Hence, it is proven that the discarded measurement, $z_5$ do consist bad data.

***The calculations are as below:***

i) For this two-bus system, firstly, is to initialize the state variables with the flat-start value which are:

$$\begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{bmatrix} = \begin{bmatrix} \delta_2 \\ |V_2| \\ |V_1| \end{bmatrix} \begin{array}{l} = 0^\circ \\ = 1.0 \\ = 1.0 \end{array}$$

Whereas, $\delta_1 = 0^0$ is chosen to be the reference angle since the phase angles, δ, of the bus voltages cannot be measured.

ii) Then, Real and reactive power injections are calculated from:

$$\boxed{\begin{aligned} P_i &= |V_i|^2 \, G_{ii} + \sum_{\substack{n=1 \\ n \neq i}}^{N} |V_i| \, |V_n| \, |Y_{in}| \; cos\,(\theta_{in} + \delta_n - \delta_i) \\ Q_i &= -|V_i|^2 \, B_{ii} - \sum_{\substack{n=1 \\ n \neq i}}^{N} |V_i| \, |V_n| \, |Y_{in}| \; sin\,(\theta_{in} + \delta_n - \delta_i) \end{aligned}}$$

While, the real and reactive power flows in line *ij* are calculated from:

$$\boxed{\begin{aligned} P_{ij} &= -|V_i|^2 \, G_{ij} + |V_i||V_j||Y_{ij}| \, cos(\theta_{ij} + \delta_j - \delta_i) \\ Q_{ij} &= -|V_i|^2 \, (\frac{B'_{ij}}{2} - B_{ij}) - |V_i||V_j||Y_{ij}| \, sin(\theta_{ij} + \delta_j - \delta_i) \end{aligned}}$$

Thus, the admittance matrix can be obtained as:

$$\boxed{Y = G + jB}$$

$$Y_{bus} = \begin{bmatrix} -j\,4.1667 & j\,4 \\ j\,4 & -j\,4 \end{bmatrix}; \;\; G = 0;$$

$$|Y_{bus}| = \begin{bmatrix} 4.1667 & 4 \\ 4 & 4 \end{bmatrix}; \qquad \theta = \begin{bmatrix} -90^0 & 90^0 \\ 90^0 & -90^0 \end{bmatrix}$$

Consequently, the expressions for the real and reactive line flows and reactive power injection in terms of $x_{Ns}$, state variables which are used for linearizing the ac power system problem are as follows:

$$\begin{aligned} P_{12} &= |V_1||V_2|\,4\cos(90 + \delta_2 - \delta_1) \\ &= -4\,|V_1||V_2|\sin\delta_2 \\ &= -4\,x_3\,x_2\sin x_1 \end{aligned}$$

$$\begin{aligned} Q_{21} &= -|V_2|^2(-4) - |V_2||V_1|\,4\sin(90^0 + \delta_1 - \delta_2) \\ &= 4\,|V_2|^2 - 4\,|V_2||V_1|\cos\delta_2 \\ &= 4(x_2^2 - x_3\,x_2\cos x_1) \end{aligned}$$

$$\begin{aligned} Q_1 &= -|V_1|^2(-4.1667) - |V_1||V_2|\,4\sin(90 + \delta_2 - \delta_1) \\ &= 4.1667\,|V_1|^2 - 4\,|V_1||V_2|\cos\delta_2 \\ &= 4.1667\,x_3^2 - 4\,x_3\,x_2\cos x_1 \end{aligned}$$

iii) Then, in determining the measurement errors equation, it is noted that the function, $h_{Nm}$ for the true values of the measured quantities, in terms of estimated state variables, $x_{Ns}$ are as below:

$$\begin{aligned} h_1 &= x_2 \\ h_2 &= x_3 \\ h_3 &= 4.1667\,x_3^2 - 4\,x_3\,x_2\cos x_1 \\ h_4 &= -4\,x_3\,x_2\sin x_1 \\ h_5 &= 4(x_2^2 - x_3\,x_2\cos x_1) \end{aligned}$$

Next, the function of $h_{Nm}$ is subtituted in the measurement errors equation in order to solve the equations iteratively, begins with the flat-start values stated in Step i).

Then, substitute the numerical values into the function of $h_{Nm}$ to solve the measurement errors for the first iteration;

$$\begin{aligned} e_1^{(0)} &= z_1 - x_2^{(0)} = 0.92 - 1.0 = -0.0800 \\ e_2^{(0)} &= z_2 - x_3^{(0)} = 1.02 - 1.0 = 0.0200 \\ e_3^{(0)} &= z_3 - (4.1667\,x_3^{(0)^2} - 4\,x_3^{(0)}\,x_2^{(0)}\cos x_1^{(0)}) = 0.605 - (4.1667 - 4) = 0.4383 \\ e_4^{(0)} &= z_4 - (-4\,x_3^{(0)}\,x_2^{(0)}\sin x_1^{(0)}) = 0.5980 - 0 = 0.5980 \\ e_5^{(0)} &= z_5 - (4\,x_2^{(0)^2} - 4\,x_3^{(0)}\,x_2^{(0)}\cos x_1^{(0)}) = 0.305 - (4 - 4) = 0.3050 \end{aligned}$$

iv) In the same iteration, the partial derivatives of $h_1$, $h_2$, $h_3$, $h_4$ and $h_5$ are evaluated according to the Jacobian matrix, $\mathrm{H}_{\mathrm{x}}^{(k)}$ which is obtained as;

$$\mathrm{H}_{\mathrm{x}}^{(0)} = \begin{bmatrix} \left.\frac{\partial h_1}{\partial x_1}\right|^{(0)} & \left.\frac{\partial h_1}{\partial x_2}\right|^{(0)} & \left.\frac{\partial h_1}{\partial x_3}\right|^{(0)} \\ \left.\frac{\partial h_2}{\partial x_1}\right|^{(0)} & \left.\frac{\partial h_2}{\partial x_2}\right|^{(0)} & \left.\frac{\partial h_2}{\partial x_3}\right|^{(0)} \\ \left.\frac{\partial h_3}{\partial x_1}\right|^{(0)} & \left.\frac{\partial h_3}{\partial x_2}\right|^{(0)} & \left.\frac{\partial h_3}{\partial x_3}\right|^{(0)} \\ \left.\frac{\partial h_4}{\partial x_1}\right|^{(0)} & \left.\frac{\partial h_4}{\partial x_2}\right|^{(0)} & \left.\frac{\partial h_4}{\partial x_3}\right|^{(0)} \\ \left.\frac{\partial h_5}{\partial x_1}\right|^{(0)} & \left.\frac{\partial h_5}{\partial x_2}\right|^{(0)} & \left.\frac{\partial h_5}{\partial x_3}\right|^{(0)} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 4\,x_3^{(0)}\,x_2^{(0)}\sin x_1^{(0)} & -4\,x_3^{(0)}\cos x_1^{(0)} & 8.3333\,x_3^{(0)} - 4x_2^{(0)}\cos x_1^{(0)} \\ -4x_3^{(0)}\,x_2^{(0)}\cos x_1^{(0)} & -4x_3^{(0)}\sin x_1^{(0)} & -4x_2^{(0)}\sin x_1^{(0)} \\ 4x_3^{(0)}\,x_2^{(0)}\sin x_1^{(0)} & 8x_2^{(0)} - 4x_3^{(0)}\cos x_1^{(0)} & 4x_2^{(0)}\cos x_1^{(0)} \end{bmatrix}$$

$$\mathrm{H}_{\mathrm{X}}^{(0)} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -4 & 4.3333 \\ -4 & 0 & 0 \\ 0 & 4 & -4 \end{bmatrix}$$

Before proceeding with the calculation of the gain matrix, $G_X^{(0)} = H_X^{(0)T} R^{-1} H_X^{(0)}$, the weighting matrix, $R^{-1}$ of reciprocal variances is calculated as below;

$$R^{-1} = \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sigma_2^2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sigma_3^2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sigma_4^2} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sigma_5^2} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{(0.01)^2} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{(0.01)^2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{(0.02)^2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{(0.015)^2} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{(0.02)^2} \end{bmatrix}$$

The gain matrix, $G_X^{(0)} = H_X^{(0)T} R^{-1} H_X^{(0)}$ can now be calculated with the achieved values of $H_X^{(0)}$ and $R^{-1}$ . It can be shown as follows:

$$G_X^{(0)} = \begin{bmatrix} 0 & 0 & 0 & -4 & 0 \\ 1 & 0 & -4 & 0 & 4 \\ 0 & 1 & 4.3333 & 0 & -4 \end{bmatrix} \begin{bmatrix} 10000 & 0 & 0 & 0 & 0 \\ 0 & 10000 & 0 & 0 & 0 \\ 0 & 0 & 2500 & 0 & 0 \\ 0 & 0 & 0 & 4444.44 & 0 \\ 0 & 0 & 0 & 0 & 2500 \end{bmatrix} H_X^{(0)}$$

$$= 10^4 \times \begin{bmatrix} 0 & 0 & 0 & -1.7778 & 0 \\ 1 & 0 & -1 & 0 & 1 \\ 0 & 1 & 1.0833 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -4 & 4.3333 \\ -4 & 0 & 0 \\ 0 & 4 & -4 \end{bmatrix}$$

$$= \begin{bmatrix} 7.1111 & 0 & 0 \\ 0 & 9 & -8.3333 \\ 0 & -8.3333 & 9.6944 \end{bmatrix} \times 10^4$$

v) In calculating the state estimates, it is based on $\mathrm{H}_{\mathrm{X}}^{(k)}$ and $e_j^{(k)}$. As for the first iteration of the estimated state variables are as below:

$$\begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \\ x_3^{(1)} \end{bmatrix} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{bmatrix} + \mathrm{G}_{\mathrm{X}}^{(0)^{-1}} \ \mathrm{H}_{\mathrm{X}}^{(0)^T} \ \mathrm{R}^{-1} \begin{bmatrix} e_1^{(0)} \\ e_2^{(0)} \\ e_3^{(0)} \\ e_4^{(0)} \\ e_5^{(0)} \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} + \mathrm{G}_{\mathrm{X}}^{(0)^{-1}} \times 10^4 \begin{bmatrix} 0 & 0 & 0 & -1.7778 & 0 \\ 1 & 0 & -1 & 0 & 1 \\ 0 & 1 & 1.0833 & 0 & 1 \end{bmatrix} \begin{bmatrix} -0.0800 \\ 0.0200 \\ 0.4383 \\ 0.5980 \\ 0.3050 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0.1406 & 0 & 0 \\ 0 & 0.5445 & 0.4680 \\ 0 & 0.4680 & 0.5055 \end{bmatrix} \begin{bmatrix} -1.0631 \\ -0.2133 \\ 0.1898 \end{bmatrix}$$

$$= \begin{bmatrix} -0.1495 \\ 0.9727 \\ 0.9961 \end{bmatrix}$$

Hence, the first iteration is completed and the present state variables are:

$$x_1^{(1)} = -0.1495 \text{ rad} = -8.56572^{\circ}$$

$$x_2^{(1)} = 0.9727 \text{ p.u.}$$

$$x_3^{(1)} = 0.9961 \text{ p.u.}$$

Consequently, the current values for the first iteration of the state variables become the input data to the second iteration and the calculations continue until the values of the state variables, $x_{Ns}$ are converged.

2nd iteration;

$$x_1^{(2)} = -0.1721 \text{ rad} = -9.86060^{\circ}$$

$$x_2^{(2)} = 0.9610 \text{ p.u.}$$

$$x_3^{(2)} = 0.9870 \text{ p.u.}$$

3rd iteration;

$$x_1^{(3)} = -0.1756 \text{ rad} = -10.06114^{\circ}$$

$$x_2^{(3)} = 0.9583 \text{ p.u.}$$

$$x_3^{(3)} = 0.9847 \text{ p.u.}$$

4th iteration;

$$x_1^{(4)} = -0.1761 \text{ rad} = -10.08979^{\circ}$$

$$x_2^{(4)} = 0.9579 \text{ p.u.}$$

$$x_3^{(4)} = 0.9843 \text{ p.u.}$$

5th iteration;

$$x_1^{(5)} = -0.1762 \text{ rad} = -10.09552^{\circ}$$

$$x_2^{(5)} = 0.9578 \text{ p.u.}$$

$$x_3^{(5)} = 0.9843 \text{ p.u.}$$

So, these values are selected as the estimated state variables, $\hat{x}_{Ns}$

$$\hat{x}_1 = \delta_2 = -0.1762 \text{ rad} = -10.09552^0$$

$$\hat{x}_2 = |V_2| = 0.9578 \text{ p.u.}$$

$$\hat{x}_3 = |V_1| = 0.9843 \text{ p.u.}$$

vi) By applying the estimated state variables obtained in the 5th iteration, the estimated measurement errors equation is used in testing the presence of the bad data in the measurements, The resultant estimates of the measurement errors in p.u. are as follows:

$$\hat{e}_1 = z_1 - \hat{x}_2 = 0.92 - 0.9578 = -0.0378$$

$$\hat{e}_2 = z_2 - \hat{x}_3 = 1.02 - 0.9843 = 0.0357$$

$$\hat{e}_3 = z_3 - (4.1667\,\hat{x}_3^{\,2} - 4\,\hat{x}_3\,\hat{x}_2 \cos \hat{x}_1) = 0.6050 - 0.3242 = 0.2808$$

$$\hat{e}_4 = z_4 - (-4\,\hat{x}_3\,\hat{x}_2 \sin \hat{x}_1) = 0.5980 - 0.6610 = -0.0630$$

$$\hat{e}_5 = z_5 - (4\,\hat{x}_2^{\,2} - 4\,\hat{x}_3\,\hat{x}_2 \cos \hat{x}_1) = 0.3050 - (-0.0432) = 0.3482$$

vii) Then, when the estimated of errors are attained, the weighted sum of squares of the estimated errors can be computed to determine the chi-square, $\hat{f}$ value. This is due to the verification on the presence of the bad data in the measurements.

$$\hat{f} = \sum_{j=1}^{5} \frac{\hat{e}_j^{\,2}}{\sigma_j^2} = (10000 \times (-0.0378)^2) + (10000 \times (0.0357)^2) + (2500 \times (0.2808)^2)$$

$$+ (4444.4444 \times (-0.0630)^2) + (2500 \times (0.3482)^2)$$

$$= 544.903$$

For this power system problem,

$N_m = 5; N_s = 3, \alpha = 0.01,$

$k = N_m - N_s = 2$

$\chi^2_{2,0.01} = 9.21$ (Refer to Table 3.1)

$\hat{f} = 545 > \chi^2_{2,0.01} = 9.21$

The condition is not satisfied. This indicates the presence of bad data in the measurements. Consequently, the estimated state variables, $\hat{x}_1$, $\hat{x}_2$ and $\hat{x}_3$ cannot be accepted as being accurate.

viii) For the identification and removal of bad data detected in the measurement, the standardized errors of all the estimated errors can be computed by determining the diagonal elements of the covariance matrix, $R'_{jj}$, using the values of the estimated state variables;

$$\boxed{R'_{jj} = R - H_X G_X^{-1} H_X^T = [\, I - H_X G_X^{-1} H_X^T R^{-1} \,] R}$$

$$
\mathrm{H_X} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 4\,\hat{x}_3\,\hat{x}_2 \sin \hat{x}_1 & -\,4\,\hat{x}_3 \cos \hat{x}_1 & 8.3333\,\hat{x}_3 - 4\,\hat{x}_2 \cos \hat{x}_1 \\ -4\,\hat{x}_3\,\hat{x}_2 \cos \hat{x}_1 & -4\,\hat{x}_3 \sin \hat{x}_1 & -4\,\hat{x}_2 \sin \hat{x}_1 \\ 4\,\hat{x}_3\,\hat{x}_2 \sin \hat{x}_1 & 8\,\hat{x}_2 - 4\,\hat{x}_3 \cos \hat{x}_1 & -4\,\hat{x}_2 \cos \hat{x}_1 \end{bmatrix}
$$

$$
= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -0.6610 & -3.8761 & 4.4303 \\ -3.7125 & 0.6901 & 0.6716 \\ -0.6610 & 3.7863 & -3.7719 \end{bmatrix}
$$

In obtaining the diagonal elements of the covariance matrix, $\mathrm{R}'_{\mathrm{jj}}$, numerical elements of $\mathrm{H_X}$ which use the estimated state variables, as well as the weighting matrix of reciprocal variances, $\mathrm{R}^{-1}$, are substituted into the gain matrix equation. It is shown as below:

$$
\mathrm{G_X} = \mathrm{H}_{\mathrm{X}}^{T}\,\mathrm{R}^{-1}\,\mathrm{H_X} = \begin{bmatrix} 6.3442 & -1.1239 & -1.2169 \\ -1.1239 & 8.5518 & -7.6575 \\ -1.2169 & -7.6575 & 9.6641 \end{bmatrix} \times 10^{4}
$$

Thus, the diagonal elements of the matrix is;

$$
\mathrm{H_X}\,\mathrm{G}_{\mathrm{x}}^{-1}\,\mathrm{H}_{\mathrm{X}}^{T}\,\mathrm{R}^{-1} = \begin{bmatrix} 0.5618 & \bullet & \bullet & \bullet & \bullet \\ \bullet & 0.4976 & \bullet & \bullet & \bullet \\ \bullet & \bullet & 0.5307 & \bullet & \bullet \\ \bullet & \bullet & \bullet & 0.9656 & \bullet \\ \bullet & \bullet & \bullet & \bullet & 0.4443 \end{bmatrix}
$$

Hence, the diagonal elements of the covariance matrix, $\mathrm{R}'_{\mathrm{jj}}$, are now calculated as follows:

$$
\mathrm{R}'_{\mathrm{jj}} = [\,\mathrm{I} - \mathrm{H_X}\,\mathrm{G}_{\mathrm{x}}^{-1}\,\mathrm{H}_{\mathrm{X}}^{T}\,\mathrm{R}^{-1}\,]\,\mathrm{R}
$$

$$R'_{jj} = \begin{bmatrix} 0.4382 & \bullet & \bullet & \bullet & \bullet \\ \bullet & 0.5024 & \bullet & \bullet & \bullet \\ \bullet & \bullet & 0.4693 & \bullet & \bullet \\ \bullet & \bullet & \bullet & 0.0344 & \bullet \\ \bullet & \bullet & \bullet & \bullet & 0.5557 \end{bmatrix} \begin{bmatrix} 1 & \bullet & \bullet & \bullet & \bullet \\ \bullet & 1 & \bullet & \bullet & \bullet \\ \bullet & \bullet & 4 & \bullet & \bullet \\ \bullet & \bullet & \bullet & 2.25 & \bullet \\ \bullet & \bullet & \bullet & \bullet & 4 \end{bmatrix} \times 10^{-4}$$

$$= \begin{bmatrix} 0.4382 & \bullet & \bullet & \bullet & \bullet \\ \bullet & 0.5024 & \bullet & \bullet & \bullet \\ \bullet & \bullet & 1.8772 & \bullet & \bullet \\ \bullet & \bullet & \bullet & 0.0774 & \bullet \\ \bullet & \bullet & \bullet & \bullet & 2.2228 \end{bmatrix} \times 10^{-4}$$

ix) Then, the standardized errors for all the measurements can be achieved by using the estimated value of the measurement errors and the diagonal elements of the covariance matrix, $R'_{jj}$.

$$\frac{\hat{e}_1}{\sqrt{R'_{11}}} = \frac{-0.0378}{\sqrt{0.4382 \times 10^{-4}}} = -5.7103$$

$$\frac{\hat{e}_2}{\sqrt{R'_{22}}} = \frac{0.0357}{\sqrt{0.5024 \times 10^{-4}}} = 5.0367$$

$$\frac{\hat{e}_3}{\sqrt{R'_{33}}} = \frac{0.2808}{\sqrt{1.8772 \times 10^{-4}}} = 20.4947$$

$$\frac{\hat{e}_4}{\sqrt{R'_{44}}} = \frac{-0.0630}{\sqrt{0.0774 \times 10^{-4}}} = -22.6449$$

$$\frac{\hat{e}_5}{\sqrt{R'_{55}}} = \frac{0.3482}{\sqrt{2.2228 \times 10^{-4}}} = 23.3549$$

From the results obtained, it is observed that the largest standardized error is the measurements, $z_5$ which has the maximum magnitude value. Hence, it has to be discarded from the measurement set. Consequently, the number of measurement is reduced by one.

x) After the removal of the bad data, the state estimation calculations are repeated with the other four data in the measurement set, which are $z_1$, $z_2$, $z_3$, and $z_4$ and the flat-start values,

$$\begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{bmatrix} = \begin{bmatrix} \delta_2 \\ |V_2| \\ |V_1| \end{bmatrix} \begin{matrix} = 0^\circ \\ = 1.0 \\ = 1.0 \end{matrix}$$

When all the steps are repeated and the convergence is reached at 3rd iteration, the estimated state variables are as follows:

$$\hat{x}_1 = \delta_2 = -0.1600 \text{ rad} = -9.1673^0$$

$$\hat{x}_2 = |V_2| = 0.9223 \text{ p.u.}$$

$$\hat{x}_3 = |V_1| = 1.0174 \text{ p.u.}$$

Accordingly, in ensuring the acceptance of the new estimated state variables, which are free from the presence of the bad data in the measurements, the calculations for the estimated measurement errors are carried out as follows:

$$\hat{e}_1 = z_1 - \hat{x}_2 = 0.92 - 0.9223 = -0.0023$$

$$\hat{e}_2 = z_2 - \hat{x}_3 = 1.02 - 1.0174 = 0.0026$$

$$\hat{e}_3 = z_3 - (4.1667\ \hat{x}_3^{\,2} - 4\ \hat{x}_3\ \hat{x}_2\ \cos\hat{x}_1) = 0.605 - 0.6072 = \text{-}0.0022$$

$$\hat{e}_4 = z_4 - (-4\ \hat{x}_3\ \hat{x}_2\ \sin\hat{x}_1) = 0.598 - 0.5978 = 0.0002$$

Then, when the estimated measurement errors are obtained, estimated errors are used in testing the presence of bad data in the measurement set. So, the weighted sum of squares is again used in comparing the chi-square value with the critical value of the chi-square distribution.

$$\hat{f} = \sum_{j=1}^{4} \frac{\hat{e}_j^{\,2}}{\sigma_j^2} = (10000 \text{ x } (-0.0023)^2) + (10000 \text{ x } (0.0026)^2) + (2500 \text{ x } (-0.0022)^2)$$

$$+ (4444.4444 \text{ x } (0.0002)^2)$$

$$= 0.1328$$

New $N_m = 4$; $N_s = 3$, $\alpha = 0.01$

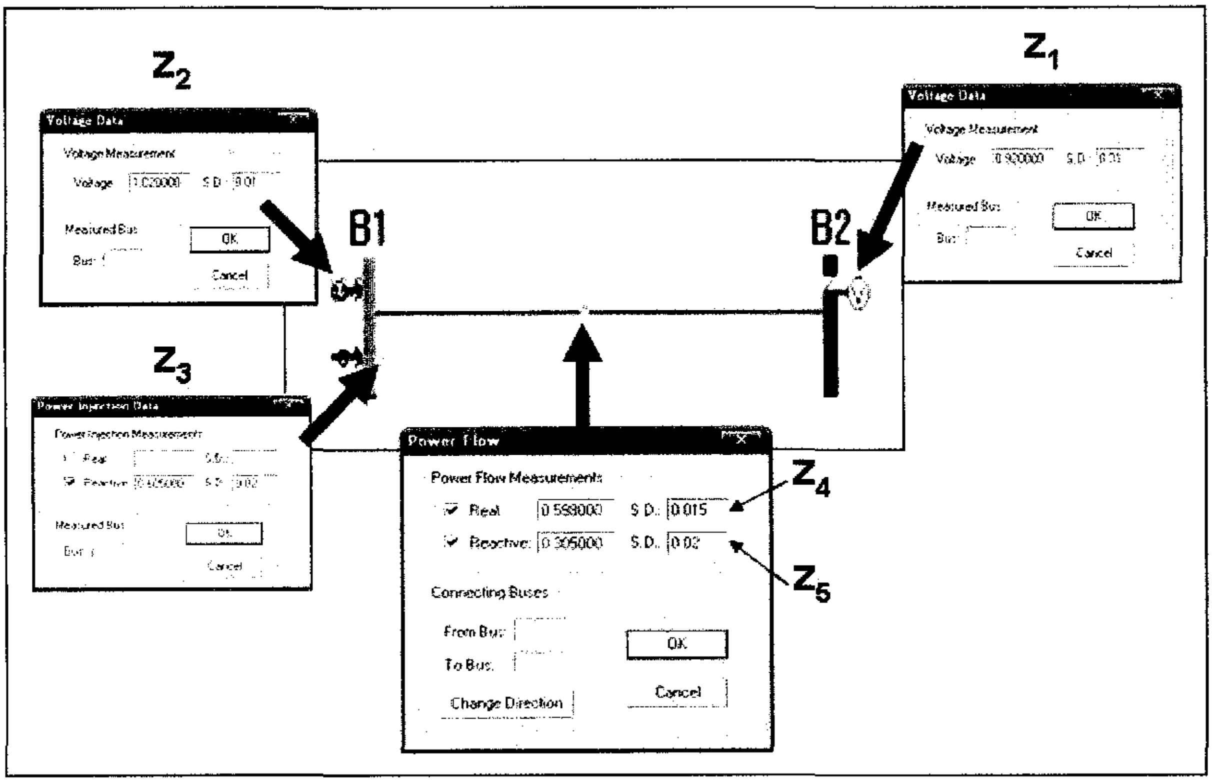
$k = N_m - N_s = 1$

$\chi^2_{1,0.01} = 6.64$ (Refer to Table 3.1)

$\hat{f} = 0.1328 < \chi^2_{1,0.01} = 6.64$

The condition is satisfied. Consequently, the estimated state variables, $\hat{x}_1$, $\hat{x}_2$ and $\hat{x}_3$ can be accepted with 99% confidence.

### 4.1.2 *MATLAB Approach*



Voltage Data
Voltage Measurement
Voltage 1.020000 S.D. 0.01
Measured Bus
Bus

Voltage Data
Voltage Measurement
Voltage 0.920000 S.D. 0.05
Measured Bus
Bus

Power Injection Data
Power Injection Measurements
☐ Real S.D.
☑ Reactive 0.405000 S.D. 0.02
Measured Bus
Bus

Power Flow
Power Flow Measurements
☑ Real 0.598000 S.D. 0.015
☑ Reactive 0.305000 S.D. 0.02
Connecting Buses
From Bus
To Bus

Figure 4.2 2-bus Power System Network drawn in PET Software

## BEFORE DETECTION & REMOVAL OF BAD DATA

Number of measurement, Nm = 5

No of iterations = 5

Table 4.9 State Variables Before Detection & Removal of Bad Data

| Iteration count, k | $\hat{e}_{Nm}$ | | | | | $x_{Ns}$ | | |
|---|---|---|---|---|---|---|---|---|
| | $\hat{e}_1$ | $\hat{e}_2$ | $\hat{e}_3$ | $\hat{e}_4$ | $\hat{e}_5$ | $x_1$ | $x_2$ | $x_3$ |
| 0 | -0.0800 | 0.0200 | 0.4383 | 0.5980 | 0.3050 | -0.1495 | 0.9727 | 0.9961 |
| 1 | -0.0527 | 0.0239 | 0.3031 | 0.0207 | 0.3529 | -0.1721 | 0.9610 | 0.9870 |
| 2 | -0.0410 | 0.0330 | 0.2841 | -0.0518 | 0.3487 | -0.1756 | 0.9583 | 0.9847 |
| 3 | -0.0383 | 0.0353 | 0.2815 | -0.0615 | 0.3480 | -0.1761 | 0.9579 | 0.9843 |
| 4 | -0.0379 | 0.0357 | 0.2811 | -0.0628 | 0.3480 | -0.1762 | 0.9578 | 0.9843 |

$\therefore \hat{x}_1 = \delta_2 = -0.1762$ rad

$\hat{x}_2 = |V_2| = 0.9578$ p.u.

$\hat{x}_3 = |V_1| = 0.9843$ p.u.

## DETECTION & REMOVAL OF BAD DATA

$\hat{f} = 544.8152 > \chi^2_{2,0.01} = 9.21$

Bad data is detected,

Table 4.10 Standardized Errors Before Detection & Removal of Bad Data

| Nm | $\hat{e}_{Nm}$ | Standardized Errors |
|---|---|---|
| $z_1$ | -0.0378 | 5.7123 |
| $z_2$ | 0.0357 | 5.0407 |
| $z_3$ | 0.2810 | 20.5086 |
| $Z_4$ | -0.0630 | 22.6473 |
| $Z_5$ | 0.3480 | 23.3403 |

The largest standardized error 23.3403 at measurement no. 5, $Z_5$

$\therefore Z_5$ is removed from the measurement

## AFTER DETECTION & REMOVAL OF BAD DATA

Number of measurement, $N_m = 4$

No of iterations = 3

Table 4.11 Estimated State Variables After Detection & Removal of Bad Data

| **Iteration count, k** | $\hat{e}_{Nm}$ | | | | $x_{Ns}$ | | |
|---|---|---|---|---|---|---|---|
| | $\hat{e}_1$ | $\hat{e}_2$ | $\hat{e}_3$ | $\hat{e}_4$ | $x_1$ | $x_2$ | $x_3$ |
| 0 | -0.0800 | 0.0200 | 0.4383 | 0.5980 | -0.1495 | 0.9167 | 1.0235 |
| 1 | 0.0033 | -0.0035 | -0.0487 | 0.0390 | -0.1600 | 0.9222 | 1.0174 |
| 2 | -0.0022 | 0.0026 | -0.0027 | 0.0002 | -0.1600 | 0.9223 | 1.0174 |

$\hat{f} = 0.1360 < 6.6400$

No bad data is detected

$\therefore\ \hat{x}_1 = \delta_2 = -0.1600$ rad

$\hat{x}_2 = |V_2| = 0.9223$ p.u.

$\hat{x}_3 = |V_1| = 1.0174$ p.u.

## 4.2 Test and Analysis on 3-bus Power System Network

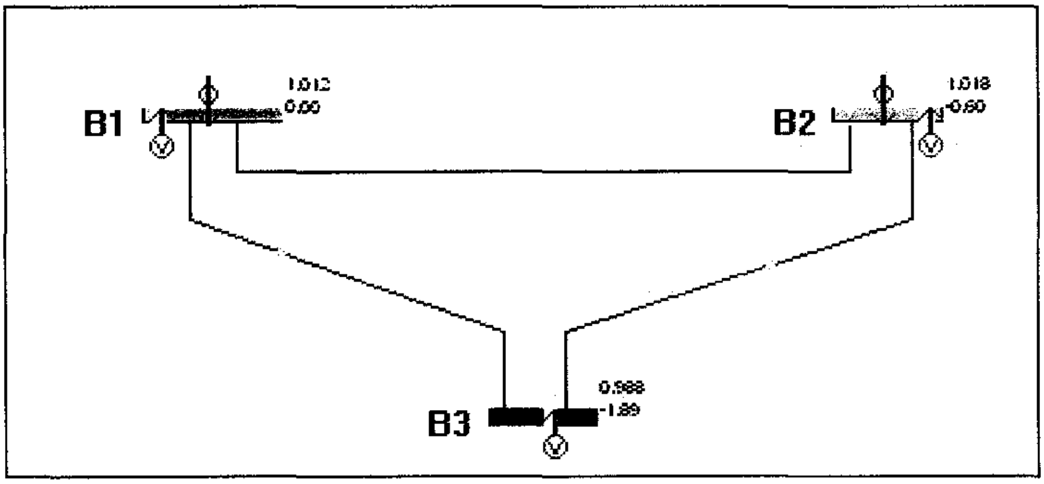


Figure 4.3 3-bus Power System Network drawn in PET Software

Referring to the Figure 4.2 above, a 3-bus system is installed with three voltmeters and four wattmeters in order to measure the seven quantities (Refer Table 4.13). In performing the State Estimation for this 3-bus system, these input data in the table below (Refer Table 4.12) are used in order to obtain the estimated state variables.

Table 4.12 Line Data for 3-Bus System (in p.u.)

| **Line No.** | **Between Buses** | **Line Impedances** |
|---|---|---|
| 1 | 1-2 | 0 + j0.25 |
| 2 | 1-3 | 0 + j0.08 |
| 3 | 2-3 | 0 + j0.05 |

Table 4.13 Measured Quantities on the Physical System Before Bad Data is Eliminated (in p.u.)

| **Bus & Line No.** | **Metering Equipment** | **Symbols** | **No. of measurement,** $z_{Nm}$ | **Measured Quantities** | **Variances,** $\sigma^2_{Nm}$ |
|---|---|---|---|---|---|
| Bus1,<br>voltage magnitude | voltmeter | $\lvert V_1 \rvert$ | $z_1$ | 1.01 | $(0.02)^2$ |
| Bus 2,<br>voltage magnitude | voltmeter | $\lvert V_2 \rvert$ | $z_2$ | 1.02 | $(0.02)^2$ |
| Bus 3,<br>voltage magnitude | voltmeter | $\lvert V_3 \rvert$ | $z_3$ | 0.98 | $(0.02)^2$ |
| Bus 1,<br>real power injection | wattmeter | $P_1$ | $z_4$ | 0.48 | $(0.05)^2$ |
| Bus 2,<br>real power injection | wattmeter | $P_2$ | $z_5$ | 0.33 | $(0.05)^2$ |
| Line $P_{13}$,<br>flow from bus 1 to bus 3 | wattmeter | $P_{13}$ | $z_6$ | 0.41 | $(0.05)^2$ |
| Line $P_{23}$,<br>flow from bus 2 to bus 3 | wattmeter | $P_{23}$ | $z_7$ | 0.38 | $(0.05)^2$ |

### *4.2.1 Results for Mathematical Approach*

**BEFORE DETECTION OF BAD DATA**

Number of measurement, Nm = 7

No of iteration = 1

Table 4.14 Estimated State Variables Before Detection of Bad Data

| $\hat{x}_{Ns}$ | **Flat-start Values** | **Estimated State Variables** |
|---|---|---|
| $\hat{x}_1 = \delta_2$ | 0 | – 0.01014 rad |
| $\hat{x}_2 = \delta_3,$ | 0 | – 0.03074 rad |
| $\hat{x}_3 = \|V_1\|$ | 1.0 | 1.01 p.u. |
| $\hat{x}_4 = \|V_2\|$ | 1.0 | 1.02 p.u. |
| $\hat{x}_5 = \|V_3\|$ | 1.0 | 0.98 p.u. |

**DETECTION OF BAD DATA**

Weighted Sum of Squares, $\hat{f} = 0.959618 < 9.21$

$\chi^2_{2,0.01} = 9.21$

As a result, the estimated state variables are accepted with 99% confidence. Hence, it is shown that for this 3-bus system, there is no presence of bad data in the measurements.

### *4.2.2 MATLAB Approach*

### BEFORE DETECTION OF BAD DATA

Number of measurement, Nm = 7

No of iterations = 1

Table 4.15 State Variables Before Detection of Bad Data

| **Iteration count, k** | $\hat{e}_{Nm}$ | | | | | $x_{Ns}$ | | |
|---|---|---|---|---|---|---|---|---|
| | $\hat{e}_1$ | $\hat{e}_2$ | $\hat{e}_3$ | $\hat{e}_4$ | $\hat{e}_5$ | $x_1$ | $x_2$ | $x_3$ |
| 0 | -0.0800 | 0.0200 | 0.4383 | 0.5980 | 0.3050 | -0.1495 | 0.9727 | 0.9961 |

$\hat{x}_1 = \delta_{2,} = -0.01229$ rad

$\hat{x}_2 = \delta_{3,} = -0.03182$ rad

$\hat{x}_3 = |V_1|, = 1.03435$ p.u.

$\hat{x}_4 = |V_2| = 0.97804$ p.u.

$\hat{x}_5 = |V_3| = 0.99589$ p.u.

### DETECTION OF BAD DATA

$\hat{f} = 0.01109947 < 9.21$

State estimation calculation can be processed

No bad data is detected

$\therefore\ \hat{x}_1 = \delta_{2,} = -0.01229$ rad

$\hat{x}_2 = \delta_{3,} = -0.03182$ rad

$\hat{x}_3 = |V_1|, = 1.03435$ p.u.

$\hat{x}_4 = |V_2| = 0.97804$ p.u.

$\hat{x}_5 = |V_3| = 0.99589$ p.u.

## 4.3 Test and Analysis on 24-bus Power System Network

A 24-bus system is installed with 117 measurements in the system network including the bus voltages, real and reactive power injection as well as real and reactive power flow. In performing the State Estimation for this 24-bus system, these input data in the table below are used in order to obtain the estimated state variables. Two case studies are done on this bus system; without bad data and with bad data.

### *4.3.1 Case 1: State Estimation with No Bad Data*

**BEFORE DETECTION OF BAD DATA**

Number of measurement, Nm = 117

No of iterations = 3

$\hat{f} = 5.536630 < 100.4251$

State estimation calculation can be proceed.

Table 4.16 State Variables After Detection of Bad Data

| Bus | $V$(pu) | $\delta$(rad) | Bus | $V$(pu) | $\delta$(rad) |
|---|---|---|---|---|---|
| 1 | 1.03500 | 0.00 | 13 | 0.96711 | -0.23688 |
| 2 | 1.03500 | -0.03138 | 14 | 0.98000 | -0.20616 |
| 3 | 0.95897 | -0.20329 | 15 | 1.01405 | 0.00073 |
| 4 | 0.96782 | -0.20563 | 16 | 1.01702 | -0.02002 |
| 5 | 0.98349 | -0.16617 | 17 | 1.03874 | 0.06429 |
| 6 | 0.96431 | -0.29579 | 18 | 1.05001 | 0.08598 |
| 7 | 1.02500 | -0.31405 | 19 | 1.02225 | -0.04609 |
| 8 | 0.97178 | -0.36699 | 20 | 1.03909 | -0.03959 |
| 9 | 0.96699 | -0.27987 | 21 | 1.04993 | 0.10317 |
| 10 | 0.98300 | -0.29097 | 22 | 1.05000 | 0.20147 |
| 11 | 0.97085 | -0.23979 | 23 | 1.04994 | -0.02238 |
| 12 | 0.97106 | -0.21514 | 24 | 0.97900 | -0.06785 |

## AFTER DETECTION OF BAD DATA

Table 4.17 Comparison of Estimated & Measured Bus Voltages

| Bus Number | Estimated Values | Measurements |
|---|---|---|
| V1 | 1.035 | 1.035 |
| V2 | 1.035 | 1.035 |
| V5 | 0.98349 | 0.9835 |
| V7 | 1.025 | 1.025 |
| V9 | 0.96699 | 0.967 |
| V10 | 0.983 | 0.983 |
| V12 | 0.97106 | 0.971 |
| V14 | 0.98 | 0.98 |
| V15 | 1.01405 | 1.014 |
| V16 | 1.01702 | 1.017 |
| V18 | 1.05001 | 1.05 |
| V19 | 1.02225 | 1.0222 |
| V21 | 1.04993 | 1.05 |
| V22 | 1.05 | 1.05 |
| V23 | 1.04994 | 1.05 |
| V24 | 0.979 | 0.979 |

Table 4.18 Comparison of Estimated & Measured Real Power Injection

| Bus Number | Estimated Values | Measurements |
|---|---|---|
| P1 | 5.40719 | 5.5687 |
| P2 | 0.52678 | 0.653 |
| P3 | -1.94184 | -1.98 |
| P4 | -0.74463 | -0.814 |
| P5 | -0.67346 | -0.781 |
| P6 | -1.45218 | -1.496 |
| P7 | 1.02559 | 1.025 |
| P8 | -1.87931 | -1.881 |
| P15 | -1.95199 | -1.337 |
| P16 | -0.3501 | 0.45 |
| P18 | 0.33492 | 0.337 |
| P20 | -1.38575 | -1.408 |
| P22 | 2.99996 | 3 |
| P23 | 6.58729 | 6.6 |

Table 4.19 Comparison of Estimated & Measured Reactive Power Injection

| Bus Number | Estimated Values | Measurements |
|---|---|---|
| Q2 | 0.67091 | 0.65952 |
| Q3 | -0.4809 | -0.407 |
| Q4 | -0.19694 | -0.165 |
| Q5 | -0.1921 | -0.154 |
| Q6 | -0.33768 | -0.308 |
| Q7 | 0.63678 | 0.63363 |
| Q8 | -0.38267 | -0.385 |
| Q16 | 0.2179 | 0.12207 |
| Q20 | -0.10308 | -0.286 |
| Q21 | 1.03541 | 1.0609 |
| Q23 | 2.28701 | 2.438 |

Table 4.20 Comparison of Estimated & Measured Real Power Flow

| Line Number | Estimated Values | Measurements |
|---|---|---|
| P1-2 | 2.34315 | 2.343 |
| P1-3 | 1.0031 | 1.0481 |
| P1-5 | 2.06095 | 2.1772 |
| P2-4 | 1.44675 | 1.5229 |
| P2-6 | 1.40946 | 1.4599 |
| P9-3 | -0.5374 | -0.5431 |
| P24-3 | 1.54215 | 1.5503 |
| P9-4 | -0.62334 | -0.6233 |
| P10-6 | 0.13803 | 0.1381 |
| P7-8 | 1.02559 | 1.025 |
| P11-9 | 0.46291 | 0.4679 |
| P12-9 | 0.747 | 0.7563 |
| P12-10 | 0.87145 | 0.8759 |
| P13-11 | 0.04678 | 0.0553 |
| P14-11 | 0.78166 | 1.005 |
| P13-12 | -0.42801 | -0.431 |
| P23-12 | 2.12208 | 2.122 |
| P21-22 | -1.56017 | -1.56 |
| P5-10 | 1.29961 | 1.298 |

| Line Number | Estimated Values | Measurements |
|---|---|---|
| P8-9 | -0.45212 | -0.453 |
| P8-10 | -0.42382 | -0.425 |
| P10-11 | -0.58595 | -0.585 |
| P23-13 | 2.61913 | 2.6174 |
| P14-16 | -3.13647 | -3.139 |
| P15-16 | 1.19689 | 1.2117 |
| P15-21 | -4.53076 | -2.0584 |
| P15-21 | -4.53076 | -2.0584 |
| P24-15 | -1.36746 | -1.5503 |
| P16-17 | -3.47034 | -3.0854 |
| P16-19 | 1.12616 | 1.5535 |
| P17-18 | -1.71524 | -1.7181 |
| P17-22 | -1.39527 | -1.3953 |
| P21-18 | 1.38872 | 0.6948 |
| P21-18 | 1.38872 | 0.6948 |
| P20-19 | 0.45512 | 0.2234 |
| P20-19 | 0.45512 | 0.2234 |
| P20-23 | -1.84087 | -0.9274 |
| P20-23 | -1.84087 | -0.9274 |

Table 4.21 Comparison of Estimated & Measured Reactive Power Flow

| Line Number | Estimated Values | Measurements |
|---|---|---|
| Q1-2 | -0.64699 | -0.647 |
| Q1-3 | 0.1794 | 0.165 |
| Q1-5 | 0.25289 | 0.2434 |
| Q2-4 | 0.27554 | 0.2679 |
| Q2-6 | 0.169 | 0.1648 |
| Q9-3 | 0.21204 | 0.1968 |
| Q24-3 | 0.29904 | 0.4461 |
| Q9-4 | 0.16505 | 0.167 |
| Q10-6 | -0.91596 | -0.9186 |
| Q7-8 | 0.63678 | 0.6336 |
| Q11-9 | 0.0425 | 0.313 |
| Q12-9 | 0.05218 | 0.3886 |
| Q12-10 | -0.13094 | 0.0249 |
| Q13-11 | -0.12789 | -0.057 |
| Q14-11 | 0.0843 | 0.1904 |
| Q13-12 | -0.06834 | -0.101 |
| Q23-12 | 0.66855 | 0.6712 |
| Q21-22 | 0.19901 | 0.2 |
| Q5-10 | -0.25647 | -0.2663 |
| Q8-9 | 0.14562 | 0.1443 |
| Q8-10 | 0.03922 | 0.0354 |
| Q10-11 | 0.17572 | 0.08 |
| Q23-13 | 0.83821 | 0.8611 |
| Q14-16 | -0.09662 | -0.0956 |
| Q15-16 | -0.33216 | -0.3355 |
| Q15-21 | -0.72756 | -0.4402 |
| Q15-21 | -0.72756 | -0.4402 |
| Q24-15 | -0.49186 | -0.4461 |
| Q16-17 | -0.33441 | -0.3961 |
| Q16-19 | -0.38662 | -0.4247 |
| Q17-18 | -0.59656 | -0.6225 |
| Q17-22 | 0.04578 | 0.0422 |
| Q21-18 | -0.19455 | -0.1092 |
| Q21-18 | -0.19455 | -0.1092 |
| Q20-19 | 0.78129 | 0.3382 |
| Q20-19 | 0.78129 | 0.3382 |
| Q20-23 | -0.81483 | -0.4812 |
| Q20-23 | -0.81483 | -0.4812 |

### 4.3.2 *Case 2: State Estimation with Bad Data*

**BEFORE DETECTION & REMOVAL OF BAD DATA**

Number of measurement, Nm = 117

No. of iterations = 4

Table 4.22 Estimated State Variables Before Detection & Removal of Bad Data

| Bus Number | $V$ (p.u.) | $\delta$(rad) |
|---|---|---|
| 1 | 1.03500 | 0.00 |
| 2 | 1.03500 | -0.03138 |
| 3 | 0.95897 | -0.20329 |
| 4 | 0.96782 | -0.20563 |
| 5 | 0.98349 | -0.16617 |
| 6 | 0.96431 | -0.29579 |
| 7 | 1.02500 | -0.31405 |
| 8 | 0.97178 | -0.36699 |
| 9 | 0.96699 | -0.27987 |
| 10 | 0.98300 | -0.29097 |
| 11 | 0.97085 | -0.23979 |
| 12 | 0.97106 | -0.21514 |
| 13 | 0.96711 | -0.23688 |
| 14 | 0.98000 | -0.20616 |
| 15 | 1.01405 | 0.00073 |
| 16 | 1.01702 | -0.02002 |
| 17 | 1.03874 | 0.06429 |
| 18 | 1.05001 | 0.08598 |
| 19 | 1.02225 | -0.04609 |
| 20 | 1.03909 | -0.03959 |
| 21 | 1.04993 | 0.10317 |
| 22 | 1.05000 | 0.20147 |
| 23 | 1.04994 | -0.02238 |
| 24 | 0.97900 | -0.06785 |

**DETECTION & REMOVAL OF BAD DATA**

$\hat{f} = 160.2185 > 100.4251$

State estimation calculation cannot be processed

The largest standardized error 11.68246 at measurement no. 19, $Z_{19} = P_3$ (Bad data)

$\therefore$ $Z_{19}$ is removed from the measurement

## AFTER DETECTION & REMOVAL OF BAD DATA

Number of measurement, $N_m = 116$

No of iterations = 3

$\hat{f} = 5.538042 < 99.2274$

Bad data at P3 is eliminated;

$\therefore$ No bad data is detected

Table 4.23 Estimated State Variables After Detection & Removal of Bad Data

| **Bus Number** | **$V$ (p.u.)** | **$\delta$(rad)** |
|---|---|---|
| 1 | 1.03500 | 0.00 |
| 2 | 1.03500 | -0.03138 |
| 3 | 0.95887 | -0.20329 |
| 4 | 0.96783 | -0.20563 |
| 5 | 0.98349 | -0.16617 |
| 6 | 0.96432 | -0.29579 |
| 7 | 1.02500 | -0.31405 |
| 8 | 0.97178 | -0.36699 |
| 9 | 0.96699 | -0.27987 |
| 10 | 0.98300 | -0.29097 |
| 11 | 0.97085 | -0.23979 |
| 12 | 0.97106 | -0.21514 |
| 13 | 0.96711 | -0.23688 |
| 14 | 0.98000 | -0.20597 |
| 15 | 1.01405 | 0.00091 |
| 16 | 1.01702 | -0.01984 |
| 17 | 1.03874 | 0.06429 |
| 18 | 1.05001 | 0.08598 |
| 19 | 1.02225 | -0.04609 |
| 20 | 1.03909 | -0.03959 |
| 21 | 1.04993 | 0.10317 |
| 22 | 1.05000 | 0.20147 |
| 23 | 1.04994 | -0.02238 |
| 24 | 0.97900 | -0.06785 |

## 4.4 Overall Analysis on 2-bus, 3-bus and 24-bus Power System Network

A mathematical approach is used for smaller bus system network in proving the reliability of the simulation on the State Estimation of Power System and in achieving consistent final results by using the developed MATLAB program. Whereas, the results of Load Flow Analysis from PSAT Software is used to be compared with the final results of a larger scale bus system network using MATLAB software. In verifying the adequacy of the software, it is tested with available measurements which are 2-bus, 3-bus and 24-bus system. Finally, the success of the bus system networks simulation is verified using the software.

In calculation, the state variables are estimated, while the measurements are calculated for the purpose of comparing the state variables with the MATLAB results. For these 2-bus, 3-bus and 24-bus system, a 99% confidence interval which has the probability of (1 – 0.01) or known as a specified probability, $\alpha = 0.01$ is used. In the analysis on the 2-bus system, it is observed that there is only one bad data presence after detection of bad data which is at the fifth measurement, $z_5$. Whereas, there is no bad data present when testing the 3-bus system. As for the 24-bus system, there are two cases where the first one shows the measurements without the presence of bad data. While, the other case shows the measurements with the presence of bad data which occurs at measurement number 19, $P_3$.

The differences between estimated state variables and state estimation achieved from the calculated and PSAT values are observed to be consistent even though the software is simulated more than once. In addition, the final results tallied with the measurements, where measured voltage buses are quite similar to its state variables. In addition, the measurements of real and reactive power injections are also quite similar with its calculated values using the real and reactive power equation. Moreover, the magnitude voltages can be used in determining the real and reactive power flow using the Power Flow Equation. The comparison of estimated values and measurements can be referred in the appendices.

# CHAPTER 5

# CONCLUSION AND RECOMMENDATIONS

This project, State Estimation of Power System is significant enough to be developed by next FYP students. The initial idea might be simple, but it surely can be elaborated more and expanded from time to time. The basic concept of State Estimation of Power Systems and the differences between State Estimation and Power Flow Analysis should be totally understandable and clarified so that the work progress is constantly on the right track.

Then, the MATLAB program is developed to test the State Estimation calculation in the Power System Network. Besides, the software is able to identify and remove the bad data and also in producing the estimated state variables, which are magnitude, $|V|$, and phase angles, $\delta$, of bus voltages consistently.

To conclude, the steps involved in estimating the state variables using WLS method as well as detection and removal of bad data are well understood. Lastly, the presence of bad data in the measurements affects the reliability of the state estimation values.

For recommendation, further studies on implementing the State Estimation on a larger scale network should be done, for example, 4 and 14-bus system. Then, the observability has to be maintained and lastly, studies on the factor of weighting matrix can be conducted.

# REFERENCES

[1] John J. Grainger, William D. Stevenson, Jr., "Power System Analysis" *McGraw-Hill Inc., 1994*

[2] Ali Abur and Antonio Gómez Expósito, Marcel Dekker, Inc., "Power System State Estimation— Theory and Implementations" *2004*

[3] ROBERT E. LARSON, WILLIAM F. TINNEY AND JOHN PESCHON, "State Estimation in Power Systems - Part I: Theory and Feasibility"

[4] E.Handscin, F.C.Schwepe, J. Kohlas, and A.Fiecher, " Bad Data Analysis for Power System State estimation", *IEEE Transactions on Power Systems, PAS 94.no2.pp.329-336,1975*

[5] Amit Jain, Yoshiyuki Kawakazoe, "Network Observability: A Solution Techniques Using Neural Networks", *Institute or Material Research (IMR),Tohoku University, Japan pp1*

[6] Ali Abur," Power System State Estimation, Chapter 2: Weight Least Squares State Estimation *,pp.19.*

# APPENDICES

# APPENDIX A

# MATLAB CODING

# MAIN PROGRAM

```
%this is the main state estimate program
clear all;
clc;
global nbus;
global nlines;
global nshunts;
global tolerance;
global confidence;

nbus=input('Please enter the number of buses...');
nlines=input('Please enter the number of lines...');
nshunts=input('Please enter the number of shunts...');
tolerance=input('Please enter the tolerance...');
disp('Please enter the level of confidence for');
confidence=input('the purpose of bad data detection...');
disp('Please select the followings:');
disp('1-Input data');
disp('2-Run program');
choice=input('to continue the programming.........');
if choice==1
   nbus=input('Please enter the number of buses...');
   nlines=input('Please enter the number of lines...');
   nshunts=input('Please enter the number of shunts...');
   tolerance=input('Please enter the tolerance...');
   disp('Please enter the level of confidence for');
   confidence=input('the purpose of bad data detection...');
   dataentry(choice,nbus,nlines,nshunts,tolerance,confidence);
   %nbus=x(1);
   %nlines=x(2);
   %nshunts=x(3);
   %tolerance=x(4);
end
if choice==2
   programengine(choice,nbus,nlines,nshunts,tolerance,confidence);
end
%if choice==3
 %  viewoutput(choice);
 %end
```

# DATA ENTRY

```
function x=dataentry(choice,nbus,nlines,nshunts,tolerance,confidence);
% nbus=input('Please enter the number of buses...');
% nlines=input('Please enter the number of lines...');
% nshunts=input('Please enter the number of shunts...');
% tolerance=input('Please enter the tolerance...');
nbus=nbus;
nlines=nlines;
nshunts=nshunts;
tolerance=tolerance;
confidence=confidence;
disp('Select the data to be modified:');
disp('1-Bus data');
disp('2-Line data');
disp('3-Shunt data');
disp('4-All data');
select=input('enter the data to be modified...');

if select==1
%Entering bus data
disp('The following code is employed for the status of the buses.');
disp('0-Slack Bus');
disp('1-Buses of PV or PQ type');
disp('You will now be asked data pertaining to initial conditions of each bus.');
bus_status=[];
v_bus=[];
ang_bus=[];
%weight variable for voltage magnitudes and power injections
%funda of wts: if the value is somewhat near desired then wts are low, else
%if values drop very low then wts have to be very high. no measure implies
%very high weights
v_wts=[];
p_wts=[];
q_wts=[];
bus_number=[];
for i=1:nbus
   bus_number=[bus_number i};
   disp('This is the data for bus');
   disp(i);
   bus_status(i)=input('Bus status:');
   disp('Please enter the starting conditions:');
   v_bus(i)=input('Initial voltage:');
   ang_bus(i)=input('Initial angle:');
   disp('Please enter the measured values:');
   v_meas(i)=input('Please enter the measured value of voltage:');
   v_wts(i)=input('Enter the weight for this voltage measure:');
   p_meas(i)=input('Please enter the measured value of real power injection:');
   p_wts(i)=input('Enter the weight for this real power injection measure:');
   q_meas(i)=input('Please enter the measured value of reactive power injection:');
   q_wts(i)=input('Enter the weight for this reactive power injection measure:');
end
a=[bus_status' v_bus' ang_bus' v_meas' p_meas' q_meas'];
a=a';
fid=fopen('busdata.txt','w');
fprintf(fid,'%6.2f %6.4f %6.2f %6.4f %6.4f %6.4`f\n',a);
fclose(fid);
%writing weights data into file
buswts=[v_wts' p_wts' q_wts'];
buswts=buswts';
fid=fopen('busweightsdata.txt','w');
fprintf(fid,'%6.5f %6.5f %6.5f',buswts);
fclose(fid);
end

if select==2
%Entering line data
disp('You will now be asked to enter line data.');
start_bus=[];
end_bus=[];
resistance=[];
reactance=[];
shunt_admit=[];
```

```
tap=[]; %This refers only to the tap magnitude
%line flows
pij_flow=[];
qij_flow=[];
pji_flow=[];
qji_flow=[];
%weights for line flows..same fundas apply
pij_wts=[];
qij_wts=[];
pji_wts=[];
qji_wts=[];
for i=1:nlines
   fprintf('This is data for line %d',i);
   start_bus(i)=input('Enter the starting bus:');
   end_bus(i)=input('Enter the ending bus:');
   resistance(i)=input('Enter the resistance of the line:');
   reactance(i)=input('Enter the reactance of the line:');
   shunt_admit(i)=input('Enter the shunt admittance of the line:');
   tap(i)=input('Enter the tap of the line (only magnitude):');
   fprintf('Line flow from bus %d to bus %d',start_bus(i),end_bus(i));
   pij_flow(i)=input('Enter the real power flow on the line:');
   pij_wts(i)=input('Enter the weights for this measure:');
   qij_flow(i)=input('Enter the reactive power flow on the line:');
   qij_wts(i)=input('Enter the weights for this measure:');
   fprintf('Line flow from bus %d to bus %d',end_bus(i),start_bus(i));
   pji_flow(i)=input('Enter the real power flow (reverse)on the line:');
   pji_wts(i)=input('Enter the weights for this measure:');
   qji_flow(i)=input('Enter the reactive power flow (reverse) on the line:');
   qji_wts(i)=input('Enter the weights for this measure:');
end
b=[start_bus' end_bus' resistance' reactance' shunt_admit' tap' pij_flow' qij_flow' pji_flow' qji_flow'];
%b=[start_bus' end_bus' resistance' reactance' shunt_admit' tap' pij_flow' qij_flow'];
b=b';
fid=fopen('linedata.txt','w');
fprintf(fid,'%6.2f %6.2f %6.4f %6.4f %6.4f %6.2f %6.4f %6.4f %6.4f %6.4f\n',b);
%fprintf(fid,'%6.2f %6.2f %6.4f %6.4f %6.4f %6.2f %6.4f %6.4f\n',b);
fclose(fid);
linewts=[pij_wts' qij_wts' pji_wts' qji_wts'];
%linewts=[pij_wts' qij_wts'];
linewts=linewts';
fid=fopen('lineweightsdata.txt','w');
fprintf(fid,'%6.5f %6.5f %6.5f %6.5f',linewts);
%fprintf(fid,'%6.4f %6.4f',linewts);
fclose(fid);
end

if select==3
%Entering shunt data
shunts=[];
bus_shunt=[];
for i=1:nshunts
   bus_shunt(i)=input('Enter the bus number of the shunt:');
   shunts(i)=input('Enter the shunt magnitude:');
end
c=[bus_shunt' shunts'];
c=c';
fid=fopen('shuntdata.txt','w');
fprintf(fid,'%6.4f %6.4f\n',c);
fclose(fid);
end

if select==4
%Entering bus data
disp('The following code is employed for the status of the buses.');
disp('0-Slack Bus');
disp('1-Buses of PV or PQ type');
disp('You will now be asked data pertaining to initial conditions of each bus.');
bus_status=[];
v_bus=[];
ang_bus=[];
%weight variable for voltage magnitudes and power injections
%funda of wts: if the value is somewhat near desired then wts are low, else
%if values drop very low then wts have to be very high. no measure implies
%very high weights
```

```
v_wts=[];
p_wts=[];
q_wts=[];
bus_number=[];
for i=1:nbus
   bus_number=[bus_number i];
   disp('This is the data for bus');
   disp(i);
   bus_status(i)=input('Bus status:');
   disp('Please enter the starting conditions:');
   v_bus(i)=input('Initial voltage:');
   ang_bus(i)=input('Initial angle:');
   disp('Please enter the measured values:');
   v_meas(i)=input('Please enter the measured value of voltage:');
   v_wts(i)=input('Enter the weight for this voltage measure:');
   p_meas(i)=input('Please enter the measured value of real power injection:');
   p_wts(i)=input('Enter the weight for this real power injection measure:');
   q_meas(i)=input('Please enter the measured value of reactive power injection:');
   q_wts(i)=input('Enter the weight for this reactive power injection measure:');
end
a=[bus_status' v_bus' ang_bus' v_meas' p_meas' q_meas'];
a=a';
fid=fopen('busdata.txt','w');
fprintf(fid,'%6.2f %6.4f %6.2f %6.4f %6.4f %6.4`f\n',a);
fclose(fid);
%writing weights data into file
buswts=[v_wts' p_wts' q_wts'];
buswts=buswts';
fid=fopen('busweightsdata.txt','w');
fprintf(fid,'%6.5f %6.5f %6.5f',buswts);
fclose(fid);

%Entering line data
disp('You will now be asked to enter line data.');
start_bus=[];
end_bus=[];
resistance=[];
reactance=[];
shunt_admit=[];
tap=[]; %This refers only to the tap magnitude
%line flows
pij_flow=[];
qij_flow=[];
pji_flow=[];
qji_flow=[];
%weights for line flows..same fundas apply
pij_wts=[];
qij_wts=[];
pji_wts=[];
qji_wts=[];
for i=1:nlines
   fprintf('This is data for line %d',i);
   start_bus(i)=input('Enter the starting bus:');
   end_bus(i)=input('Enter the ending bus:');
   resistance(i)=input('Enter the resistance of the line:');
   reactance(i)=input('Enter the reactance of the line:');
   shunt_admit(i)=input('Enter the shunt admittance of the line:');
   tap(i)=input('Enter the tap of the line (only magnitude):');
   fprintf('Line flow from bus %d to bus %d',start_bus(i),end_bus(i));
   pij_flow(i)=input('Enter the real power flow on the line:');
   pij_wts(i)=input('Enter the weights for this measure:');
   qij_flow(i)=input('Enter the reactive power flow on the line:');
   qij_wts(i)=input('Enter the weights for this measure:');
   fprintf('Line flow from bus %d to bus %d',end_bus(i),start_bus(i));
   pji_flow(i)=input('Enter the real power flow (reverse)on the line:');
   pji_wts(i)=input('Enter the weights for this measure:');
   qji_flow(i)=input('Enter the reactive power flow (reverse) on the line:');
   qji_wts(i)=input('Enter the weights for this measure:');
end
b=[start_bus' end_bus' resistance' reactance' shunt_admit' tap' pij_flow' qij_flow' pji_flow' qji_flow'];
%b=[start_bus' end_bus' resistance' reactance' shunt_admit' tap' pij_flow' qij_flow'];
b=b';
fid=fopen('linedata.txt','w');
fprintf(fid,'%6.2f %6.2f %6.4f %6.4f %6.4f %6.2f %6.4f %6.4f %6.4f %6.4f\n',b);
```

```
%fprintf(fid,'%6.2f %6.2f %6.4f %6.4f %6.4f %6.2f %6.4f %6.4f\n',b);
fclose(fid);
linewts=[pij_wts' qij_wts' pji_wts' qji_wts'];
%linewts=[pij_wts' qij_wts'];
linewts=linewts';
fid=fopen('lineweightsdata.txt','w');
fprintf(fid,'%6.4f %6.4f %6.4f %6.4f',linewts);
%fprintf(fid,'%6.4f %6.4f',linewts);
fclose(fid);

%Entering shunt data
shunts=[];
bus_shunt=[];
for i=1:nshunts
   bus_shunt(i)=input('Enter the bus number of the shunt:');
   shunts(i)=input('Enter the shunt magnitude:');
end
c=[bus_shunt' shunts'];
c=c';
fid=fopen('shuntdata.txt','w');
fprintf(fid,'%6.4f %6.4f\n',c);
fclose(fid);
end
%x=[nbus nlines nshunts tolerance confidence];
```

# PROGRAM ENGINE

```
function x1=programengine(choice,nbus,nlines,nshunts,tolerance,confidence);
nbus=nbus;
nlines=nlines;
nshunts=nshunts;
tolerance=tolerance;
% confidence=0.01; %tak yah bubuh kalau original command

%reading data related to buses
fid=fopen('busdata.txt','r');
a=textread('busdata.txt');
fclose(fid);

%reading data pertaining to lines
fid=fopen('linedata.txt','r');
b=textread('linedata.txt');
fclose(fid);

%reading data pertaining to shunts
fid=fopen('shuntdata.txt','r');
c=textread('shuntdata.txt');
fclose(fid);

%reading data pertaining to admittances bus to ground
fid=fopen('bustogrounddata.txt','r');
d=textread('bustogrounddata.txt');
fclose(fid);

%reading data pertaining to weights
fid=fopen('busweightsdata.txt','r');
buswts=textread('busweightsdata.txt');
fclose(fid);

fid=fopen('lineweightsdata.txt','r');
linewts=textread('lineweightsdata.txt');
fclose(fid);

%reading data pertaining to chi-square distribution
fid=fopen('chisquare.txt','r');
chi2=textread('chisquare.txt');
fclose(fid);

%reading data pertaining to load flow data
fid=fopen('loadflow.txt','r');
loadflow=textread('loadflow.txt');
fclose(fid);

%reading data pertaining to load flow data
fid=fopen('powerflow.txt','r');
powerflow=textread('powerflow.txt');
fclose(fid);


act_v=loadflow(:,1);
act_ang=loadflow(:,2);
p_inj=loadflow(:,3);
q_inj=loadflow(:,4);
pij_act=powerflow(:,1);
qij_act=powerflow(:,2);
pji_act=powerflow(:,3);
qji_act=powerflow(:,4);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%

v_wts1=buswts(:,1);
p_wts1=buswts(:,2);
q_wts1=buswts(:,3);
pij_wts1=linewts(:,1);
qij_wts1=linewts(:,2);
pji_wts1=linewts(:,3);
qji_wts1=linewts(:,4);
```

```
v_wts=v_wts1';
p_wts=p_wts1';
q_wts=q_wts1';
pij_wts=pij_wts1';
qij_wts=qij_wts1';
pji_wts=pji_wts1';
qji_wts=qji_wts1';

%%%%%%%%%%%%%%%%%%%%%%%%STATE ESTIMATION
PROGRAM%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Formation of Ybus
Ybus=zeros(nbus);
tap=b(:,6);
x=b(:,1);
y=b(:,2);
zreal=b(:,3);
zimag=b(:,4);
y1=b(:,5)*sqrt(-1);
% tlsuscep=b(:,5);
for i=1:nlines
   a1=x(i,:);
   b1=y(i,:);
   line_charging(a1,b1)=y1(i,:);
end
line_charging;
Gline=real(line_charging);
Bline=imag(line_charging);
% mags=abs(line_charging);
if length(c)~=0
   shun_bus=c(:,1);
   shun_res=c(:,2);
   shun_reac=c(:,3);
   for i=1:size(c)
   Yshun(i)=shun_res(i)+shun_reac(i)*sqrt(-1);
   end
end

if length(d)~=0
   add_bus=d(:,1);
   add_res=d(:,2);
   add_reac=d(:,3);
   for i=1:size(d)
   Yadd(i)=(1/(add_res(i)+add_reac(i)*sqrt(-1)));
   end
end


%effect of line impedances and shunt admittances
 for i=1:length(x)
   tap_effect(x(i),x(i))=(1/(zreal(i)+zimag(i)*sqrt(-1)))/(tap(i)^2);
   tap_effect(x(i),y(i))=(-1/(zreal(i)+zimag(i)*sqrt(-1)))/(conj(tap(i)));
   tap_effect(y(i),x(i))=(-1/(zreal(i)+zimag(i)*sqrt(-1)))/(tap(i));
   tap_effect(y(i),y(i))=(1/(zreal(i)+zimag(i)*sqrt(-1)));
   Ybus(x(i),x(i))=Ybus(x(i),x(i))+tap_effect(x(i),x(i))+0.5*line_charging(x(i),y(i));
   Ybus(y(i),y(i))=Ybus(y(i),y(i))+tap_effect(y(i),y(i))+0.5*line_charging(x(i),y(i));
   Ybus(x(i),y(i))=Ybus(x(i),y(i))+tap_effect(x(i),y(i));
   Ybus(y(i),x(i))=Ybus(y(i),x(i))+tap_effect(y(i),x(i));
 end
Ybus;

%inclusion of Y to ground admittances
if size(d)~=0
   for i=1:length(add_bus)
      Ybus(add_bus(i),add_bus(i))=Ybus(add_bus(i),add_bus(i))+Yadd(i);
   end
end
%inclusion of shunts
if size(c)~=0
   for i=1:length(shun_bus)
      Ybus(shun_bus(i),shun_bus(i))=Ybus(shun_bus(i),shun_bus(i))+Yshun(i);
   end
end
```

```
format long e,Ybus;
theta1=angle(Ybus);
theta=theta1*180/pi;


mag=abs(Ybus);
%separation into G and B
G=real(Ybus);
B=imag(Ybus);

%computation of measurement mismatch
%the measurement mismatch vector is of the form [V Pi Qi Pij Qij Pji Qji]' where each
%term represents a subvector

%state mismatch vector
%state mismatch vector is of the form [theta V] where term represents a
%subvector and the theta subvector excludes the slack bus angle


v_bus=a(:,2);
ang_busr=a(:,3);
v_meas=a(:,4);
p_meas=a(:,5);
q_meas=a(:,6);
pij_flow=b(:,7);
qij_flow=b(:,8);
pji_flow=b(:,9);
qji_flow=b(:,10);
% ang_bus=ang_busr*180/pi;

%%%%%%%%%big looping starting count%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:length(v_meas)
  if abs((v_meas(i,:)-act_v(i,:))/act_v(i,:))==1
    v_meas(i,:)=v_meas(i,:);
  end
  if abs((v_meas(i,:)-act_v(i,:))/act_v(i,:))<0.5
     v_meas(i,:)=v_meas(i,:);
  end
  if abs((v_meas(i,:)-act_v(i,:))/act_v(i,:))>0.5 & abs((v_meas(i,:)-act_v(i,:))/act_v(i,:))~=1
    v_meas(i,:)=20;
  end
end
for i=1:length(p_meas)
  if p_inj(i,:)~=0
  if abs((p_meas(i,:)-p_inj(i,:))/p_inj(i,:))==1
    p_meas(i,:)=p_meas(i,:);
  end
  if abs((p_meas(i,:)-p_inj(i,:))/p_inj(i,:))<0.5
     p_meas(i,:)=p_meas(i,:);
  end
  if abs((p_meas(i,:)-p_inj(i,:))/p_inj(i,:))>0.5 & abs((p_meas(i,:)-p_inj(i,:))/p_inj(i,:))~=1
    p_meas(i,:)=20;
  end
  end
  if p_inj(i,:)==0
  if abs(p_meas(i,:))==0
    p_meas(i,:)=p_meas(i,:);
  end
  if abs(p_meas(i,:))>0.5
     p_meas(i,:)=20;
  end
  end
end
p_meas;
for i=1:length(q_meas)
  if q_inj(i,:)~=0
  if abs((q_meas(i,:)-q_inj(i,:))/q_inj(i,:))==1
    q_meas(i,:)=q_meas(i,:);
  end
  if abs((q_meas(i,:)-q_inj(i,:))/q_inj(i,:))<0.5
     q_meas(i,:)=q_meas(i,:);
  end
  if abs((q_meas(i,:)-q_inj(i,:))/q_inj(i,:))>0.5 & abs((q_meas(i,:)-q_inj(i,:))/q_inj(i,:))~=1
    q_meas(i,:)=20;
```

```
  end
  end
  if q_inj(i,:)==0
     if abs(q_meas(i,:))==0
        q_meas(i,:)=q_meas(i,:);
     end
     if abs(q_meas(i,:))>0.5
        q_meas(i,:)=20;
     end
  end

end
for i=1:length(pij_flow)
  if pij_act(i,:)~=0
  if abs((pij_flow(i,:)-pij_act(i,:))/pij_act(i,:))==1
     pij_flow(i,:)=pij_flow(i,:);
  end
  if abs((pij_flow(i,:)-pij_act(i,:))/pij_act(i,:))<0.5
      pij_flow(i,:)=pij_flow(i,:);
  end
  if abs((pij_flow(i,:)-pij_act(i,:))/pij_act(i,:))>0.5 & abs((pij_flow(i,:)-pij_act(i,:))/pij_act(i,:))~=1
     pij_flow(i,:)=20;
  end
  end
  if pij_act(i,:)==0
     if abs(pij_flow(i,:))==0
        pij_flow(i,:)=pij_flow(i,:);
     end
     if abs(pij_flow(i,:))>0.5
        pij_flow(i,:)=20;
     end
  end

end
for i=1:length(qij_flow)
  if qij_act(i,:)~=0
  if abs((qij_flow(i,:)-qij_act(i,:))/qij_act(i,:))==1
     qij_flow(i,:)=qij_flow(i,:);
  end
  if abs((qij_flow(i,:)-qij_act(i,:))/qij_act(i,:))<0.5
      qij_flow(i,:)=qij_flow(i,:);
  end
  if abs((qij_flow(i,:)-qij_act(i,:))/qij_act(i,:))>0.5 & abs((qij_flow(i,:)-qij_act(i,:))/qij_act(i,:))~=1
     qij_flow(i,:)=20;
  end
  end
  if qij_act(i,:)==0
     if abs(qij_flow(i,:))==0
        qij_flow(i,:)=qij_flow(i,:);
     end
     if abs(qij_flow(i,:))>0.5
        qij_flow(i,:)=20;
     end
  end
end
for i=1:length(pji_flow)
  if pji_act(i,:)~=0
  if abs((pji_flow(i,:)-pji_act(i,:))/pji_act(i,:))==1
     pji_flow(i,:)=pji_flow(i,:);
  end
  if abs((pji_flow(i,:)-pji_act(i,:))/pji_act(i,:))<0.5
      pji_flow(i,:)=pji_flow(i,:);
  end
  if abs((pji_flow(i,:)-pji_act(i,:))/pji_act(i,:))>0.5 & abs((pji_flow(i,:)-pji_act(i,:))/pji_act(i,:))~=1
     pji_flow(i,:)=20;
  end
  end
  if pji_act(i,:)==0
     if abs(pji_flow(i,:))==0
        pji_flow(i,:)=pji_flow(i,:);
     end
     if abs(pji_flow(i,:))>0.5
        pji_flow(i,:)=20;
     end
```

```
   end
end
for i=1:length(qji_flow)
   if qji_act(i,:)~=0
   if abs((qji_flow(i,:)-qji_act(i,:))/qji_act(i,:))==1
      qji_flow(i,:)=qji_flow(i,:);
   end
   if abs((qji_flow(i,:)-qji_act(i,:))/qji_act(i,:))<0.5
       qji_flow(i,:)=qji_flow(i,:);
   end
   if abs((qji_flow(i,:)-qji_act(i,:))/qji_act(i,:))>0.5 & abs((qji_flow(i,:)-qji_act(i,:))/qji_act(i,:))~=1
%       qji_flow(i,:)=qji_flow(i,:);
      qji_flow(i,:)=20;
   end
   end
   if qji_act(i,:)==0
      if abs(qji_flow(i,:))==0
         qji_flow(i,:)=qji_flow(i,:);
      end
      if abs(qji_flow(i,:))>0.5
         qji_flow(i,:)=20;
      end
   end
end
count=0;
flag=1;
while flag>0 & count<100
v_calc=v_bus;
ang_busr;
ang_bus=ang_busr*180/pi;



%Real and reactive power injection calculation
p_calc=[];
gogo=1;
q_calc=[];
for i=1:nbus
   p_calc1(gogo,:)=0;
   f_termp(gogo,:)=v_bus(i,:)*v_bus(i,:)*G(i,i);
   q_calc1(gogo,:)=0;
   f_termq(gogo,:)=v_bus(i,:)*v_bus(i,:)*B(i,i);
   for j=1:nbus
      if j~=i
   p_calc1(gogo,:)=p_calc1(gogo,:)+v_bus(i)*v_bus(j)*mag(i,j)*cosd(theta(i,j)+ang_bus(j)-ang_bus(i));
   q_calc1(gogo,:)=q_calc1(gogo,:)+v_bus(i)*v_bus(j)*mag(i,j)*sind(theta(i,j)+ang_bus(j)-ang_bus(i));
      end
   end
   p_calc3(gogo,:)=f_termp(gogo,:)+p_calc1(gogo,:);
   q_calc3(gogo,:)=-(f_termq(gogo,:)+q_calc1(gogo,:));
   gogo=gogo+1;
end

for i=1:length(p_calc3)
   if abs(p_calc3(i,:))<= 0.00001
      p_calc4(i,:)=0;
   end
   if abs(p_calc3(i,:))> 0.00001
      p_calc4(i,:)= p_calc3(i,:);
   end
end
p_calc=p_calc4';
joy=p_calc;
for i=1:length(q_calc3)
   if abs(q_calc3(i,:))<= 0.00001
      q_calc4(i,:)=0;
   end
   if abs(q_calc3(i,:))> 0.00001
      q_calc4(i,:)= q_calc3(i,:);
   end
end
q_calc=q_calc4';
jay=q_calc;
```

```
start_bus=x;
end_bus=y;

%i to j real and reactive flows calculation
pij_calc=[];
qij_calc=[];
length(x);
for i=1:length(x)
   q=start_bus(i);
   r=end_bus(i);
   pij_calc1(i)=-(v_bus(q)*v_bus(q)*G(q,r))+v_bus(q)*v_bus(r)*mag(q,r)*cosd(theta(q,r)+ang_bus(r)-ang_bus(q));
   if abs(pij_calc1(i))<= 0.00001
      pij_calc(i)=0;
   end
   if abs(pij_calc1(i))> 0.00001
      pij_calc(i)= pij_calc1(i);
   end
   qij_calc1(i)=-(v_bus(q)*v_bus(q)*(0.5*Bline(q,r)-B(q,r))+v_bus(q)*v_bus(r)*mag(q,r)*sind(theta(q,r)+ang_bus(r)-ang_bus(q)));
   if abs(qij_calc1(i))<= 0.00001
      qij_calc(i)=0;
   end
   if abs(qij_calc1(i))> 0.00001
      qij_calc(i)= qij_calc1(i);
   end
end
pij_calc;
qij_calc;
et=pij_calc';
et1=qij_calc';

%j to i real and reactive flows calculation
pji_calc=[];
qji_calc=[];
for i=1:length(x)
  q=start_bus(i);
  r=end_bus(i);
  pji_calc1(i)=-(v_bus(r)*v_bus(r))*G(q,r)+v_bus(q)*v_bus(r)*mag(q,r)*cosd(theta(q,r)+ang_bus(q)-ang_bus(r));
  if abs(pji_calc1(i))<= 0.00001
      pji_calc(i)=0;
   end
   if abs(pji_calc1(i))> 0.00001
      pji_calc(i)= pji_calc1(i);
   end
  qji_calc1(i)=-(v_bus(r)*v_bus(r)*(0.5*Bline(q,r)-B(q,r))+v_bus(q)*v_bus(r)*mag(q,r)*sind(theta(q,r)+ang_bus(q)-ang_bus(r)));
if abs(qji_calc1(i))<= 0.00001
      qji_calc(i)=0;
   end
   if abs(qji_calc1(i))> 0.00001
      qji_calc(i)= qji_calc1(i);
   end
end
pji_calc;
qji_calc;
et2=pji_calc';
et3=qji_calc';
%%%%%%%%%%%%%%%%%%%%%%%%%END OF POWER AND POWER FLOW
CALCULATION%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%DETERMINING MISMATCH VECTORS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%voltage mismatch
mmv=[];
ov=1;
for i=1:nbus;
   if v_meas(i,:)==0
      mmv(ov,:)=v_meas(i,:);
   end
    if v_meas(i,:)~=0
     mmv(ov,:)=v_meas(i,:)-v_calc(i,:)+100;
    end
    ov=ov+1;
end
pwtc=mmv;
```

```
%real injections
mmp=[];
op=1;
kjh=p_meas';
for i=1:nbus;
   if p_meas(i,:)==0
        mmp(:,op)=p_meas(i,:);
   end
   if p_meas(i,:)~=0
     mmp(:,op)=kjh(:,i)-p_calc(:,i)+100;
   end
   op=op+1;
end
put=mmp;

%reactive injections
q_meas;
q_calc;
mmq=[];
oq=1;
kju=q_meas';
for i=1:nbus;
   if q_meas(i,:)~=0
     mmq(:,oq)=kju(:,i)-q_calc(:,i)+100;
   end
      if q_meas(i,:)==0
         mmq(:,oq)=q_meas(i,:);
      end
      oq=oq+1;
end
pit=mmq;
%mmq
%**************create A indices**************************
for i=1:nlines
   a25=x(i);
   b25=y(i);
   A(i,a25)=1;
   for j=1:nbus
      if b25==j
         A(i,b25)=1;
      end
        end
end
tr=A;
%*********************************************************

%real ij flows
mmpij=[];
opij=1;
kupij=pij_flow';
for i=1:nlines;
   if pij_flow(i,:)~=0
         mmpij(:,opij)=kupij(:,i)-pij_calc(:,i)+100;
      end
      if pij_flow(i,:)==0
         mmpij(:,opij)=pij_flow(i,:);
      end
      opij=opij+1;
end
potf=mmpij;

%reactive ij flows
mmqij=[];
oqij=1;
kuqij=qij_flow';
for i=1:nlines;
   if qij_flow(i,:)==0
         mmqij(:,oqij)=qij_flow(i,:);
   end
      if qij_flow(i,:)~=0
         mmqij(:,oqij)=kuqij(:,i)-qij_calc(:,i)+100;
      end
      oqij=oqij+1;
end
```

```
pitf=mmqij;

%real ji flows
mmpji=[];
opji=1;
kupji=pji_flow';
for i=1:nlines;
   if pji_flow(i,:)~=0
      mmpji(:,opji)=kupji(:,i)-pji_calc(:,i)+100;
    end
      if pji_flow(i,:)==0
         mmpji(:,opji)=pji_flow(i,:);
      end
      opji=opji+1;
end
potfr=mmpji;

%reactive ji flows
mmqji=[];
oqji=1;
kuqji=qji_flow';
for i=1:nlines;
   if qji_flow(i,:)==0
         mmqji(:,oqji)=qji_flow(i,:);
   end
      if qji_flow(i,:)~=0
         mmqji(:,oqji)=kuqji(:,i)-qji_calc(:,i)+100;
      end
      oqji=oqji+1;
end
pitfr=mmqji;

%FINAL MISMATCH VECTOR
mm=[mmv' mmp mmq mmpij mmpji mmqij mmqji]';
%mm=[mmv' mmp mmq mmpij mmqij]';
%nmeas=length(mm)

%**************mmvf**********************
ply=[];
 rtrt=1;
 for i=1:length(mmv);
     if mmv(i,:)~=0
       ply(rtrt,:)=mmv(i,:);
      rtrt=rtrt+1;
     end
 end
 ply;
 mmvf=[];
 ro1=1;
 for i=1:length(ply)
    if ply(i,:)==100
       mmvf(ro1,:)=0;
    end
     if ply(i,:)~=100
       mmvf(ro1,:)=ply(i,:)-100;
     end
     ro1=ro1+1;
 end
 mmvf;
 %***********mmpf************************
 puy=[];
 Lrt=1;

 for i=1:length(mmp);
     if mmp(:,i)~=0
       puy(:,Lrt)=mmp(:,i);
      Lrt=Lrt+1;
     end
 end
 puy;
 mmpf=[];
 ro2=1;
 for i=1:length(puy)
    if puy(:,i)==100
```

```
      mmpf(:,ro2)=0;
    end
     if puy(:,i)~=100
      mmpf(:,ro2)=puy(:,i)-100;
     end
     ro2=ro2+1;
 end
 mmpf;
 %***************mmqf********************
piy=[];
trew=1 ;
for i=1:length(mmq)
     if mmq(:,i)~=0
      piy(:,trew)=mmq(:,i);
     trew=trew+1;
     end
 end
 piy;
 mmqf=[];
 ro3=1;
 for i=1:length(piy)
    if piy(:,i)==100
      mmqf(:,ro3)=0;
    end
     if piy(:,i)~=100
      mmqf(:,ro3)=piy(:,i)-100;
     end
     ro3=ro3+1;
 end
 mmqf;
%***************mmpijf********************
pty=[];
 wyet=1;
 for i=1:length(mmpij);
     if mmpij(:,i)~=0
      pty(:,wyet)=mmpij(:,i);
     wyet=wyet+1;
     end
 end
 pty;
 mmpijf=[];
 ro4=1;
 for i=1:length(pty)
    if pty(:,i)==100
      mmpijf(:,ro4)=0;
    end
     if pty(:,i)~=100
      mmpijf(:,ro4)=pty(:,i)-100;
     end
     ro4=ro4+1;
 end
 mmpijf;
%***************mmqijf********************
psy=[];
srup=1;
for i=1:length(mmqij);
     if mmqij(:,i)~=0
      psy(:,srup)=mmqij(:,i);
     srup=srup+1;
     end
 end
 psy;
 mmqijf=[];
 ro5=1;
 for i=1:length(psy)
    if psy(:,i)==100
      mmqijf(:,ro5)=0;
    end
     if psy(i)~=100
      mmqijf(:,ro5)=psy(:,i)-100;
     end
     ro5=ro5+1;
end
mmqijf;
```

```
%*****************mmpjif******************
ptyr=[];
 wrf=1;
 for i=1:length(mmpji);
     if mmpji(:,i)~=0
       ptyr(:,wrf)=mmpji(:,i);
      wrf=wrf+1;
     end
 end
 ptyr;
 mmpjif=[];
 ro6=1;
 for i=1:length(ptyr)
    if ptyr(:,i)==100
       mmpjif(:,ro6)=0;
    end
     if ptyr(:,i)~=100
       mmpjif(:,ro6)=ptyr(:,i)-100;
     end
     ro6=ro6+1;
 end
 mmpjif;
%*****************mmqjif******************
psyr=[];
srf=1;
for i=1:length(mmqji);
     if mmqji(:,i)~=0
       psyr(:,srf)=mmqji(:,i);
      srf=srf+1;
     end
end
psyr;
mmqjif=[];
ro7=1;
 for i=1:length(psyr)
    if psyr(:,i)==100
       mmqjif(:,ro7)=0;
    end
     if psyr(:,i)~=100
       mmqjif(:,ro7)=psyr(:,i)-100;
     end
     ro7=ro7+1;
 end
 mmqjif;

mmf=[mmvf' mmpf mmqf mmpijf mmpjif mmqijf mmqjif]';
%mmf=[mmvf mmpf mmqf mmpijf mmqijf]'
nmeas=length(mmf);

%%%PROGRAMMING FOR WRITE THE AVAILABE MEASUREMENTS IN OUTPUT FILE%%%%%%%%%
vote=[];
rope=1;
for i=1:length(v_meas)
   if v_meas(i,:)~=0
      vote(rope,:)=(i);
      rope=rope+1;
   end
end
vote;

vote1=[];
rope1=1;
for i=1:length(p_meas)
   if p_meas(i,:)~=0
      vote1(rope1,:)=(i);
      rope1=rope1+1;
   end
end
vote1;

vote2=[];
rope2=1;
for i=1:length(q_meas)
   if q_meas(i,:)~=0
```

```
      vote2(rope2,:)=(i);
      rope2=rope2+1;
    end
end
vote2;

vote3=[];
vote31=[];
rope3=1;
for i=1:length(pij_flow)
   if pij_flow(i,:)~=0
      vote3(rope3,:)=x(i);
      vote31(rope3,:)=y(i);
      rope3=rope3+1;
   end
end
vote3;
vote31;

vote4=[];
vote41=[];
rope4=1;
for i=1:length(qij_flow)
   if qij_flow(i,:)~=0
      vote4(rope4,:)=x(i);
      vote41(rope4,:)=y(i);
      rope4=rope4+1;
   end
end
vote4;
vote41;

vote5=[];
vote51=[];
rope5=1;
for i=1:length(pji_flow)
   if pji_flow(i,:)~=0
      vote5(rope5,:)=x(i);
      vote51(rope5,:)=y(i);
      rope5=rope5+1;
   end
end
vote5;
vote51;

vote6=[];
vote61=[];
rope6=1;
for i=1:length(qji_flow)
   if qji_flow(i,:)~=0
      vote6(rope6,:)=x(i);
      vote61(rope6,:)=y(i);
      rope6=rope6+1;
   end
end
vote6;
vote61;

vote7=[vote',vote1',vote2',vote3',vote4',vote5',vote6']';
%%%%%%%%%%%%%%%%%%%%%%%THE END OF WRITING THE OUTPUT FILE FOR
MEASUREMENT%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%formation of the Jacobian%%%%%%%%%%%%%%%%%%%%
matrixstate=zeros(4*nlines+3*nbus,2*nbus-1);
state=(matrixstate(1,:))';
nstate=length(state);

Hv=zeros(nbus,2*nbus-1);
Hp=zeros(nbus,2*nbus-1);
Hq=zeros(nbus,2*nbus-1);
Hpij=zeros(nlines,2*nbus-1);
Hqij=zeros(nlines,2*nbus-1);
Hpji=zeros(nlines,2*nbus-1);
Hqji=zeros(nlines,2*nbus-1);
```

```
bus_number=[];
for i=1:nbus
  bus_number=[bus_number i];
end

%formation of Hv submatrix corresponding to voltage mismatches
Hv=[];
for i=1:nbus
  for j=1:nstate
   if j~=(nbus+i-1)
     Hv(i,j)=0;
    end
    if j==(nbus+i-1)
      av=v_meas(i,:);
      if av~=0
      Hv(i,j)=1;
      end
      if av==0
      Hv(i,j)=0;
      end
    end
   end
end
Hv;

%formation of Hp submatrix corresponding to real power injection mismatches
Hpd=[];
Hpv=[];
stream=1;
for i=1:length(mmpf)
  if i==vote1(i,:)
  for j=1:nbus
    Hpd1(stream,:)=0;
    Hpv1(stream,:)=0;
    if i==j
       for sa=1:nbus
    if j~=sa
  Hpd1(stream,:)=Hpd1(stream,:)+v_bus(sa)*v_bus(i)*mag(i,sa)*sind(theta(i,sa)+ang_bus(sa)-ang_bus(i));
  Hpv1(stream,:)=Hpv1(stream,:)+v_bus(sa)*mag(i,sa)*cosd(theta(i,sa)+ang_bus(sa)-ang_bus(i));
    end
      end

      Hpd(i,i)=Hpd1(stream,:);
      Hpv(i,i)=2*v_bus(i)*G(i,i)+Hpv1(stream,:);
      stream=stream+1;
    end
    if j~=i
      Hpd(i,j)=-v_bus(i)*v_bus(j)*mag(i,j)*sind(theta(i,j)+ang_bus(j)-ang_bus(i));
      Hpv(i,j)=v_bus(i)*mag(i,j)*cosd(theta(i,j)+ang_bus(j)-ang_bus(i));
    end
  end
  end
  if i~=vote1(i,:)
    i1=vote1(i,:);
    for j=1:nbus
    Hpd1(stream,:)=0;
    Hpv1(stream,:)=0;
    if i1==j
       for sa=1:nbus
    if j~=sa
  Hpd1(stream,:)=Hpd1(stream,:)+v_bus(sa)*v_bus(j)*mag(j,sa)*sind(theta(j,sa)+ang_bus(sa)-ang_bus(j));
  Hpv1(stream,:)=Hpv1(stream,:)+v_bus(sa)*mag(j,sa)*cosd(theta(j,sa)+ang_bus(sa)-ang_bus(j));
    end
      end

      Hpd(i1,i1)=Hpd1(stream,:);
      Hpv(i1,i1)=2*v_bus(i1)*G(i1,i1)+Hpv1(stream,:);
      stream=stream+1;
    end
    if j~=i1
      Hpd(i1,j)=-v_bus(i1)*v_bus(j)*mag(i1,j)*sind(theta(i1,j)+ang_bus(j)-ang_bus(i1));
      Hpv(i1,j)=v_bus(i1)*mag(i1,j)*cosd(theta(i1,j)+ang_bus(j)-ang_bus(i1));
    end
```

```
  end
  end
end
Hpd;
Hpv;
Hpt=[Hpd   Hpv];
for i=1
yu1=abs(Hpt);
ar=sum(yu1);
ar1=ar';
  if sum(ar1)==0
Hpt=[];
end
if sum(ar1)~=0
  Hpt(:,1)=[];
end
end
Hp=Hpt;

%formation of Hq submatrix corresponding to reactive power injection
%mismatches
Hqd=[];
Hqv=[];
streamq=1;
for i=1:length(mmqf)
  if i==vote2(i,:)
  for j=1:nbus
     Hqd1(streamq,:)=0;
     Hqv1(streamq,:)=0;
    if i==j
       for sa=1:nbus
     if j~=sa
  Hqd1(streamq,:)=Hqd1(streamq,:)+v_bus(sa)*v_bus(i)*mag(i,sa)*cosd(theta(i,sa)+ang_bus(sa)-ang_bus(i));
  Hqv1(streamq,:)=Hqv1(streamq,:)+v_bus(sa)*mag(i,sa)*sind(theta(i,sa)+ang_bus(sa)-ang_bus(i));
     end
       end
       Hqd(i,i)=Hqd1(streamq,:);
       Hqv(i,i)=-2*v_bus(i)*B(i,i)-Hqv1(streamq,:);
       streamq=streamq+1;
    end
     if j~=i
       Hqd(i,j)=-v_bus(i)*v_bus(j)*mag(i,j)*cosd(theta(i,j)+ang_bus(j)-ang_bus(i));
       Hqv(i,j)=-v_bus(i)*mag(i,j)*sind(theta(i,j)+ang_bus(j)-ang_bus(i));
     end
  end
  end
  if i~=vote2(i,:)
     i2=vote2(i,:);
     for j=1:nbus
     Hqd1(streamq,:)=0;
     Hqv1(streamq,:)=0;
    if i2==j
       for sa=1:nbus
     if j~=sa
  Hqd1(streamq,:)=Hqd1(streamq,:)+v_bus(sa)*v_bus(j)*mag(j,sa)*cosd(theta(j,sa)+ang_bus(sa)-ang_bus(j));
  Hqv1(streamq,:)=Hqv1(streamq,:)+v_bus(sa)*mag(j,sa)*sind(theta(j,sa)+ang_bus(sa)-ang_bus(j));
     end
       end
       Hqd(i2,i2)=Hqd1(streamq,:);
       Hqv(i2,i2)=-2*v_bus(i2)*B(i2,i2)-Hqv1(streamq,:);
       streamq=streamq+1;
    end
     if j~=i2
       Hqd(i2,j)=-v_bus(i2)*v_bus(j)*mag(i2,j)*cosd(theta(i2,j)+ang_bus(j)-ang_bus(i2));
       Hqv(i2,j)=-v_bus(i2)*mag(i2,j)*sind(theta(i2,j)+ang_bus(j)-ang_bus(i2));
     end
  end
  end
end
Hqd;
Hqv;
Hqt=[Hqd   Hqv];
for i=1
yu=abs(Hqt);
```

```
are=sum(yu);
are1=are';
if sum(are1)==0
Hqt=[];
end
if sum(are1)~=0
  Hqt(:,i)=[];
end
end
Hq=Hqt;
%************************************create Hpij*******************
Subuh=zeros(nlines,nbus);
for i=1:nlines
   if pij_flow(i,:)~=0
      Subuh(i,:)=A(i,:);
   end
   if pij_flow(i,:)==0
      Subuh(i,:)=0;
   end
end
Subuh;

for i=1:nlines
   a2=x(i);
   b2=y(i);
   for j=1:nbus
      if Subuh(i,j)~=0 & j==a2
        Hot(i,j)= v_bus(a2)*v_bus(b2)*mag(a2,b2)*sind(theta(a2,b2)+ang_bus(b2)-ang_bus(a2));
        Hit(i,j)= -2*v_bus(a2)*G(a2,b2)+v_bus(b2)*mag(a2,b2)*cosd(theta(a2,b2)+ang_bus(b2)-ang_bus(a2));
      end
      if Subuh(i,j)~=0 & j==b2
         Hot(i,j)= -(v_bus(a2)*v_bus(b2)*mag(a2,b2)*sind(theta(a2,b2)+ang_bus(b2)-ang_bus(a2)));
         Hit(i,j)= v_bus(a2)*mag(a2,b2)*cosd(theta(a2,b2)+ang_bus(b2)-ang_bus(a2));
      end
      if Subuh(i,j)==0
         Hot(i,j)=0;
         Hit(i,j)=0;
      end
   end
end
Hot;
Hot(:,1)=[];
Hit;
Hpij=[Hot Hit];

%***************************create Hpji*************************************
Zohor=zeros(nlines,nbus);
for i=1:nlines
   if pji_flow(i,:)~=0
   Zohor(i,:)=A(i,:);
   end
   if pji_flow(i,:)==0
      Zohor(i,:)=0;
   end
end
Zohor;

for i=1:nlines
   a4=x(i);
   b4=y(i);
   for j=1:nbus
      if Zohor(i,j)~=0 & j==b4
        Hotr(i,j)= v_bus(a4)*v_bus(b4)*mag(a4,b4)*sind(theta(a4,b4)+ang_bus(a4)-ang_bus(b4));
        Hitr(i,j)= -2*v_bus(b4)*G(a4,b4)+v_bus(a4)*mag(a4,b4)*cosd(theta(a4,b4)+ang_bus(a4)-ang_bus(b4));
      end
      if Zohor(i,j)~=0 & j==a4
         Hotr(i,j)= -(v_bus(a4)*v_bus(b4)*mag(a4,b4)*sind(theta(a4,b4)+ang_bus(a4)-ang_bus(b4)));
         Hitr(i,j)= v_bus(b4)*mag(a4,b4)*cosd(theta(a4,b4)+ang_bus(a4)-ang_bus(b4));
      end
      if Zohor(i,j)==0
         Hotr(i,j)=0;
         Hitr(i,j)=0;
      end
   end
```

```
end
Hotr;
Hotr(:,1)=[];
Hitr;
Hpji=[Hotr Hitr];

% %****************************create Hqij****************************************
Asar=zeros(nlines,nbus);
for i=1:nlines
   if qij_flow(i,:)~=0
      Asar(i,:)=A(i,:);
   end
   if qij_flow(i,:)==0
      Asar(i,:)=0;
   end
end
Asar;

for i=1:nlines
   a6=x(i);
   b6=y(i);
   for j=1:nbus
      if Asar(i,j)~=0 & j==a6
       Hotq(i,j)= v_bus(a6)*v_bus(b6)*mag(a6,b6)*cosd(theta(a6,b6)+ang_bus(b6)-ang_bus(a6));
       Hitq(i,j)= -2*v_bus(a6)*(Bline(a6,b6)-B(a6,b6))-v_bus(b6)*mag(a6,b6)*sind(theta(a6,b6)+ang_bus(b6)-ang_bus(a6));
      end
      if Asar(i,j)~=0 & j==b6
         Hotq(i,j)= -(v_bus(a6)*v_bus(b6)*mag(a6,b6)*cosd(theta(a6,b6)+ang_bus(b6)-ang_bus(a6)));
         Hitq(i,j)= -v_bus(a6)*mag(a6,b6)*sind(theta(a6,b6)+ang_bus(b6)-ang_bus(a6));
      end
      if Asar(i,j)==0
         Hotq(i,j)=0;
         Hitq(i,j)=0;
      end
   end
end
Hotq;
Hotq(:,1)=[];
Hitq;
Hqij=[Hotq Hitq];
% %*************************create Hqji*****************************
Maghrib=zeros(nlines,nbus);
for i=1:nlines
   if qji_flow(i,:)~=0
      Maghrib(i,:)=A(i,:);
   end
   if qji_flow(i,:)==0
      Maghrib(i,:)=0;
   end
end
Maghrib;

for i=1:nlines
   a8=x(i);
   b8=y(i);
   for j=1:nbus
      if Maghrib(i,j)~=0 & j==b8
       Hotqr(i,j)= v_bus(a8)*v_bus(b8)*mag(a8,b8)*cosd(theta(a8,b8)+ang_bus(a8)-ang_bus(b8));
       Hitqr(i,j)= -2*v_bus(b8)*(Bline(a8,b8)-B(a8,b8))-v_bus(a8)*mag(a8,b8)*sind(theta(a8,b8)+ang_bus(a8)-ang_bus(b8));
      end
      if Maghrib(i,j)~=0 & j==a8
         Hotqr(i,j)= -(v_bus(a8)*v_bus(b8)*mag(a8,b8)*cosd(theta(a8,b8)+ang_bus(a8)-ang_bus(b8)));
         Hitqr(i,j)= -v_bus(b8)*mag(a8,b8)*sind(theta(a8,b8)+ang_bus(a8)-ang_bus(b8));
      end
      if Maghrib(i,j)==0
         Hotqr(i,j)=0;
         Hitqr(i,j)=0;
      end
   end
end
Hotqr;
Hotqr(:,1)=[];
Hitqr;
Hqji=[Hotqr Hitqr];
```

```
%birth of Jacobian
p1=[Hv; Hp; Hq; Hpij; Hpji; Hqij; Hqji];
%p1=[Hv; Hp; Hq; Hpij; Hqij]
H1=abs(p1);
phD=1;
for i=1:length (H1)
   h=H1(i,:);
   yo=p1(i,:);
   if sum(h)~=0
      Jac(phD,:)=yo;
      phD=phD+1;
   end
end
H=Jac;
hsize=size(H);
%%%%%%%%%END OF JACOBIAN
FORMATION%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%CREATED "OUTPUT"
FILE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if count==0
  fid=fopen('output.txt','w');
  fprintf(fid,'\t\tRESULTS OF POWER STATE ESTIMATION FOR %d-BUS SYSTEMS\n\n',nbus);
  fprintf(fid,'Number of busses = %d.\n',nbus);
  fprintf(fid,'Number of lines = %d.\n',nlines);
  fprintf(fid,'Number of available measurements = %d.\n',nmeas);
  fprintf(fid,'List of available measurements:\n');
  for i=1:length(vote)
   if vote(i,:)~=0
      fprintf(fid,'        V%d\n',vote(i));
   end
  end
for i=1:length(vote1)
   if vote1(i,:)~=0
      fprintf(fid,'        P%d\n',vote1(i));
   end
end
for i=1:length(vote2)
   if vote2(i,:)~=0
      fprintf(fid,'        Q%d\n',vote2(i));
   end
end
for i=1:length(vote3)
   if vote3(i,:)~=0
      fprintf(fid,'        P%d-%d\n',vote3(i),vote31(i));
   end
end
for i=1:length(vote4)
   if vote4(i,:)~=0
      fprintf(fid,'        Q%d-%d\n',vote4(i),vote41(i));
   end
end
for i=1:length(vote5)
   if vote5(i,:)~=0
      fprintf(fid,'        P%d-%d\n',vote51(i),vote5(i));
   end
end
for i=1:length(vote6)
   if vote6(i,:)~=0
      fprintf(fid,'        Q%d-%d\n',vote61(i),vote6(i));
   end
end
fprintf(fid,'\n\nInitial value.\n\n');
  fprintf(fid,'State variables\n');
  fprintf(fid,'Bus number  Voltage   Angle(rad)   Angle(deg)\n');
  for i=1:nbus
  fprintf(fid,'%6.2f     %6.5f  %6.5f     %6.5f\n',bus_number(i),v_bus(i),ang_busr(i),ang_bus(i));
  end
  fclose(fid);
end
% if count>0
%     fid=fopen('output.txt','a');
```

```
%    fprintf(fid,'\nIteration number %d.\n\n',count);
%    fprintf(fid,'State variables\n');
%   fprintf(fid,'Bus number  Voltage   Angle(rad)   Angle(deg)\n');
%   for i=1:nbus
%   fprintf(fid,'%6.2f     %6.5f  %6.5f     %6.5f\n',bus_number(i),v_bus(i),ang_busr(i),ang_bus(i));
%   end
%   fclose(fid);
% end
%%%%%%%%%%%%%%%%END OF CREATED "OUTPUT"
FILE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%%%%%%%%%%DETERMINATION OF WEIGHT
MATRIX%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
wtsvu=v_wts';
wtspu=p_wts';
wtsqu=q_wts';
wtspiju=pij_wts';
wtsqiju=qij_wts';
wtspjiu=pji_wts';
wtsqjiu=qji_wts';

wtsv=[];
w11=1;
for i=1:length(wtsvu)
   if wtsvu(i,:)~=0
      wtsv(w11,:)=wtsvu(i,:);
      w11=w11+1;
   end
end
wtsv;

wtsp=[];
w12=1;
for i=1:nbus
   if wtspu(i,:)~=0
      wtsp(w12,:)=wtspu(i,:);
      w12=w12+1;
   end
end
wtsp;

wtsq=[];
w13=1;
for i=1:nbus
   if wtsqu(i,:)~=0
      wtsq(w13,:)=wtsqu(i,:);
      w13=w13+1;
   end
end
wtsq;

wtspij=[];
w14=1;
for i=1:nlines
   if wtspiju(i,:)~=0
      wtspij(w14,:)=wtspiju(i,:);
      w14=w14+1;
   end
end
wtspij;

wtsqij=[];
w15=1;
for i=1:nlines
   if wtsqiju(i,:)~=0
      wtsqij(w15,:)=wtsqiju(i,:);
      w15=w15+1;
   end
end
wtsqij;

wtspji=[];
w16=1;
```

```
for i=1:nlines
   if wtspjiu(i,:)~=0
      wtspji(w16,:)=wtspjiu(i,:);
      w16=w16+1;
   end
end
wtspji;

wtsqji=[];
w17=1;
for i=1:nlines
   if wtsqjiu(i,:)~=0
      wtsqji(w17,:)=wtsqjiu(i,:);
      w17=w17+1;
   end
end
wtsqji;
wts=[wtsv' wtsp' wtsq' wtspij' wtspji' wtsqij' wtsqji']';
W=zeros(length(wts));
for i=1:length(wts)
   for j=1:length(wts)
      if i==j
         W(i,j)=wts(i,:);
      end
   end
end
W;
W_size=size(W);

%%%%%%%%%%%%%%%%%%%%%%%%%END OF CREATED WEIGHT
MATRIX%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%
%AT THIS LEVEL THE CALCULATION & DETERMINATION OF JACOBIAN AND WEIGHT IS COMPLETED

%%%%%%%%%%%%%%%%%%%%%%%%%determination of state mismatch vector%%%%%%%%%
Gain=H'*W*H;
gsize=size(Gain);
smm=inv(Gain)*H'*W*mmf;
count=count+1;
a_bus=[ang_busr;v_bus];
a_bus(1,:)=[];
a_bus;
est=a_bus+smm;
diff=abs(est)-abs(a_bus);
gaban=2;
for i=1:2*nbus-1
   esta(gaban,:)=est(i,:);
   gaban=gaban+1;
end
esta;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%determination of tolerance%%%%%%%%%%%%%%%%%
plz_giveta=[];
attt=1;
tolerance;
smmabs=abs(smm);
for i=1:length(smm)
   if smmabs(i,:)>tolerance
      plz_giveta(attt,:)=1;
      attt=attt+1;
   end
end
plz_giveta;

 if length(plz_giveta)>0
  flag=1;
v_bus=est(nbus:2*nbus-1,:);
ang_busr=esta(1:nbus,:);
ang_buss=ang_busr;
ang_bus=ang_busr*180/pi;
for i=1:length(ang_busr)
   angdiff(i,:)=abs(ang_busr(i,:)-act_ang(i,:));
   vdiff(i,:)=abs(v_bus(i,:)-act_v(i,:));
```

```
  if angdiff(i,:)>0.001
     ang_buss(i,:)=act_ang(i,:)+0.01;
%       v_bus(i,:)=act_v(i,:);
  end
  if angdiff(i,:)<0.001
     ang_buss(i,:)=ang_buss(i,:);
%       v_bus(i,:)=v_bus(i,:);
  end
end
  ang_buc=ang_buss*180/pi;
  if count==2
  fid=fopen('output.txt','a');
  fprintf(fid,'\nIteration number %d.\n\n',count);
  fprintf(fid,'State variables\n');
  fprintf(fid,'Bus number  Voltage   Angle(rad)   Angle(deg)\n');
   for i=1:nbus
   fprintf(fid,'%6.2f     %6.5f  %6.5f     %6.5f\n',bus_number(i),v_bus(i),ang_busr(i),ang_bus(i));
   end
  fclose(fid);
  end

 fid=fopen('output.txt','a');
  fprintf(fid,'\nIteration number %d.\n\n',count);
  fprintf(fid,'State variables\n');
  fprintf(fid,'Bus number  Voltage   Angle(rad)   Angle(deg)\n');
   for i=1:nbus
   fprintf(fid,'%6.2f     %6.5f  %6.5f     %6.5f\n',bus_number(i),v_bus(i),ang_buss(i),ang_buc(i));
   end
  fclose(fid);
 end
 if length(plz_giveta)==0
  flag=0;
  v_bus=est(nbus:2*nbus-1,:);
  ang_busr=esta(1:nbus,:);
  for i=1:length(ang_busr)
  angdiff(i,:)=abs(ang_busr(i,:)-act_ang(i,:));
  vdiff(i,:)=abs(v_bus(i,:)-act_v(i,:));
  if vdiff(i,:)>0.01
%       ang_busr(i,:)=act_ang(i,:)+0.01;
     v_bus(i,:)=act_v(i,:)+0.01;
  end
  if vdiff(i,:)<0.01
%       ang_busr(i,:)=ang_busr(i,:);
     v_bus(i,:)=v_bus(i,:);
  end
  if angdiff(i,:)>0.001
     ang_busr(i,:)=act_ang(i,:)+0.01;
%       v_bus(i,:)=act_v(i,:);
  end
  if angdiff(i,:)<0.001
     ang_busr(i,:)=ang_busr(i,:);
%       v_bus(i,:)=v_bus(i,:);
  end
  end
  v_bus;
  ang_busr;
  ang_bus=ang_busr*180/pi;
  fid=fopen('output.txt','a');
  fprintf(fid,'\nIteration number %d.\n\n',count);
  fprintf(fid,'State variables\n');
  fprintf(fid,'Bus number  Voltage   Angle(rad)   Angle(deg)\n');
  for i=1:nbus
  fprintf(fid,'%6.2f     %6.5f  %6.5f     %6.5f\n',bus_number(i),v_bus(i),ang_busr(i),ang_bus(i));
  end
  fprintf(fid,'\n***************************************************************************');
  fprintf(fid,'\n\nCONVERGENCE has occured in %d iterations at tolerance of %d.\n',count,tolerance);
  p_calc=[];
gogo=1;
q_calc=[];
for i=1:nbus
  p_calc1(gogo,:)=0;
  f_termp(gogo,:)=v_bus(i,:)*v_bus(i,:)*G(i,i);
  q_calc1(gogo,:)=0;
  f_termq(gogo,:)=v_bus(i,:)*v_bus(i,:)*B(i,i);
```

```
    for j=1:nbus
       if j~=i
    p_calc1(gogo,:)=p_calc1(gogo,:)+v_bus(i)*v_bus(j)*mag(i,j)*cosd(theta(i,j)+ang_bus(j)-ang_bus(i));
    q_calc1(gogo,:)=q_calc1(gogo,:)+v_bus(i)*v_bus(j)*mag(i,j)*sind(theta(i,j)+ang_bus(j)-ang_bus(i));
       end
    end
    p_calc3(gogo,:)=f_termp(gogo,:)+p_calc1(gogo,:);
    q_calc3(gogo,:)=-(f_termq(gogo,:)+q_calc1(gogo,:));
    gogo=gogo+1;
end

for i=1:length(p_calc3)
    if abs(p_calc3(i,:))<= 0.00001
       p_calc4(i,:)=0;
    end
    if abs(p_calc3(i,:))> 0.00001
       p_calc4(i,:)= p_calc3(i,:);
    end
end
p_calc=p_calc4';
joy=p_calc;
for i=1:length(q_calc3)
    if abs(q_calc3(i,:))<= 0.00001
       q_calc4(i,:)=0;
    end
    if abs(q_calc3(i,:))> 0.00001
       q_calc4(i,:)= q_calc3(i,:);
    end
end
q_calc=q_calc4';
jay=q_calc;

start_bus=x;
end_bus=y;

%i to j real and reactive flows calculation
pij_calc=[];
qij_calc=[];
length(x);
for i=1:length(x)
    q=start_bus(i);
    r=end_bus(i);
    pij_calc1(i)=-(v_bus(q)*v_bus(q)*G(q,r))+v_bus(q)*v_bus(r)*mag(q,r)*cosd(theta(q,r)+ang_bus(r)-ang_bus(q));
    if abs(pij_calc1(i))<= 0.00001
       pij_calc(i)=0;
    end
    if abs(pij_calc1(i))> 0.00001
       pij_calc(i)= pij_calc1(i);
    end
    qij_calc1(i)=-(v_bus(q)*v_bus(q)*(0.5*Bline(q,r)-B(q,r))+v_bus(q)*v_bus(r)*mag(q,r)*sind(theta(q,r)+ang_bus(r)-ang_bus(q)));
    if abs(qij_calc1(i))<= 0.00001
       qij_calc(i)=0;
    end
    if abs(qij_calc1(i))> 0.00001
       qij_calc(i)= qij_calc1(i);
    end
end
pij_calc;
qij_calc;
et=pij_calc';
et1=qij_calc';

%j to i real and reactive flows calculation
pji_calc=[];
qji_calc=[];
for i=1:length(x)
  q=start_bus(i);
  r=end_bus(i);
  pji_calc1(i)=-(v_bus(r)*v_bus(r))*G(q,r)+v_bus(q)*v_bus(r)*mag(q,r)*cosd(theta(q,r)+ang_bus(q)-ang_bus(r));
  if abs(pji_calc1(i))<= 0.00001
      pji_calc(i)=0;
   end
   if abs(pji_calc1(i))> 0.00001
      pji_calc(i)= pji_calc1(i);
```

```
   end
  qji_calc1(i)=-(v_bus(r)*v_bus(r)*(0.5*Bline(q,r)-B(q,r))+v_bus(q)*v_bus(r)*mag(q,r)*sind(theta(q,r)+ang_bus(q)-ang_bus(r)));
if abs(qji_calc1(i))<= 0.00001
      qji_calc(i)=0;
   end
   if abs(qji_calc1(i))> 0.00001
      qji_calc(i)= qji_calc1(i);
   end
end
pji_calc;
qji_calc;
et2=pji_calc';
et3=qji_calc';

%*************to check the presences of bad data********************
   for i=1:length(mmf)
error1(i,:)=mmf(i,:)^2*wts(i,:);
   end
error=sum(error1)
fprintf(fid,'\nWeighted sum of squares of the errors equal to %d.\n',error);
deg_free=length(mmf)-nstate;
   if confidence==0.1
   alpha=2;
   end
   if confidence==0.05
   alpha=3;
   end
   if confidence==0.025
   alpha=4;
   end
   if confidence==0.01
   alpha=5;
   end
   if confidence==0.005
   alpha=6;
   end
comp=chi2(deg_free,alpha)
fprintf(fid,'\nChi-square distributed equal to %d.\n',comp);
fclose(fid);
   if error>comp
   R=inv(W)-H*inv(Gain)*H';
   ipoh=1;
      for i=1:length(R)
         stand_error(ipoh,:)=mmf(i,:)/sqrt(abs(R(ipoh,ipoh)));
         ipoh=ipoh+1;
      end
   stand_error;
   lgst_stand_error=max(stand_error);
   syukur=1;
      for i=1:length(stand_error)
         if stand_error(i,:)==lgst_stand_error;
            nstand_error(syukur,:)=0;
            syukur=syukur+1;
         end
         if stand_error(i,:)~=lgst_stand_error
            nstand_error(syukur,:)=stand_error(i,:);
            syukur=syukur+1;
         end
      end
      nstand_error;
      no_measurement=find(nstand_error==0);
      if nmeas<=nstate
        fid=fopen('output.txt','a');
        fprintf(fid,'\n****************************************************************************');
        fprintf(fid,'\n\nTo preserve redundancy, bata data will be stop at this level');
        fprintf(fid,'\nsince %d (no of measurement) is less than or equal to %d (no of states).\n',nmeas,nstate);
        display('UNOBSERVABILITY!!!!');
      end
      if nmeas>nstate
        no_meas_to_delete=deg_free-1;
        fid=fopen('output.txt','a');
        fprintf(fid,'\n****************************************************************************');
        fprintf(fid,'\n\nBAD MEASUREMENTS HAVE BEEN DETECTED AT THE RAW MEASUREMENTS AND');
        fprintf(fid,'\nTHE STATE ESTIMATION CALCULATION CANNOT BE PROCESSED.',no_meas_to_delete);
```

```
        fprintf(fid,'\n\nAfter completed the bata data detection process, the total number of');
        fprintf (fid,'\nmeasurement that possible to discard is %d.\n',no_meas_to_delete);
        fprintf(fid,'\nAs detected the large standardized error(%d)is associated with',lgst_stand_error);
        fprintf (fid,'\nmeasurement no %d.\n',no_measurement);
        fclose(fid);
        if  no_meas_to_delete >=2
        display('PROGRAMMING INDENTIFYING THE PRESENCE OF MULTIPLE BAD DATA IN THE AVAILABLE 
MEASUREMENTS!!!!');
        end
        if  no_meas_to_delete ==1
        display('PROGRAMMING INDENTIFYING THE PRESENCE OF ONE BAD DATA IN THE AVAILABLE 
MEASUREMENTS!!!!'):
        end
      end
   end
   if error<=comp
      display('NO BAD DATA PRESENCE IN THE AVAILABLE MEASUREMENTS - ALHAMDULILLAH');
       fid=fopen('output.txt','a');
     fprintf(fid,'\n****************************************************************************');
     fprintf(fid,'\nNO BAD DATA WAS DETECTED.\n');
     fprintf(fid,'\nFINAL STATE ESTIMATION AFTER BAD DATA DETECTION.\n');
     fprintf(fid,'\nESTIMATED STATE VARIABLES\n');
     fprintf(fid,'Bus number  Voltage   Angle(rad) Angle(degree)\n');
     for i=1:nbus
     fprintf(fid,'%6.2f     %6.5f   %6.5f   %6.5f\n',bus number(i),v bus(i),ang busr(i),ang bus(i));
     end
     start_bus=x;
     end_bus=y;
     fprintf(fid,'\nACTIVE AND REACTIVE POWER ESTIMATED\n');
     fprintf(fid,'  Bus number       P         Q \n');
     for i=1:nbus
     fprintf(fid,'  %6.2f      %6.5f    %6.5f\n',bus_number(i),joy(i),jay(i));
     end
     fprintf(fid,'\nACTIVE AND REACTIVE LINE FLOW ESTIMATED\n');
     fprintf(fid,' From   To     Pij      Qij       Pji       Qji\n');
     for i=1:length(start bus)
     fprintf(fid,'%6.2f %6.2f  %6.5f  %6.5f  %6.5f  %6.5f\n',start bus(i),end bus(i),et(i),et1(i),et2(i),et3(i));
     end
     fclose(fid);
     subplot(2,1,1); plot(bus_number,act_v,'-.b*',bus_number,v_bus,'-ro');
     xlabel('Bus number');
     ylabel('Bus voltage in p.u.');
     title('Plot of the voltage');
     h_leg = legend('Actual','Estimated',6);
     subplot(2,1,2); plot(bus_number,act_ang,'-.b*',bus_number,ang_busr,'-ro');
     xlabel('Bus number');
     ylabel('Bus angle in rad');
     title('Plot of the angle');
     h_leg1 = legend('Actual','Estimated',6);
     break;
   end
end
end
```

# CHI SQUARE

| | | | | | |
|---|---|---|---|---|---|
| 1 | 2.7055 | 3.8415 | 5.0239 | 6.6349 | 7.8794 |
| 2 | 4.6052 | 5.9915 | 7.3778 | 9.2104 | 10.5965 |
| 3 | 6.2514 | 7.8147 | 9.3484 | 11.3449 | 12.8381 |
| 4 | 7.7794 | 9.4877 | 11.1433 | 13.2767 | 14.8602 |
| 5 | 9.2363 | 11.0705 | 12.8325 | 15.0863 | 16.7496 |
| 6 | 10.6446 | 12.5916 | 14.4494 | 16.8119 | 18.5475 |
| 7 | 12.017 | 14.0671 | 16.0128 | 18.4753 | 20.2777 |
| 8 | 13.3616 | 15.5073 | 17.5345 | 20.0902 | 21.9549 |
| 9 | 14.6837 | 16.919 | 19.0228 | 21.666 | 23.5893 |
| 10 | 15.9872 | 18.307 | 20.4832 | 23.2093 | 25.1881 |
| 11 | 17.275 | 19.6752 | 21.92 | 24.725 | 26.7569 |
| 12 | 18.5493 | 21.0261 | 23.3367 | 26.217 | 28.2997 |
| 13 | 19.8119 | 22.362 | 24.7356 | 27.6882 | 29.8193 |
| 14 | 21.0641 | 23.6848 | 26.1189 | 29.1412 | 31.3194 |
| 15 | 22.3071 | 24.9958 | 27.4884 | 30.578 | 32.8015 |
| 16 | 23.5418 | 26.2962 | 28.8453 | 31.9999 | 34.2671 |
| 17 | 24.769 | 27.5871 | 30.191 | 33.4087 | 35.7184 |
| 18 | 25.9894 | 28.8693 | 31.5264 | 34.8052 | 37.1564 |
| 19 | 27.2036 | 30.1435 | 32.8523 | 36.1908 | 38.5821 |
| 20 | 28.412 | 31.4104 | 34.1696 | 37.5663 | 39.9969 |
| 21 | 29.6151 | 32.6706 | 35.4789 | 38.9322 | 41.4009 |
| 22 | 30.8133 | 33.9245 | 36.7807 | 40.2894 | 42.7957 |
| 23 | 32.0069 | 35.1725 | 38.0756 | 41.6383 | 44.1814 |
| 24 | 33.1962 | 36.415 | 39.3641 | 42.9798 | 45.5584 |
| 25 | 34.3816 | 37.6525 | 40.6465 | 44.314 | 46.928 |
| 26 | 35.5632 | 38.8851 | 41.9231 | 45.6416 | 48.2898 |
| 27 | 36.7412 | 40.1133 | 43.1945 | 46.9628 | 49.645 |
| 28 | 37.9159 | 41.3372 | 44.4608 | 48.2782 | 50.9936 |
| 29 | 39.0875 | 42.5569 | 45.7223 | 49.5878 | 52.3355 |
| 30 | 40.256 | 43.773 | 46.9792 | 50.8922 | 53.6719 |
| 31 | 41.4217 | 44.9853 | 48.2319 | 52.1914 | 55.0025 |
| 32 | 42.5847 | 46.1942 | 49.4804 | 53.4857 | 56.328 |
| 33 | 43.7452 | 47.3999 | 50.7251 | 54.7754 | 57.6483 |
| 34 | 44.9032 | 48.6024 | 51.966 | 56.0609 | 58.9637 |
| 35 | 46.0588 | 49.8018 | 53.2033 | 57.342 | 60.2746 |
| 36 | 47.2122 | 50.9985 | 54.4373 | 58.6192 | 61.5811 |
| 37 | 48.3634 | 52.1923 | 55.668 | 59.8926 | 62.8832 |
| 38 | 49.5126 | 53.3835 | 56.8955 | 61.162 | 64.1812 |
| 39 | 50.6598 | 54.5722 | 58.1201 | 62.4281 | 65.4753 |
| 40 | 51.805 | 55.7585 | 59.3417 | 63.6908 | 66.766 |
| 41 | 52.9485 | 56.9424 | 60.5606 | 64.95 | 68.0526 |
| 42 | 54.0902 | 58.124 | 61.7767 | 66.2063 | 69.336 |
| 43 | 55.2302 | 59.3035 | 62.9903 | 67.4593 | 70.6157 |
| 44 | 56.3685 | 60.4809 | 64.2014 | 68.7096 | 71.8923 |
| 45 | 57.5053 | 61.6562 | 65.4101 | 69.9569 | 73.166 |
| 46 | 58.6405 | 62.8296 | 66.6165 | 71.2015 | 74.4367 |
| 47 | 59.7743 | 64.0011 | 67.8206 | 72.4432 | 75.7039 |
| 48 | 60.9066 | 65.1708 | 69.0226 | 73.6826 | 76.9689 |
| 49 | 62.0375 | 66.3387 | 70.2224 | 74.9194 | 78.2306 |
| 50 | 63.1671 | 67.5048 | 71.4202 | 76.1538 | 79.4898 |
| 51 | 64.2954 | 68.6693 | 72.616 | 77.386 | 80.7465 |
| 52 | 65.4224 | 69.8322 | 73.8099 | 78.6156 | 82.0006 |
| 53 | 66.5482 | 70.9934 | 75.0019 | 79.8434 | 83.2525 |
| 54 | 67.6728 | 72.1532 | 76.1921 | 81.0688 | 84.5018 |
| 55 | 68.7962 | 73.3115 | 77.3804 | 82.292 | 85.7491 |
| 56 | 69.9185 | 74.4683 | 78.5671 | 83.5136 | 86.994 |
| 57 | 71.0397 | 75.6237 | 79.7522 | 84.7327 | 88.2366 |
| 58 | 72.1598 | 76.7778 | 80.9356 | 85.9501 | 89.477 |
| 59 | 73.2789 | 77.9305 | 82.1174 | 87.1658 | 90.7153 |
| 60 | 74.397 | 79.082 | 83.2977 | 88.3794 | 91.9518 |
| 61 | 75.5141 | 80.2321 | 84.4764 | 89.5912 | 93.1862 |
| 62 | 76.6302 | 81.381 | 85.6537 | 90.8015 | 94.4185 |
| 63 | 77.7454 | 82.5287 | 86.8296 | 92.0099 | 95.6492 |
| 64 | 78.8597 | 83.6752 | 88.004 | 93.2167 | 96.8779 |
| 65 | 79.973 | 84.8206 | 89.1772 | 94.422 | 98.1049 |
| 66 | 81.0855 | 85.9649 | 90.3488 | 95.6256 | 99.3303 |
| 67 | 82.1971 | 87.108 | 91.5193 | 96.8277 | 100.5538 |
| 68 | 83.3079 | 88.2502 | 92.6885 | 98.0283 | 101.7757 |
| 69 | 84.4179 | 89.3912 | 93.8565 | 99.2274 | 102.9961 |
| 70 | 85.527 | 90.5313 | 95.0231 | 100.4251 | 104.2148 |
| 71 | 86.6354 | 91.6703 | 96.1887 | 101.6214 | 105.4323 |
| 72 | 87.7431 | 92.8083 | 97.353 | 102.8163 | 106.6473 |

| | | | | | |
|---|---|---|---|---|---|
| 73 | 88.8499 | 93.9453 | 98.5162 | 104.0098 | 107.8619 |
| 74 | 89.9561 | 95.0815 | 99.6784 | 105.2019 | 109.0742 |
| 75 | 91.0615 | 96.2167 | 100.8393 | 106.3929 | 110.2854 |
| 76 | 92.1662 | 97.351 | 101.9992 | 107.5824 | 111.4954 |
| 77 | 93.2702 | 98.4844 | 103.1581 | 108.7709 | 112.7037 |
| 78 | 94.3735 | 99.617 | 104.3159 | 109.9582 | 113.9107 |
| 79 | 95.4762 | 100.7486 | 105.4727 | 111.144 | 115.1163 |
| 80 | 96.5782 | 101.8795 | 106.6285 | 112.3288 | 116.3209 |
| 81 | 97.6796 | 103.0095 | 107.7834 | 113.5123 | 117.524 |
| 82 | 98.7803 | 104.1387 | 108.9373 | 114.6948 | 118.7261 |
| 83 | 99.8805 | 105.2672 | 110.0902 | 115.8762 | 119.927 |
| 84 | 100.98 | 106.3949 | 111.2422 | 117.0566 | 121.1262 |
| 85 | 102.0789 | 107.5217 | 112.3933 | 118.2356 | 122.3244 |
| 86 | 103.1773 | 108.6479 | 113.5436 | 119.4137 | 123.5218 |
| 87 | 104.275 | 109.7733 | 114.6929 | 120.5909 | 124.7176 |
| 88 | 105.3723 | 110.898 | 115.8415 | 121.7672 | 125.9123 |
| 89 | 106.4689 | 112.022 | 116.989 | 122.9422 | 127.106 |
| 90 | 107.565 | 113.1452 | 118.1359 | 124.1162 | 128.2987 |
| 91 | 108.6606 | 114.2679 | 119.282 | 125.2893 | 129.4902 |
| 92 | 109.7556 | 115.3898 | 120.427 | 126.4616 | 130.6812 |
| 93 | 110.8501 | 116.511 | 121.5714 | 127.633 | 131.8705 |
| 94 | 111.9442 | 117.6317 | 122.7152 | 128.8032 | 133.0589 |
| 95 | 113.0377 | 118.7516 | 123.858 | 129.9725 | 134.2466 |
| 96 | 114.1307 | 119.8709 | 125.0001 | 131.1411 | 135.4327 |
| 97 | 115.2232 | 120.9897 | 126.1414 | 132.3089 | 136.6188 |
| 98 | 116.3153 | 122.1077 | 127.2821 | 133.4756 | 137.803 |
| 99 | 117.4069 | 123.2252 | 128.4219 | 134.6415 | 138.9869 |
| 100 | 118.498 | 124.3421 | 129.5613 | 135.8069 | 140.1697 |

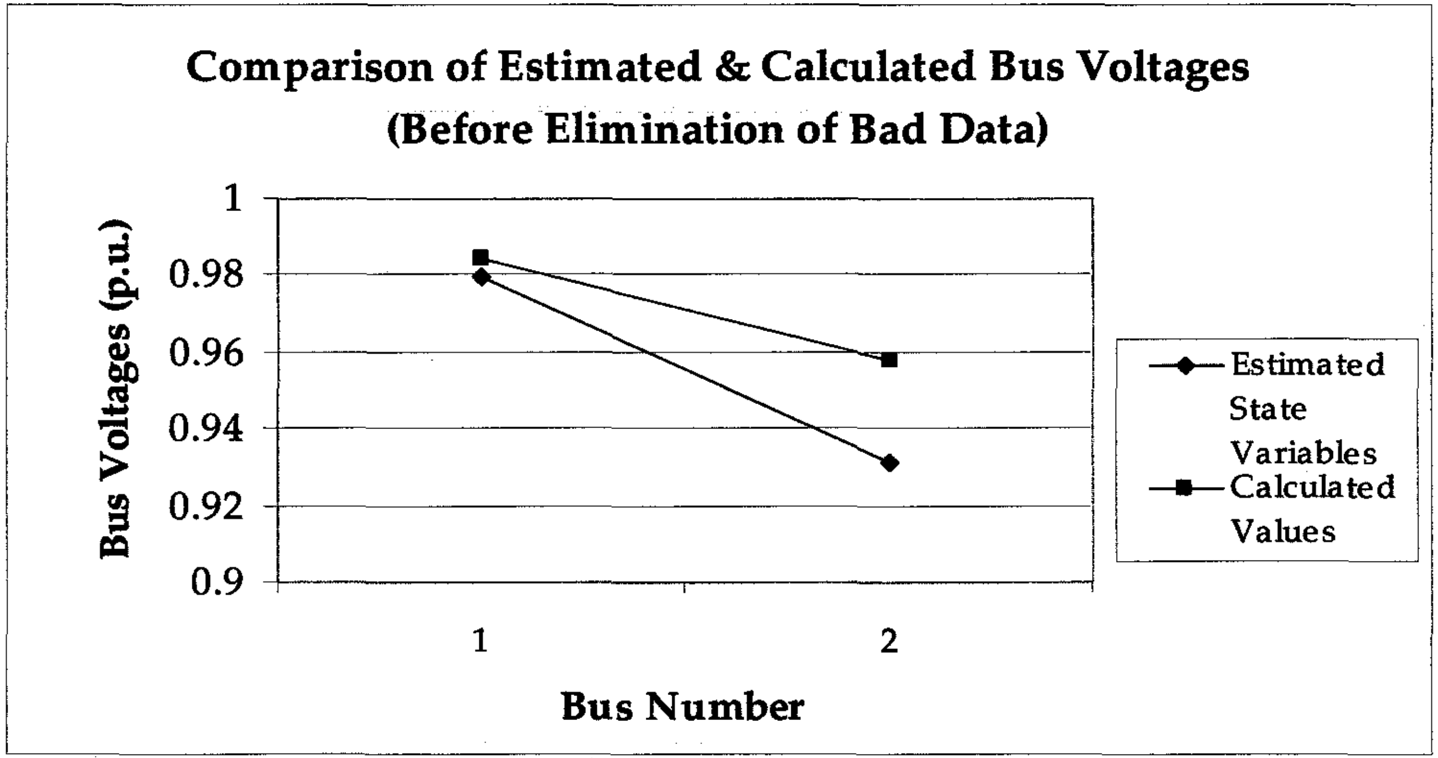
# APPENDIX B

# 2-BUS POWER SYSTEM NETWORK

## COMPARISON OF ESTIMATED & CALCULATED VALUES

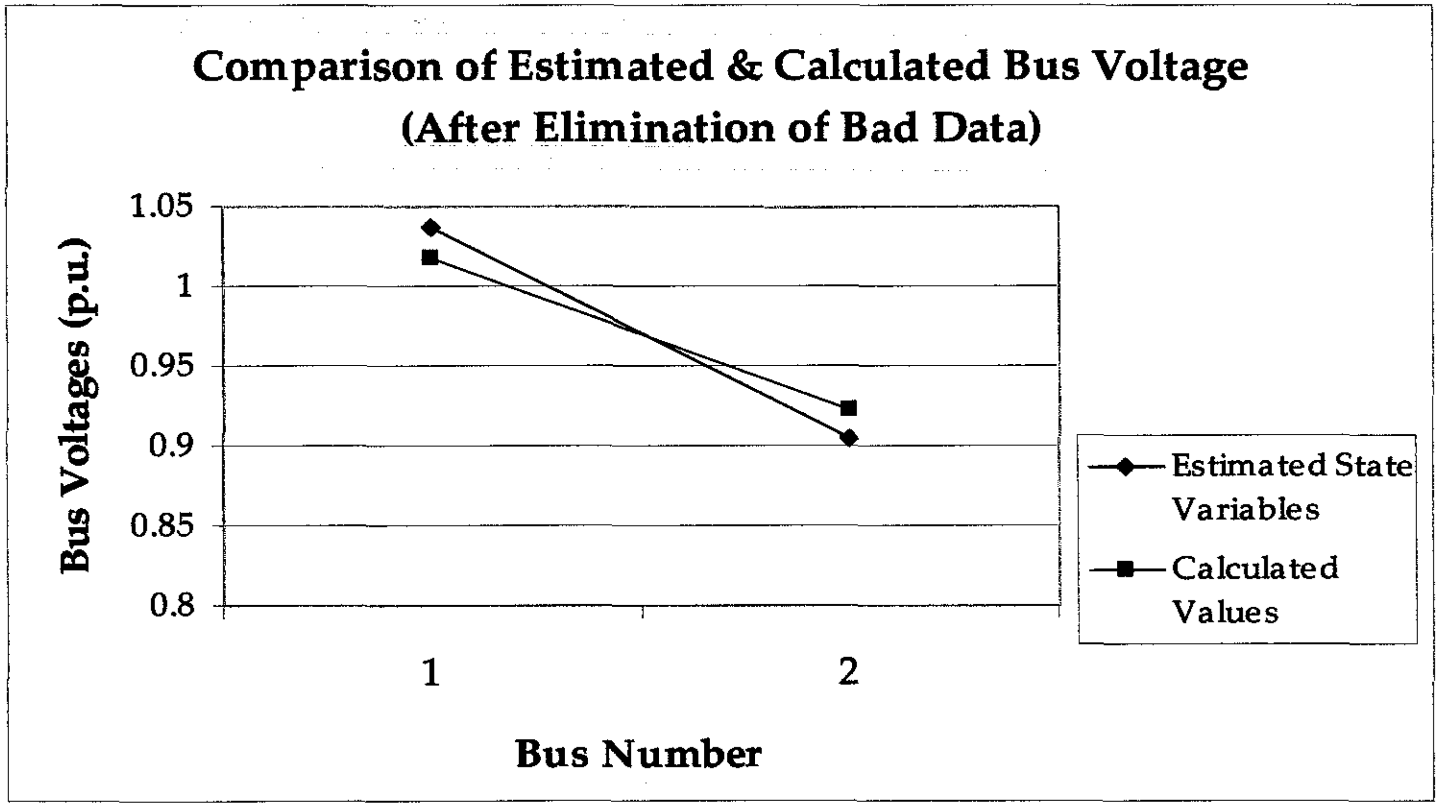
**Before Elimination of Bad Data**

| | Symbols | Estimated State Variables | Calculated Values |
|---|---|---|---|
| **Bus voltages** | $\|V1\|$ | 0.97927 | 0.9843 |
| | $\|V2\|$ | 0.93136 | 0.9578 |
| **Phase angles** | $\delta 1$ | 0 | 0 |
| | $\delta 2$ | -0.1881 | − 0.1762 |


Comparison of Estimated & Calculated Bus Voltages (Before Elimination of Bad Data)

**After Elimination of Bad Data**

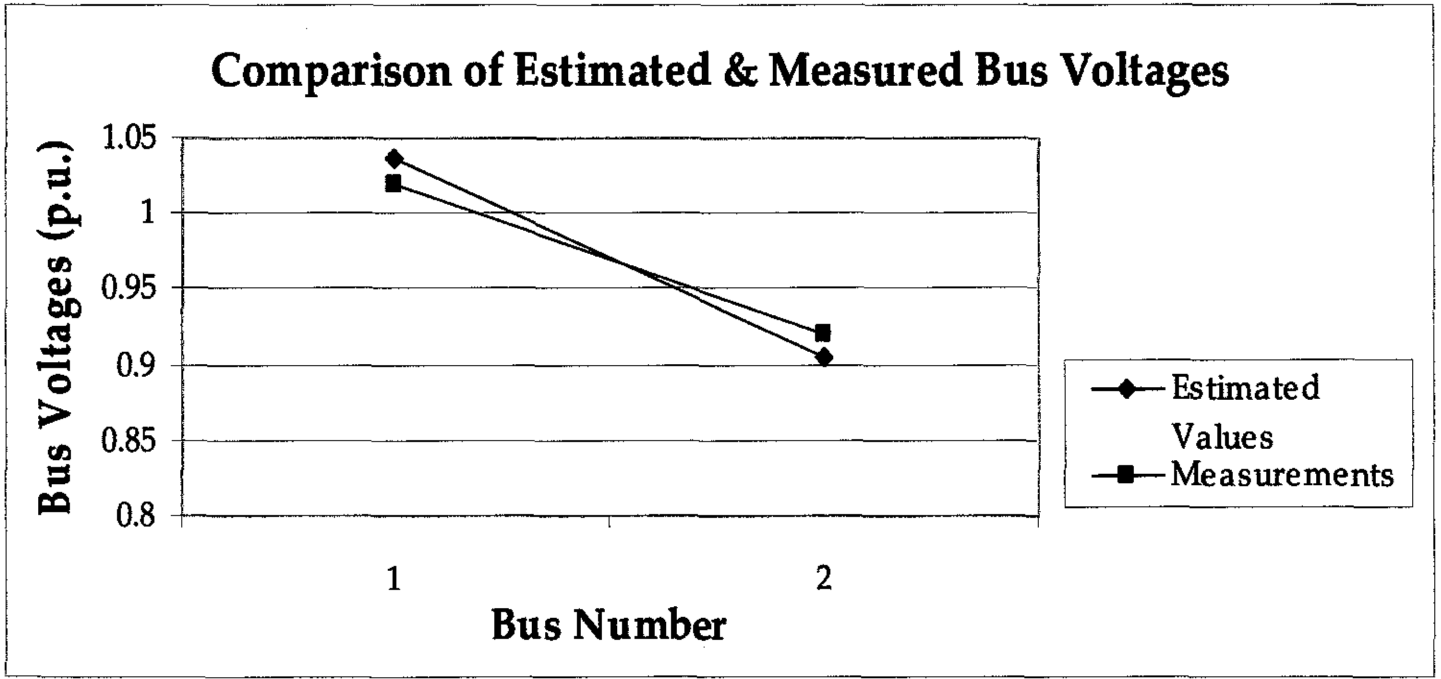
| | Symbols | Estimated State Variables | Calculated Values |
|---|---|---|---|
| **Bus voltages** | $\lvert V1 \rvert$ | 1.03638 | 1.0174 |
| | $\lvert V2 \rvert$ | 0.90534 | 0.9223 |
| **Phase angles** | $\delta 1$ | 0 | 0 |
| | $\delta 2$ | -0.16089 | – 0.1600 |



## COMPARISON OF ESTIMATED & MEASURED VALUES

**After Elimination of Bad Data**

| | Symbols | Estimated Values | Measurements |
|---|---|---|---|
| **Bus voltages** | $\|V1\|$ | 1.03638 | 1.02 |
| | $\|V2\|$ | 0.90534 | 0.92 |
| **Power Injection** | *Q1* | 0.59170 | 0.605 |
| **Power Flow** | *P1-2* | 0.60123 | 0.598 |



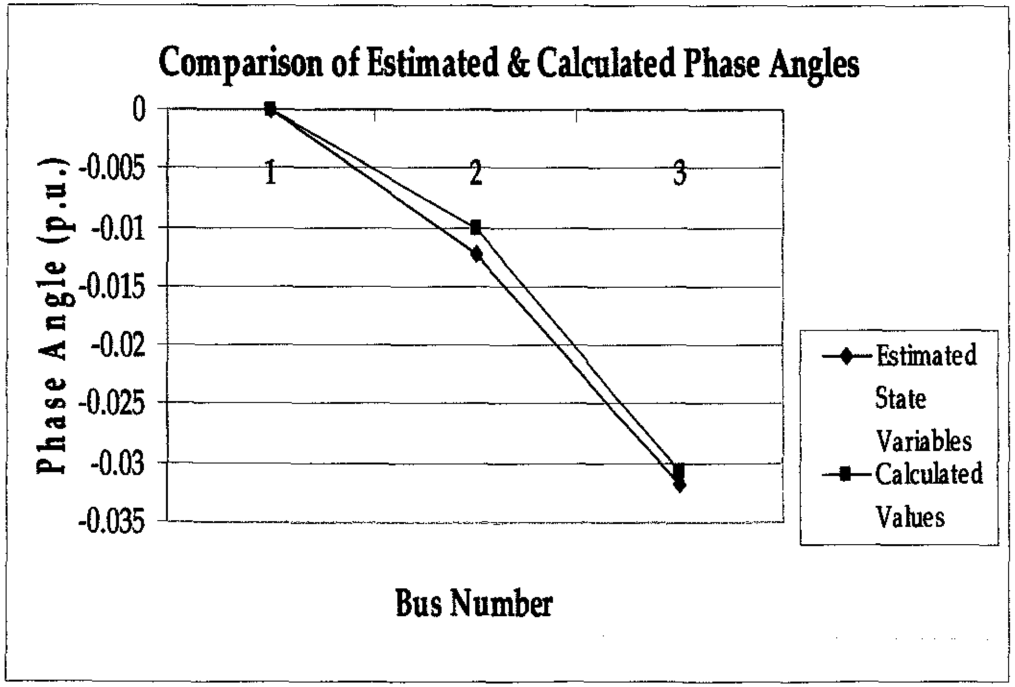
# APPENDIX C

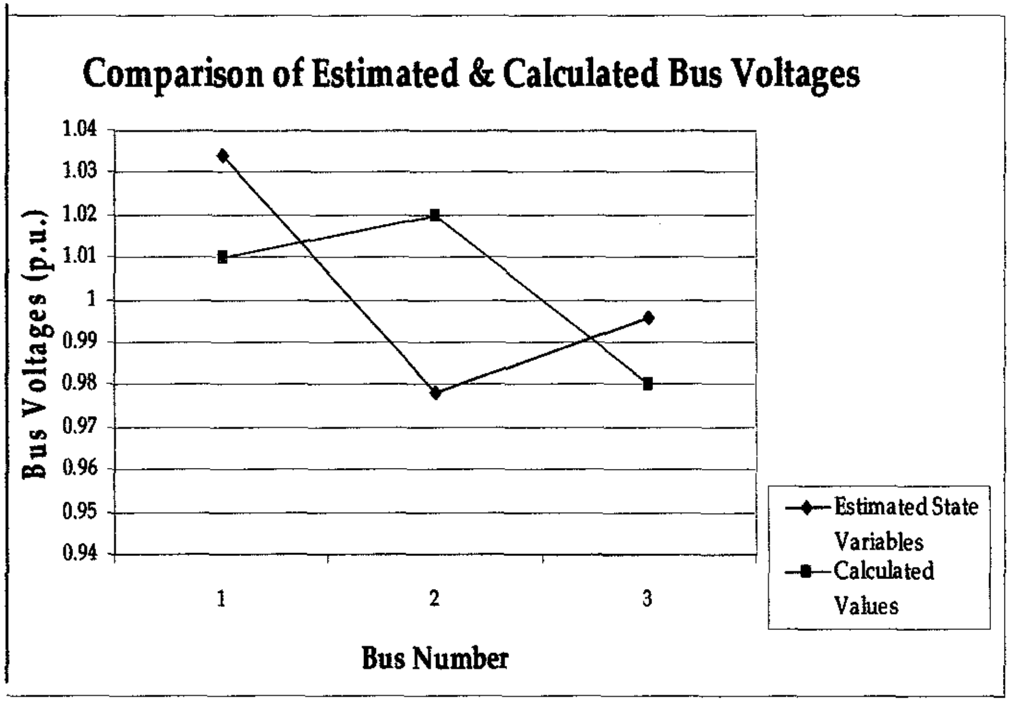
# 3-BUS POWER SYSTEM NETWORK

## COMPARISON OF ESTIMATED & CALCULATED VALUES

**After Detection of Bad Data**

| | Symbols | Estimated State Variables | Calculated Values |
|---|---|---|---|
| **Bus voltages** | $\lvert V1 \rvert$ | 1.03435 | 1.01 |
| | $\lvert V2 \rvert$ | 0.97804 | 1.02 |
| | $\lvert V3 \rvert$ | 0.99589 | 0.98 |
| **Phase angles** | $\delta 1$ | 0.00000 | 0 |
| | $\delta 2$ | -0.01229 | – 0.01014 |
| | $\delta 3$ | -0.03182 | – 0.03074 |

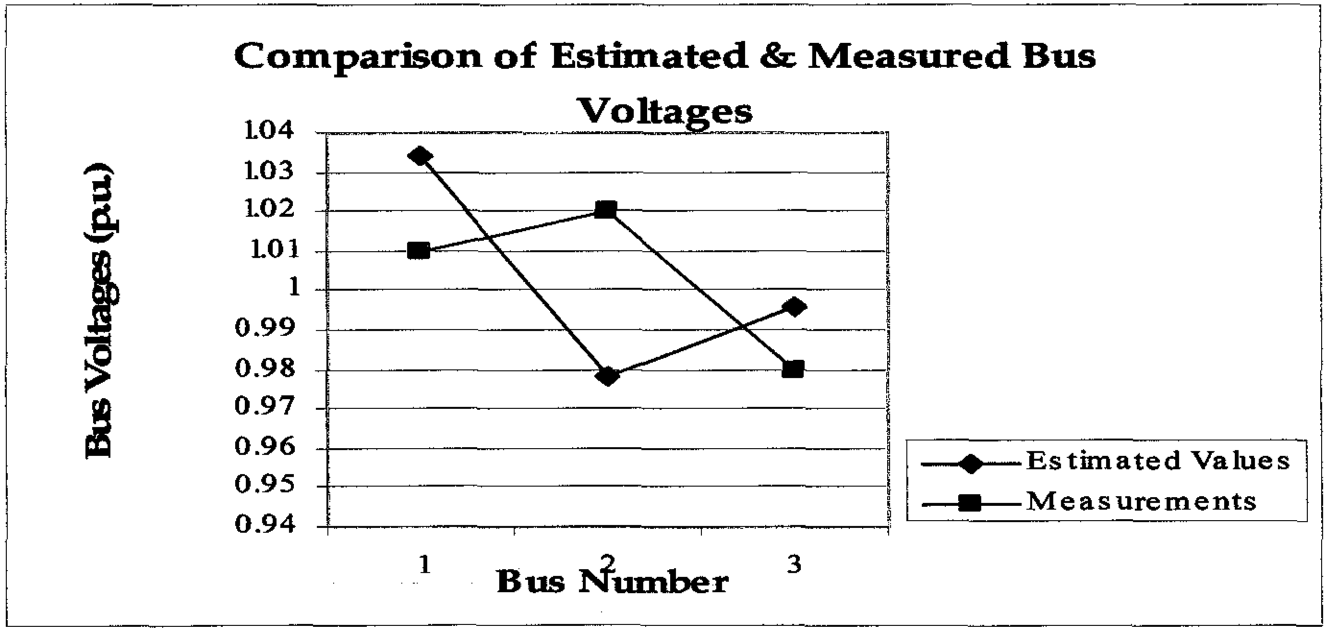


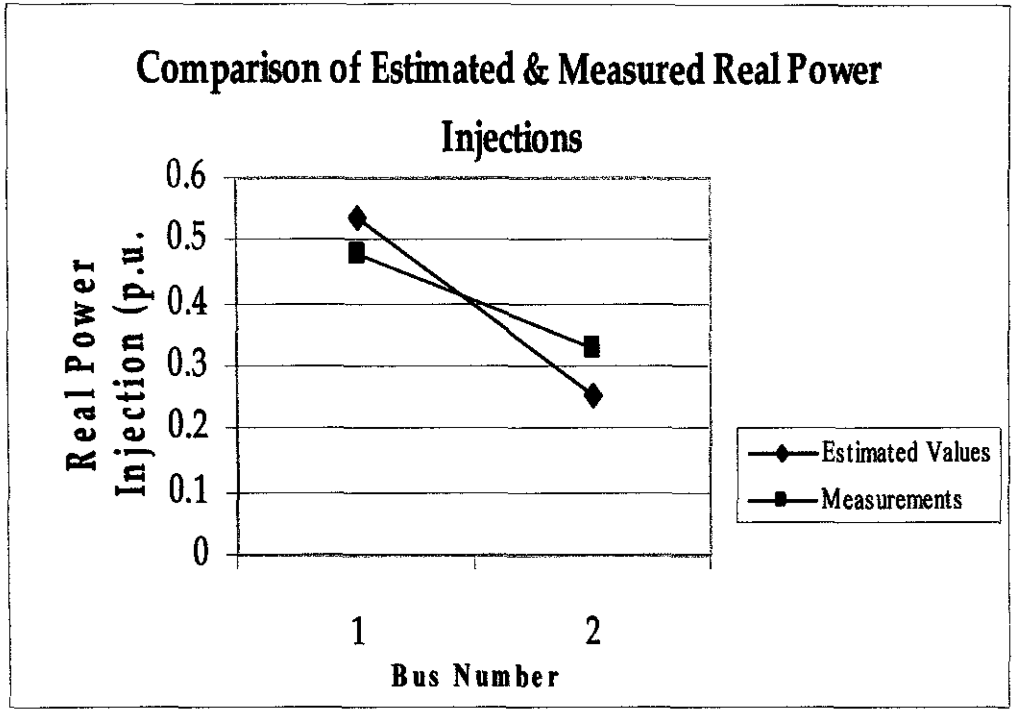


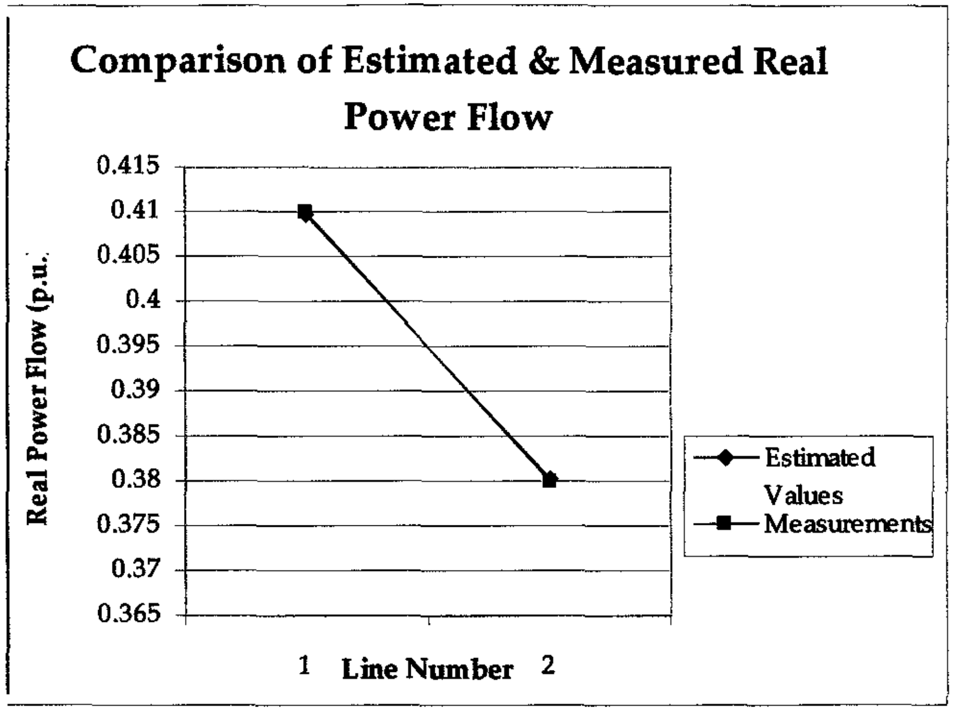
# COMPARISON OF ESTIMATED & MEASURED VALUES

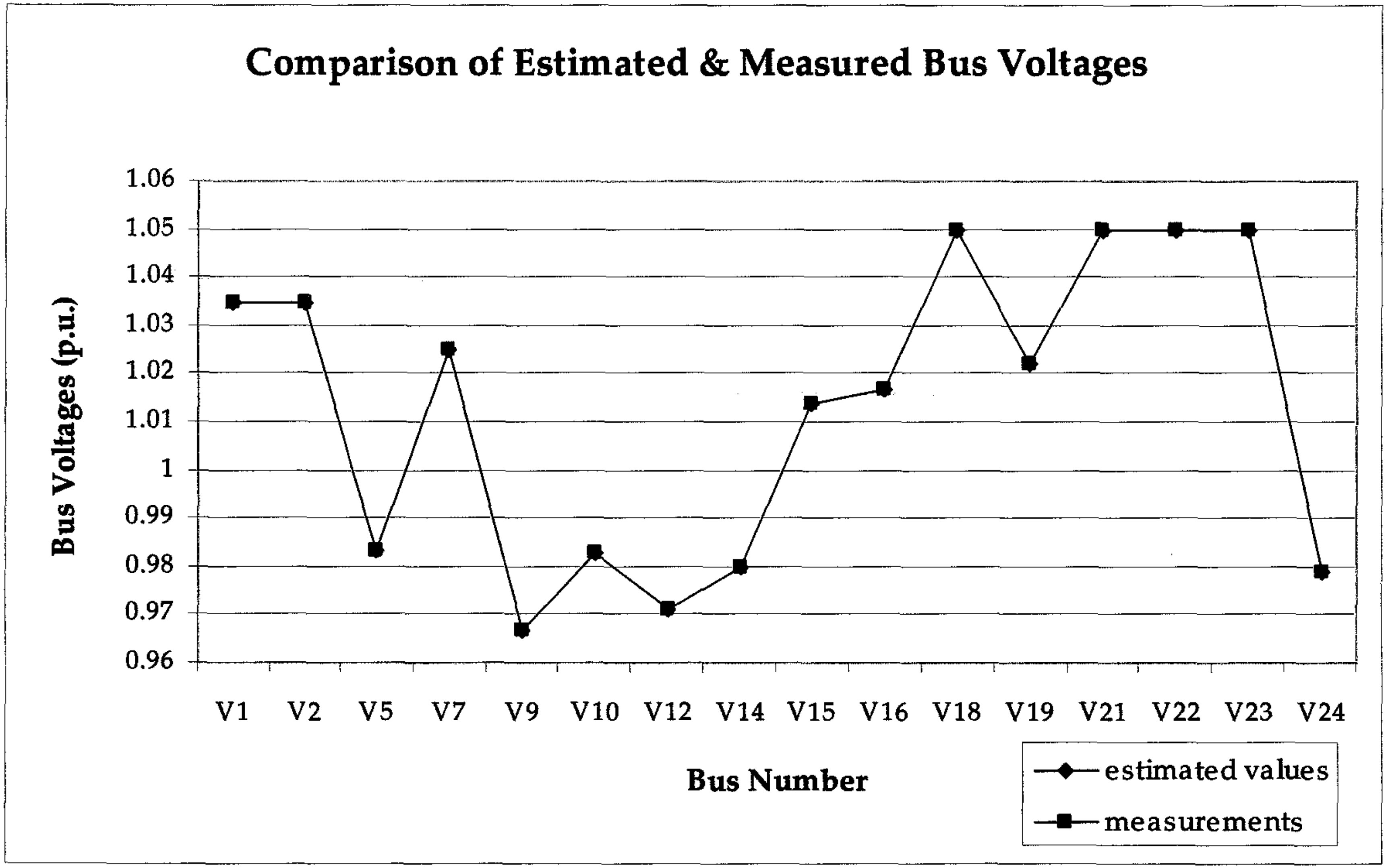
## After Detection of Bad Data

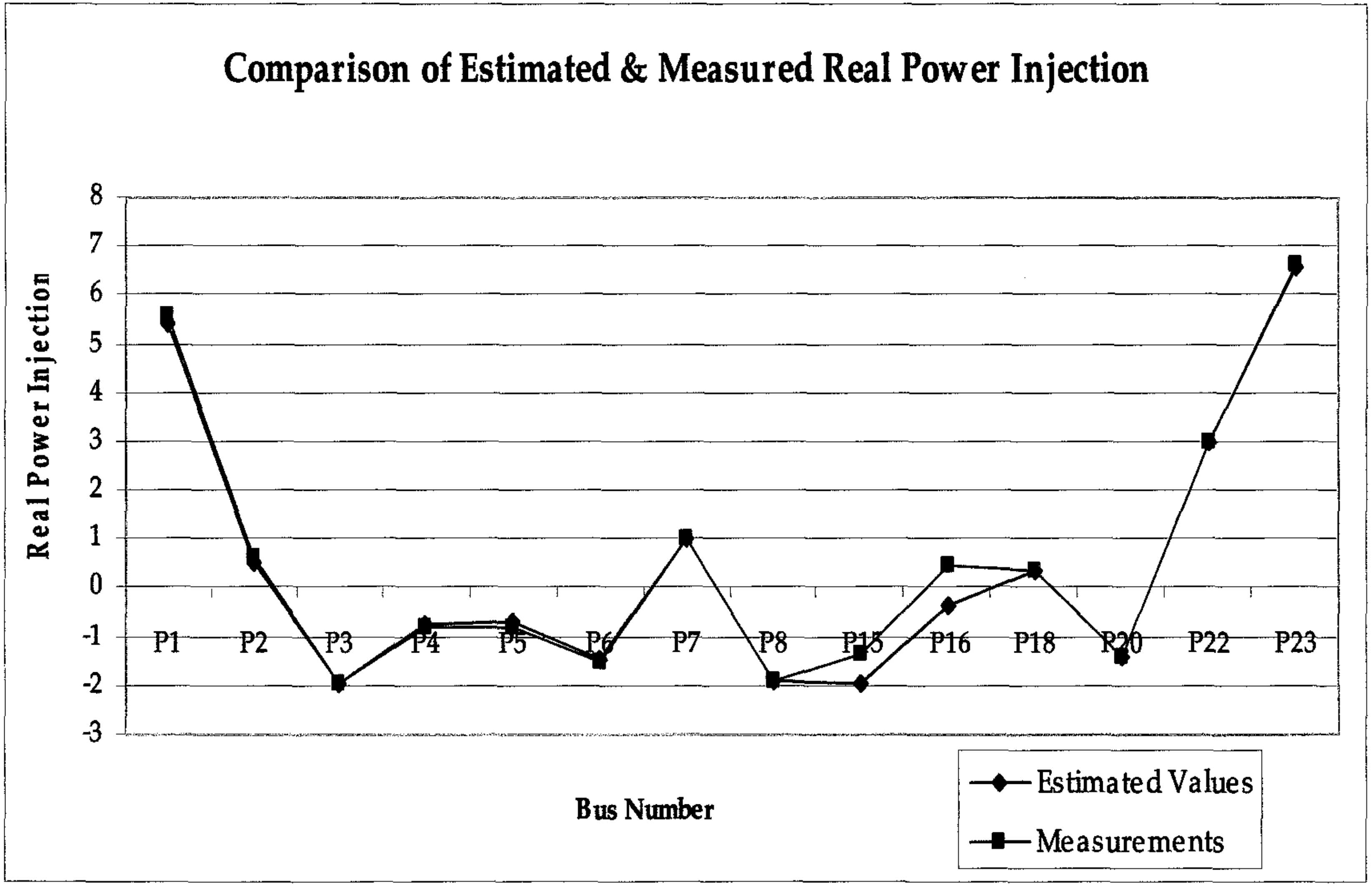
| | Symbols | Estimated Values | Measurements |
|---|---|---|---|
| **Bus voltages** | $\lvert V1\rvert$ | 1.03435 | 1.01 |
| | $\lvert V2\rvert$ | 0.97804 | 1.02 |
| | $\lvert V3\rvert$ | 0.99589 | 0.98 |
| **Power Injection** | *P1* | 0.59170 | 0.605 |
| | *P2* | 0.53395 | 0.48 |
| **Power Flow** | *P1-3* | 0.40961 | 0.41 |
| | *P2-3* | 0.38035 | 0.38 |

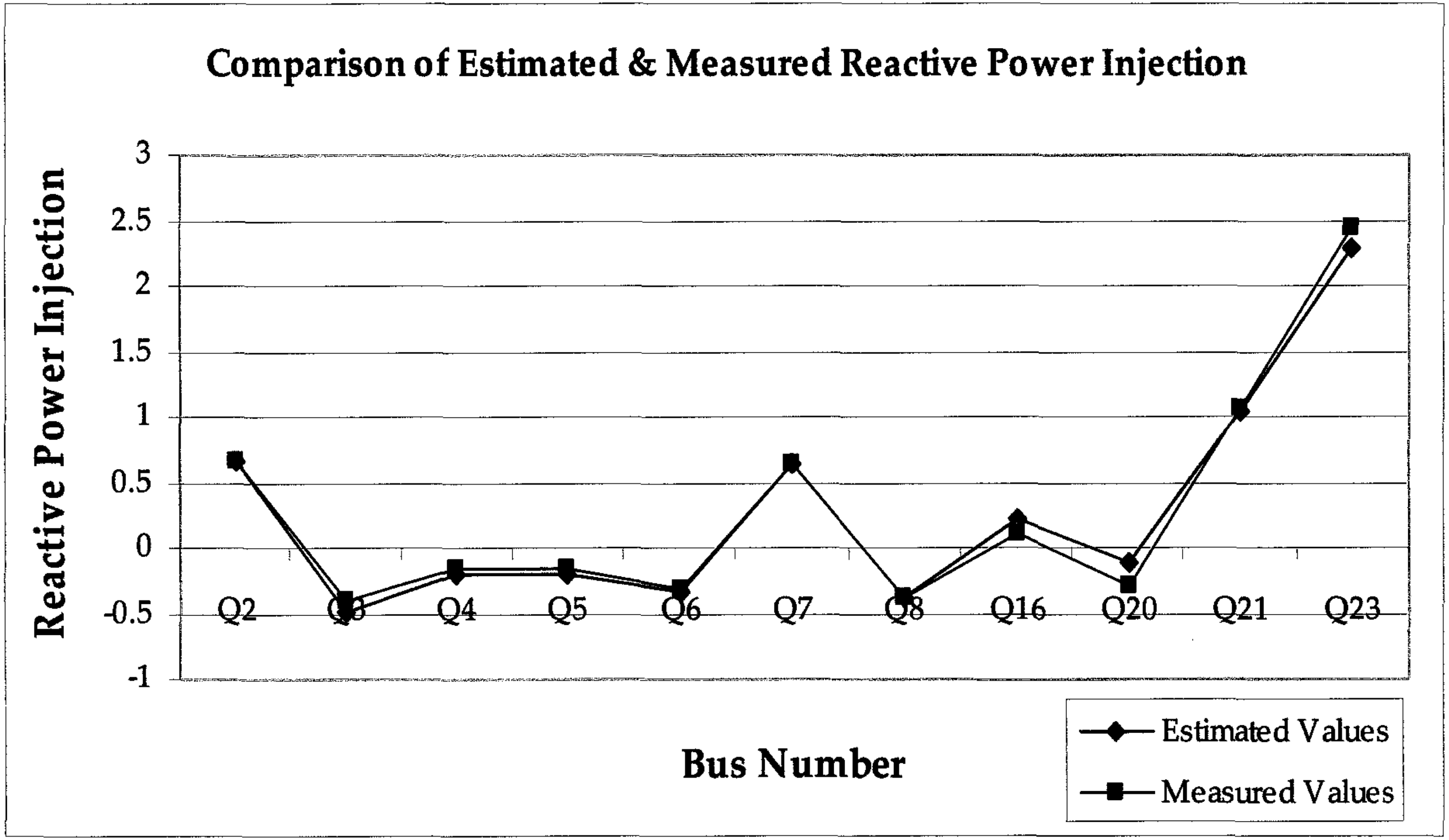


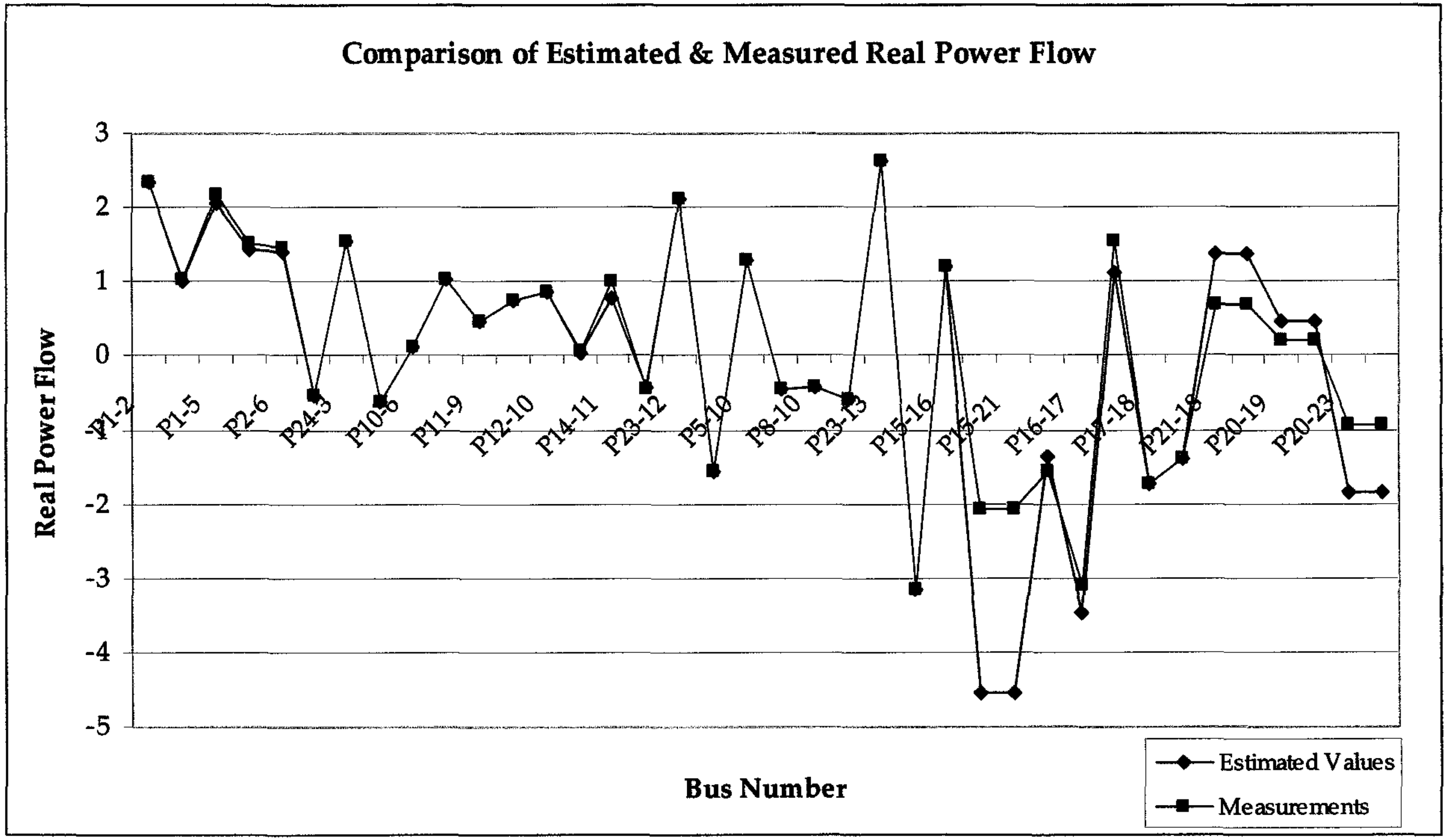

Comparison of Estimated & Measured Real Power Injections
Real Power Injection (p.u.)
0.6
0.5
0.4
0.3
0.2
0.1
0
1
2
Bus Number
Estimated Values
Measurements

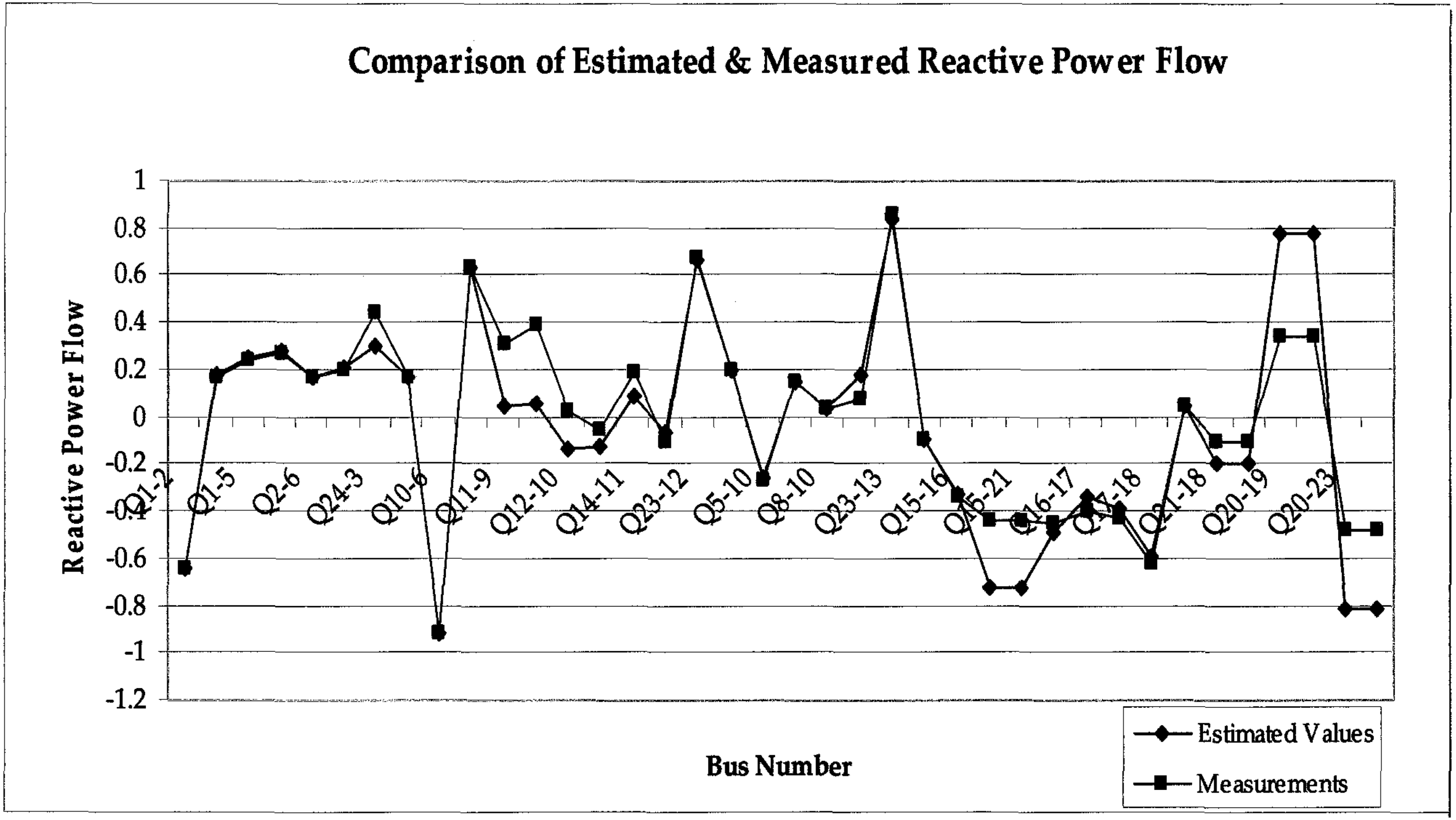

Comparison of Estimated & Measured Real Power Flow
Real Power Flow (p.u.)
0.415
0.41
0.405
0.4
0.395
0.39
0.385
0.38
0.375
0.37
0.365
1
Line Number
2
Estimated Values
Measurements


Comparison of Estimated & Measured Bus Voltages
Bus Voltages (p.u.)
1.06
1.05
1.04
1.03
1.02
1.01
1
0.99
0.98
0.97
0.96
V1
V2
V5
V7
V9
V10
V12
V14
V15
V16
V18
V19
V21
V22
V23
V24
Bus Number
estimated values
measurements


Comparison of Estimated & Measured Real Power Injection
Real Power Injection
8
7
6
5
4
3
2
1
0
-1
-2
-3
P1
P2
P3
P4
P5
P6
P7
P8
P15
P16
P18
P20
P22
P23
Bus Number
Estimated Values
Measurements


Comparison of Estimated & Measured Reactive Power Injection
Reactive Power Injection
3
2.5
2
1.5
1
0.5
0
-0.5
-1
Q2
Q3
Q4
Q5
Q6
Q7
Q8
Q16
Q20
Q21
Q23
Bus Number
Estimated Values
Measured Values


Comparison of Estimated & Measured Real Power Flow
Real Power Flow
3
2
1
0
-1
-2
-3
-4
-5
P1-2
P1-5
P2-6
P24-3
P10-6
P11-9
P12-10
P14-11
P23-12
P5-10
P8-10
P23-13
P15-16
P15-21
P16-17
P17-18
P21-18
P20-19
P20-23
Bus Number
Estimated Values
Measurements


Comparison of Estimated & Measured Reactive Power Flow
Reactive Power Flow
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
Q1-2
Q1-5
Q2-6
Q24-3
Q10-6
Q11-9
Q12-10
Q14-11
Q23-12
Q5-10
Q8-10
Q23-13
Q15-16
Q15-21
Q16-17
Q17-18
Q21-18
Q20-19
Q20-23
Bus Number
Estimated Values
Measurements