

POWER FLOW ANALYSIS SOFTWARE USING MATLAB

MOHD SHAHIMI BIN MOHAMAD ISA

UNIVERSITY MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

**BORANG PENGESAHAN STATUS TESIS ♦**

**JUDUL: POWER FLOW ANALYSIS SOFTWARE USING MATLAB**

**SESI PENGAJIAN: 2007/2008**

Saya MOHD SHAHIMI BIN MOHAMAD ISA (850220-07-5567)  
(HURUF BESAR)

mengaku membenarkan tesis (Sarjana Muda/~~Sarjana~~ /~~Doktor Falsafah~~)\* ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Kolej Universiti Kejuruteraan & Teknologi Malaysia.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. \*\*Sila tandakan ( √ )

**SULIT**

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

**TERHAD**

(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

**TIDAK TERHAD**

Disahkan oleh:

\_\_\_\_\_  
(TANDATANGAN PENULIS)

\_\_\_\_\_  
(TANDATANGAN PENYELIA)

Alamat Tetap:

**75 BLOK I MK 12.**  
**KAMPUNG BAROH,**  
**11900 BAYAN LEPAS.**  
**PULAU PINANG.**

**MRS. NORHAFIDZAH BT.**  
**MOHD SAAD**  
( Nama Penyelia )

Tarikh: **30 NOVEMBER 2007**

Tarikh: : **30 NOVEMBER 2007**

- CATATAN:
- \* Potong yang tidak berkenaan.
  - \*\* Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh tesis ini perlu dikelaskan sebagai atau TERHAD.
  - ♦ Tesis dimaksudkan sebagai tesis bagi Ijazah doktor Falsafah dan Sarjana secara Penyelidikan, atau disertasi bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjana Muda (PSM).

“I hereby acknowledge that the scope and quality of this thesis is qualified  
for the award of the Bachelor Degree of Electrical Engineering (Power  
System)”

Signature: \_\_\_\_\_

Name: NORHAFIDZAH BT MOHD SAAD

Date: 30 NOVEMBER 2007

# POWER FLOW ANALYSIS SOFTWARE USING MATLAB

MOHD SHAHIMI BIN MOHAMAD ISA

This thesis is submitted as partial fulfillment of the requirements for the award of the Bachelor Degree of Electrical Engineering (Power System)

Faculty of Electrical & Electronics Engineering

University Malaysia Pahang

NOVEMBER 2007

“All the trademark and copyrights use here in are property of their respective owner. References of information from other sources are quoted accordingly; otherwise the information presented in this report is solely work of the author.”

Signature : \_\_\_\_\_

Author : MOHD SHAHIMI BIN MOHAMAD ISA

Date : 26 NOVEMBER 2007

Specially dedicated to  
my beloved family and those people who have guided and inspired me  
throughout my journey of education.

## ACKNOWLEDGEMENT

First and foremost, I am very grateful to the almighty ALLAH S.W.T for giving me this opportunity to accomplish my Final Year Project.

Firstly, I wish to express my deep gratitude to my supervisor, Mrs. Norhafidzah bt. Mohd Saad for all his valuable guidance, assistance and support all through this work.

Secondly, I wish to thank lecturers and technicians, for their suggestions and support on this project. Their comments on this project are greatly appreciated. My thanks are also to all my friends who have involved and helped me in this project.

Most importantly I extend my gratitude to my parents who have encouraged me throughout my education and I will always be grateful for their sacrifice, generosity and love.

## ABSTRACT

Power flow analysis is the backbone of power system analysis and design. They are necessary for planning, operation, economic scheduling and exchange of power between utilities. The principal information of power flow analysis is to find the magnitude and phase angle of voltage at each bus and the real and reactive power flowing in each transmission lines. Power flow analysis is an importance tool involving numerical analysis applied to a power system. In this analysis, iterative techniques are used due to there no known analytical method to solve the problem. To finish this analysis there are methods of mathematical calculations which consist plenty of step depend on the size of system. This process is difficult and takes a lot of times to perform by hand. The objective of this project is to develop a toolbox for power flow analysis that will help the analysis become easier. Power flow analysis software package develops by the author use MATLAB programming and MATLAB GUI. Data visualization and GUI design in MATLAB are based on the Handle Graphics System in which the objects organized in a Graphics Object Hierarchy can be manipulated by various high and low level commands. This software provides all three methods that commonly used, Newton Raphson method, Gauss-Seidel method and Fast Decoupled method in solving the power flow or load flow problem.



## ABSTRAK

Analisis sistem kuasa merupakan suatu asas kepada semua analisis sistem kuasa dan pelan sistem kuasa. Analisis sistem kuasa adalah penting di dalam pelan, operasi, pelan ekonomi, dan pertukaran kuasa antara pengedar tenaga. Asas analisis sistem kuasa ialah mendapatkan nilai voltan busbar dan sudut fasa bagi voltan serta menganalisa aliran kuasa didalam talian. Analisis sistem kuasa terdiri daripada analisa pengiraan keatas sesuatu sistem kuasa. Teknik lelaran diperlukan didalam analisis ini kerana ia melibatkan set penyelesaian yang tidak linear. Untuk menyelesaikan analisis ini memerlukan banyak pengiraan berulang kali bergantung kepada saiz sesuatu sistem kuasa. Analisis ini adalah amat sukar dan memerlukan masa yang panjang bagi menyelesaikannya. Tujuan utama projek ini adalah membina suatu perisian komputer bagi memudahkan analisis system kuasa ini dilakukan. Perisian ini dibangunkan dengan menggunakan aturcara serta grafik antaramuka didalam MATLAB. Gambaran data dan pelan GUI didalam MATLAB berdasarkan system pengendalian grafik dimana objek-objek diatur didalam hierarki objek grafik dikendalikan menggunakan berbagai arahan. Perisian analisis sistem kuasa ini menyediakan tiga jenis penyelesaian, kaedah Newton-Raphson, kaedah Gauss-Seidel dan kaedah Fast Decoupled.

**TABLE OF CONTENTS**

<b>CHAPTER</b>	<b>TITLE</b>	<b>PAGE</b>
	<b>TITLE PAGE</b>	i
	<b>DECLARATION</b>	ii
	<b>DEDICATION</b>	iii
	<b>ACKNOWLEDGEMENT</b>	iv
	<b>ABSTRACT</b>	v
	<b>ABSTRAK</b>	iv
	<b>TABLE OF CONTENTS</b>	vii
	<b>LIST OF TABLES</b>	x
	<b>LIST OF FIGURES</b>	xii

## TABLE OF CONTENTS

<b>CHAPTER</b>	<b>TITLE</b>	<b>PAGE</b>
<b>1</b>	<b>INTRODUCTION</b>	
	1.1 Overview	1
	1.2 Objective	2
	1.3 Scope of Project	3
	1.4 Thesis Organization	3
<b>2</b>	<b>LITERATURE REVIEW</b>	
	2.1 Introduction	5
	2.2 Power Flow Analysis	5
	2.2.1 Newton-Raphson Method	6
	2.2.2 Gauss-Seidel Method	7
	2.2.3 Fast Decoupled method	8
	2.3 Line Flow and Losses	12
	2.4 MATLAB GUI	13
	2.5 Similar Software in market	15
<b>3</b>	<b>METHODOLOGY</b>	
	3.1 Introduction	17
	3.2 Project Flow Chart	18
	3.3 Developing GUI using MATLAB GUIDE	19
	3.4 Problems on Power System	23
	3.4.1 Example of 3-bus Power System	23
	3.4.2 Example of IEEE 30-Bus Power System	25

<b>4</b>	<b>RESULTS AND DISCUSSIONS</b>	
4.1	Introduction	29
4.2	Power Flow Analysis Toolbox Package Ver. 1.0 (POWERSOLVE)	29
4.2.1	Introduction	29
4.2.2	Regarding the Software	31
	4.2.2.1 Tutorial & Examples	32
	4.2.2.2 POWERSOLVE Program	42
4.3	Power Flow Input Data Formula	46
4.4	Power Flow Analysis Simulation using MATLAB	47
4.4.1	Result on 3-Bus System Analysis	47
	4.4.1.1 Newton-Raphson Method	47
	4.4.1.2 Gauss-Seidel Method	48
	4.4.1.3 Fast Decoupled Method	49
	4.4.1.4 Discussion on Analysis	51
4.4.2	Result on IEEE 30-Bus System Analysis	52
	4.4.2.1 Newton-Raphson Method	52
	4.4.2.2 Gauss-Seidel Method	57
	4.4.2.3 Fast Decoupled Method	62
	4.4.2.4 Discussion on Analysis	67
4.5	Discussion & Analysis	67
<b>5</b>	<b>CONCLUSION</b>	
5.1	Conclusion	68
5.2	Future Recommendations	69
5.2.1	Commercialization	69
	<b>REFERENCES</b>	<b>70</b>
	APPENDIX A-C	71-78

## LIST OF TABLES

<b>TABLE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
3.1	Basic MATLAB GUI Component	21
3.2	Bus Data for 3-Bus System	24
3.3	Line Data for 3-Bus System	24
3.4	Regulated Bus Data	26
3.5	Transformer Data	26
3.6	Q injection Data	26
3.7	Bus Data for IEEE 30 Bus System	27
3.8	Line Data for IEEE 30 Bus System	28
4.1	Bus Data Format	46
4.2	Line Data Format	47
4.3	Newton-Raphson Analysis on 3-Bus System	47
4.4	Newton-Raphson Line Loss on 3-Bus System	48
4.5	Gauss-Seidel Analysis on 3-Bus System	48
4.6	Gauss-Seidel Line Loss on 3-Bus System	49
4.7	Fast Decoupled Analysis on 3-Bus System	49
4.8	Fast Decoupled Line Loss on 3-Bus System	50
4.9	Simulation Result for 3-Bus System	51
4.10	Newton-Raphson Analysis on 30-Bus System	52
4.11	Newton-Raphson Line Loss on 30-Bus System	53
4.12	Gauss-Seidel Analysis on 30-Bus System	57
4.13	Gauss-Seidel Line Loss on 30-Bus System	58

4.14	Fast Decoupled Analysis on 30-Bus System	62
4.15	Fast Decoupled Line Loss on 30-Bus System	63
4.16	Simulation Result for IEEE 30-Bus System	67

## LIST OF FIGURES

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
2.1	Transmission Line Model for calculation line flow	12
2.2	A Windows-based Interactive and Graphic Package	16
3.1	Flowchart for Whole Project	18
3.2	MATLAB GUIDE Layout Editor	19
3.3	Property Inspector	20
3.4	Example of GUI	22
3.5	Example of M-files for GUI	23
3.6	Example of 3 Bus System	23
3.7	IEEE 30 Bus System	25
4.1	Addpath	30
4.2	Main Menu of GUI	31
4.3	General information of POWERSOLVE	32
4.4	Exit Button Confirmation	32
4.5	Tutorial & Examples Menu	33
4.6	Tutorial	34
4.7	Bus Data Format	34
4.8	Line Data Format	35
4.9	Example 1	36
4.10	Example 1 - Data Preparation	36
4.11	Example 1 - Result	37
4.12	Example 1 – Power Flow Analysis	37
4.13	Example 1 - Line Flow Analysis	38

4.14	Example 2	39
4.15	Example 2 - Data Preparation	40
4.16	Example 2 - Result	40
4.17	Example 2 - Power Flow Analysis	41
4.18	Example 2 - Line Flow Analysis	42
4.19	Input Data Window	43
4.20	Result Data Window	44
4.21	Power Flow Analysis Window	45
4.22	Line Flow Analysis Window	45



## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Overview**

Power flow analysis is the backbone of power system analysis and design. They are necessary for planning, operation, economic scheduling and exchange of power between utilities. Power flow analysis is required for many other analyses such as transient stability, optimal power flow and contingency studies. The principal information of power flow analysis is to find the magnitude and phase angle of voltage at each bus and the real and reactive power flowing in each transmission lines.

Power flow analysis is an importance tool involving numerical analysis applied to a power system. In this analysis, iterative techniques are used due to there no known analytical method to solve the problem. This resulted nonlinear set of equations or called power flow equations are generated. To finish this analysis there are methods of mathematical calculations which consist plenty of step depend on the size of system. This process is difficult and takes much time to perform by hand. By develop a toolbox for power flow analysis surely will help the analysis become easier.

Power flow analysis software can help users to calculate the power flow problem. Over the past decade, a few versions of educational software packages using

advanced programming languages, such as C, C++, Pascal, or FORTRAN have been developed for power engineering curriculums. These choose an integrated study platform with support of database and GUI functions.

Power flow analysis software develops by the author use MATLAB software. MATLAB as a high-performance language for technical computation integrates calculation, visualization and programming in an easy-to-use environment, thus becomes a standard instructional tool for introductory and advanced courses in mathematics, engineering and science in the university environment. Most of the students are familiar with it.

MATLAB is viewed by many users not only as a high-performance language for technical computing but also as a convenient environment for building graphical user interfaces (GUI). Data visualization and GUI design in MATLAB are based on the Handle Graphics System in which the objects organized in a Graphics Object Hierarchy can be manipulated by various high and low level commands. If using MATLAB7 the GUI design more flexible and versatile, they also increase the complexity of the Handle Graphics System and require some effort to adapt to.

## **1.2 Objectives**

The overall aim of the whole project is to develop a toolbox that allow user to solve power flow problem. However the other objective that needed to complete are:

- i. To study the performance of the transmission lines, transformer and generator at steady state condition.
- ii. To obtain the simulation of power flow analysis by MATLAB.
- iii. To build a toolbox for power flow analysis for education and training purposes

### **1.3 Scope of Project**

This project concentrates on MATLAB programming and GUI designing to enable the users to calculate power flow problem. MATLAB is used to program the power flow solution and Graphical User Interface (GUI) use to help a user easy to use.

To achieve all the project's objectives, the developer must have fulfilled all the scope below:

- i. Studies MATLAB programming and MATLAB GUI
- ii. Identify appropriate command for MATLAB M-files
- iii. Build MATLAB program for the power flow analysis using M-files
- iv. Run simulation of power flow analysis using MATLAB for small, medium and large scale system.
- v. Design window for Power Flow Analysis Toolbox using MATLAB GUI

### **1.4 Thesis Organization**

This thesis consists of five chapters including this chapter. The contents of each chapter are outlined as follows;

Chapter 2 contains a detailed description each part of project. It will explain about the MATLAB GUIDE and MATLAB Programming.

Chapter 3 includes the project methodology. This will explain how the project is organized and the flow of the process in completing this project.

Chapter 4 presents the expected result of simulation runs using MATLAB GUIDE.

Finally the conclusions and future recommendation of this project are presented in Chapter 5.

## **CHAPTER 2**

### **LITERATURE REVIEW**

#### **2.1 Introduction**

This chapter discuss about literature review that been collected for this project. Author has refers through journals and paper especially from IEEE. This chapter consists of three parts. It is describe generally on power flow analysis problems and the solutions, Graphical User Interface in MATLAB and power system toolbox in market.

#### **2.2 Power Flow Analysis**

In power engineering, the power flow analysis (also known as load-flow study) is an importance tool involving numerical analysis applied to a power system. Unlike traditional circuit analysis, a power flow study usually uses simplified notation such as a one-line diagram and per-unit system, and focuses on various form of AC power (ie: reactive, real and apparent) rather than voltage and current. The advantage in studying power flow analysis is in planning the future expansion of power systems as well as in determining the best operation of existing systems. Power flow analysis is being used for solving power flow problem. There are three methods can be used to solve power flow analysis. The methods are Newton-Raphson method, Fast-Decoupled method and

Gauss-Seidel method. This sub-chapter will discuss all three methods generally on formula or mathematical step in order to solve power flow problem.

### 2.2.1 Newton-Raphson Method

Newton-Raphson method is commonly use and introduce in most text book. This method widely used for solving simultaneous nonlinear algebraic equations. A Newton-Raphson method is a successive approximation procedure based on an initial estimate of the one-dimensional equation given by series expansion.

*The Newton-Raphson method using the bus admittance matrix in either first or second – order expansion of Taylor series has been evaluate as a best solution for the reliability and the rapid convergence [3].*

$$f(x)=c \quad (1)$$

If  $x^{(0)}$  is an initial estimate of the solution, and  $\Delta x^{(0)}$  is a small deviation from the correct solution, we must have

$$f(x^{(0)} + \Delta x^{(0)})=c \quad (2)$$

Expanding the left-hand side of the above equation in Taylor's series about  $x^{(0)}$  yields

$$f(x^{(0)}) + (df/dx)^{(0)} \Delta x^{(0)} + 1/2! (d^2f/dx^2)^{(0)} (\Delta x^{(0)})^2 + \dots = c \quad (3)$$

Assuming the error  $\Delta x^{(0)}$  is very small, the higher-order terms can be neglected, which result in

$$\Delta c^{(0)} \approx (df/dx)^{(0)} \Delta x^{(0)} \quad (4)$$

where

$$\Delta c^{(0)} = c - f(x^{(0)})$$

Adding  $\Delta x^{(0)}$  to the initial estimate will result in the second approximation

$$x^{(1)} = x^{(0)} + \Delta c^{(0)} / (df/dx)^{(0)} \quad (5)$$

Successive use of this procedure yields the Newton-Raphson algorithm

$$\Delta c^{(k)} = c - f(x^{(k)}) \quad (6)$$

$$\Delta x^{(k)} = \Delta c^{(k)} / (df/dx)^{(k)} \quad (7)$$

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)} \quad (8)$$

(7) can be rearranged as

$$\Delta c^{(k)} = j^{(k)} \Delta x^{(k)} \quad \text{where } j^{(k)} = (df/dx)^{(k)} \quad (9)$$

*In power system analysis,  $J^{(k)}$  is called the Jacobian matrix. Element of this matrix are the partial derivatives evaluated at  $X^{(k)}$ . It is assumed that  $J^{(k)}$  has an inverse during each iteration. Newton's method, as applied to a set of nonlinear equations reduces the problem to solving a set of linear equations in order to determine the values that improve the accuracy of the estimates. [1]*

### 2.2.2 Gauss-Seidel Method

Gauss-Seidel method is also known as the method of successive displacements. To illustrate the technique, consider the solution of the nonlinear equation given by

$$F(x) = 0 \quad (10)$$

Above function is rearrange and writes as

$$x = g(x) \quad (11)$$

If  $x^{(k)}$  is an initial estimate of the variable  $x$ , the following iterative sequence is formed

$$X^{(k+1)} = g(x^{(k)}) \quad (12)$$

A solution is obtained when the difference between the absolute value of the successive iteration is less than a specified accuracy, i.e.,

$$|x^{(+k)} - x^{(k)}| \leq \varepsilon \quad (13)$$

Where  $\varepsilon$  is the desire accuracy

*The process is repeated until the change in variable is within the desired accuracy. So the Gauss-Seidel method needs much iteration to achieve the desired accuracy, and there is no guarantee for the convergence.[1]*

### **2.2.3 Fast Decoupled Method**

When solving large scale power transmission systems, an alternative strategy for improving computational efficiency and reducing computer storage requirements is the decoupled power flow method, which makes use of an approximate version of the Newton-Raphson procedure.

The Fast decoupled power flow solution requires more iterations than the Newton-Raphson method, but requires considerably less time per iteration and a power flow solution is obtained very rapidly. This technique is very useful in contingency analysis where numerous outages are to be simulated or a power flow solution is required for on-line control.[1]

For large scale power system, usually the transmission lines have a very high X/R ratio. For such a system, real power changes  $\Delta P$  are less sensitive to changes in voltage magnitude and are most sensitive to changes in phase angle  $\Delta\delta$ . Similarly, reactive power is less sensitive to changes in angle and most sensitive on changes in voltage magnitude. Incorporate of these approximations into the Jacobian matrix in Newton-Raphson power flow solution makes the elements of the submatrices  $J_{12}$  and  $J_{21}$  zero.



We are then left with two separated systems of equations,

$$\begin{bmatrix} \frac{\partial P_2}{\partial \delta_2} & \dots & \frac{\partial P_2}{\partial \delta_n} \\ \vdots & J_{11} & \vdots \\ \frac{\partial P_n}{\partial \delta_2} & \dots & \frac{\partial P_n}{\partial \delta_n} \end{bmatrix} \begin{bmatrix} \Delta \delta_2 \\ \vdots \\ \Delta \delta_n \end{bmatrix} = \begin{bmatrix} \Delta P_2 \\ \vdots \\ \Delta P_n \end{bmatrix} \quad (14)$$

$$\begin{bmatrix} |V_2| \frac{\partial Q_2}{\partial |V_2|} & \dots & |V_n| \frac{\partial Q_2}{\partial |V_n|} \\ \vdots & J_{22} & \vdots \\ |V_2| \frac{\partial Q_n}{\partial |V_2|} & \dots & |V_n| \frac{\partial Q_n}{\partial |V_n|} \end{bmatrix} \begin{bmatrix} \frac{\Delta |V_2|}{|V_2|} \\ \vdots \\ \frac{\Delta |V_n|}{|V_n|} \end{bmatrix} = \begin{bmatrix} \Delta Q_2 \\ \vdots \\ \Delta Q_n \end{bmatrix} \quad (15)$$

In well-designed and properly operated power transmission system:

- i) Angular differences between typical buses of the system are usually so small.

$\delta_{ij} = (\delta_i - \delta_j)$  very small that results,

$$\cos \delta_{ij} \approx 1$$

$$\sin \delta_{ij} \approx 0.0$$

- ii) The line susceptances  $B_{ij}$  are many times larger than the line conductances  $G_{ij}$  so that  $G_{ij} \sin \delta_{ij} \ll B_{ij}$ .

- iii) The reactive power  $Q_i$  injected into any bus  $i$  of the system during normal operation is much less than the reactive power which would flow if all lines from that bus were short circuited to reference.

That is  $Q_i \ll |V_i|^2 B_{ii}$ .

$$\frac{\partial P_i}{\partial \delta_j} = -|Y_{ij} V_i V_j| \sin(\theta_{ij} + \delta_j - \delta_i) \quad (16)$$

$$|V_j| \frac{\partial Q_i}{\partial V_j} = -|V_j| |Y_{ij} V_i| \sin(\theta_{ij} + \delta_j - \delta_i) = \frac{\partial P_i}{\partial \delta_j} \quad (17)$$

In Eq.(16) and Eq.(17), the off diagonal elements of  $J_{11}$  and  $J_{22}$  are given by

$$|V_j| \frac{\partial Q_i}{\partial V_j} = -|V_j| |Y_{ij} V_i| \sin(\theta_{ij} + \delta_j - \delta_i) = \frac{\partial P_i}{\partial \delta_j} \quad (18)$$

Using the identity  $\sin(\alpha + \beta) = \sin \alpha \cos \beta + \cos \alpha \sin \beta$  in Eq.(18) gives us

$$\frac{\partial P_i}{\partial \delta_j} = |V_j| \frac{\partial Q_i}{\partial |V_j|} = -|V_i V_j| B_{ij} \cos(\delta_j - \delta_i) + G_{ij} \sin(\delta_j - \delta_i) \quad (19)$$

The approximation listed above then yield the off diagonal elements

$$\frac{\partial P_i}{\partial \delta_j} = |V_j| \frac{\partial Q_i}{\partial |V_j|} = -|V_i V_j| B_{ij} \quad (20)$$

$$\frac{\partial P_i}{\partial \delta_i} = \sum_{\substack{j=1 \\ j \neq i}}^n |Y_{ij} V_i V_j| \sin(\theta_{ij} + \delta_j - \delta_i) \quad (21)$$

$$|V_i| \frac{\partial Q_i}{\partial V_i} = -\frac{\partial P_i}{\partial \delta_i} - 2|V_i|^2 B_{ii} = Q_i - |V_i|^2 B_{ii} \quad (22)$$

The diagonal elements of  $J_{11}$  and  $J_{22}$  are shown in Eq. (8) and Eq. (9) respectively.

Applying the inequality  $Q_i \ll |V_i|^2 B_{ii}$  to those expressions yields

$$\frac{\partial P_i}{\partial \delta_i} \cong |V_i| \frac{\partial Q_i}{\partial \delta_i} \cong -|V_i|^2 B_{ii} \quad (23)$$

Substitute Eq. (19) and Eq. (20) into Eq. (14) and Eq. (15), we obtain

$$\begin{bmatrix} -|V_2 V_2| B_{22} & -|V_2 V_3| B_{23} & \cdots & -|V_2 V_n| B_{2n} \\ -|V_2 V_3| B_{32} & -|V_3 V_3| B_{33} & \cdots & -|V_3 V_n| B_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ -|V_2 V_n| B_{n2} & -|V_3 V_n| B_{n3} & \cdots & -|V_n V_n| B_{nn} \end{bmatrix} \begin{bmatrix} \Delta \delta_2 \\ \Delta \delta_3 \\ \vdots \\ \Delta \delta_n \end{bmatrix} = \begin{bmatrix} \Delta P_2 \\ \Delta P_3 \\ \vdots \\ \Delta P_n \end{bmatrix} \quad (24)$$

$$\begin{bmatrix} -|V_2 V_2| B_{22} & -|V_2 V_3| B_{23} & \cdots & -|V_2 V_n| B_{2n} \\ -|V_2 V_3| B_{32} & -|V_3 V_3| B_{33} & \cdots & -|V_3 V_n| B_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ -|V_2 V_n| B_{n2} & -|V_3 V_n| B_{n3} & \cdots & -|V_n V_n| B_{nn} \end{bmatrix} \begin{bmatrix} \frac{\Delta|V_2|}{|V_2|} \\ \frac{\Delta|V_3|}{|V_3|} \\ \vdots \\ \frac{\Delta|V_n|}{|V_n|} \end{bmatrix} = \begin{bmatrix} \Delta Q_2 \\ \Delta Q_3 \\ \vdots \\ \Delta Q_n \end{bmatrix} \quad (25)$$

We can also modify Eq. (24) and Eq. (25) to two decoupled systems of equations for n-bus network.

$$\underbrace{\begin{bmatrix} -B_{22} & -B_{23} & \cdots & -B_{2n} \\ -B_{32} & -B_{33} & \cdots & -B_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ -B_{n2} & -B_{n3} & \cdots & -B_{nn} \end{bmatrix}}_{B'} \begin{bmatrix} \Delta\delta_2 \\ \Delta\delta_3 \\ \vdots \\ \Delta\delta_n \end{bmatrix} = \begin{bmatrix} \frac{\Delta P_2}{|V_2|} \\ \frac{\Delta P_3}{|V_3|} \\ \vdots \\ \frac{\Delta P_n}{|V_n|} \end{bmatrix} \quad (26)$$

or

$$\begin{bmatrix} \Delta\delta_2 \\ \Delta\delta_3 \\ \vdots \\ \Delta\delta_n \end{bmatrix} = -[B']^{-1} \begin{bmatrix} \frac{\Delta P_2}{|V_2|} \\ \frac{\Delta P_3}{|V_3|} \\ \vdots \\ \frac{\Delta P_n}{|V_n|} \end{bmatrix} \quad (27)$$

And

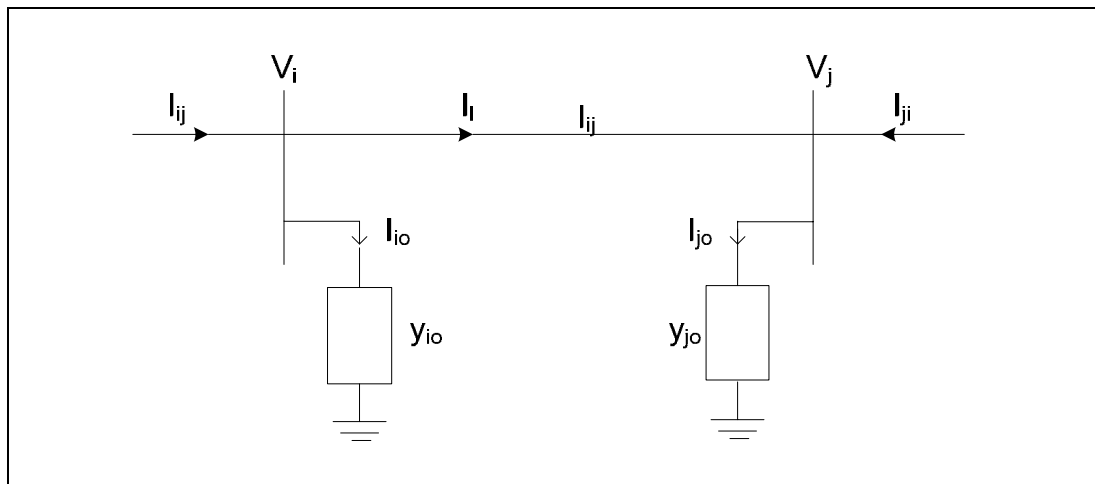
$$\underbrace{\begin{bmatrix} -B_{22} & -B_{23} & \cdots & -B_{2n} \\ -B_{32} & -B_{33} & \cdots & -B_{34} \\ \vdots & \vdots & \vdots & \vdots \\ -B_{n2} & -B_{n3} & \cdots & -B_{nn} \end{bmatrix}}_{B'} \begin{bmatrix} \Delta|V_2| \\ \Delta|V_3| \\ \vdots \\ \Delta|V_n| \end{bmatrix} = \begin{bmatrix} \frac{\Delta Q_2}{|V_2|} \\ \frac{\Delta Q_3}{|V_3|} \\ \vdots \\ \frac{\Delta Q_n}{|V_n|} \end{bmatrix} \quad (28)$$

or

$$\begin{bmatrix} \Delta|V_2| \\ \Delta|V_3| \\ \vdots \\ \Delta|V_n| \end{bmatrix} = -[B']^{-1} \begin{bmatrix} \frac{\Delta Q_2}{|V_2|} \\ \frac{\Delta Q_3}{|V_3|} \\ \vdots \\ \frac{\Delta Q_n}{|V_n|} \end{bmatrix}$$

$B_{ij}$  are the imaginary parts of the corresponding  $Y_{\text{bus}}$  elements.

### 2.3 Line Flow and Losses Calculation



**Figure 2.1:** Transmission line model for calculating line flows

$i \rightarrow j$  is given by

$$I_{ij} = I_l + I_{io} = y_{ij} (V_i - V_j) + y_{io} V_i \quad (29)$$

Similarly, the line current  $I_{ij}$  measured at bus  $j$  and defined positive in the direction  $j \rightarrow i$  is given by

$$I_{ij} = -I_l + I_{io} = y_{ij} (V_j - V_i) + y_{jo} V_j \quad (30)$$

The complex power  $S_{ij}$  from bus  $i$  to  $j$  and  $S_{ji}$  from bus  $j$  to  $i$  are

$$S_{ij} = V_i I_{ij}^* \quad (31)$$

$$S_{ji} = V_j I_{ji}^* \quad (32)$$

The power loss in line  $i - j$  is the algebraic sum of the power flows determined from (6.40) and (6.41), i.e.,

$$S_{Lij} = S_{ij} + S_{ji} \quad (33)$$

## 2.4 MATLAB GUI

A graphical user interface (GUI) is a pictorial interface to a program. A good GUI can make programs easier to use by providing them with a consistent appearance and with intuitive controls like pushbuttons, list boxes, sliders, menus, and so forth.[6][7][8]. The GUI should behave in an understandable and predictable manner, so that a user knows what to expect when he or she performs an action. For example, when a mouse click occurs on pushbutton, the GUI should initiate the action described on the label of the button. This chapter introduces the basic elements of the MATLAB GUIs [6] [7] [8]. The chapter does not contain a complete description of components or GUI features, but it does provide the basics required to create functional GUIs for your programs [6].

Applications that provide GUIs are generally easier to learn and use since the person using the application does not need to know what commands are available or how they work [7] [8]. The action that results from a particular user action can be made clear by the design of the interface [6] [7] [8].

A graphical user interface provides the user with a familiar environment in which to work. This environment contains pushbuttons, toggle buttons, lists, menus, text boxes, and so forth [5] [6] [7] [8]. All of which are already familiar to the user, so that he or she can concentrate on using the application rather than on the mechanics involved in doing things. However, GUIs are harder for the programmer because a GUI-based program must be prepared for mouse clicks (or possibly keyboard input) for any GUI element at any time [5] [6] [7].

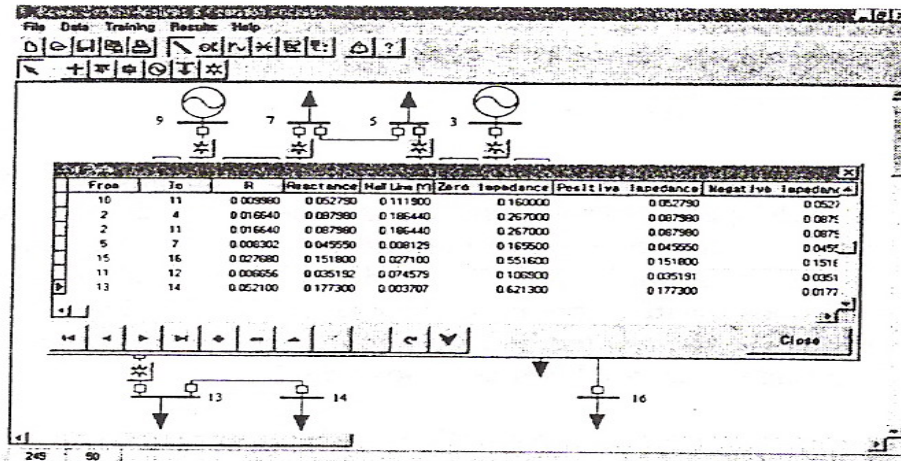
Such inputs are known as events, and a program that responds to events is said to be *event driven*. The three principal elements required to create a MATLAB Graphical User Interfaces are [6]:-

- i. Components. Each item on a MATLAB GUI (pushbuttons, labels, edit boxes, etc.) is a graphical component. The types of components include graphical controls (pushbuttons, edit boxes, lists, sliders, etc.), static elements (frames and text strings), menus, and axes. Graphical controls and static elements are created by the function `uicontrol`, and menus are created by the functions `uimenu` and `uicontextmenu`. Axes, which are used to display graphical data, are created by the function `axes` [5] [6] [7] [8].
- ii. Figures. The components of a GUI must be arranged within a figure, which is a window on the computer screen. In the past, figures have been created automatically whenever we have plotted data. However, empty figures can be created with the function `figure` and can be used to hold any combination of components [6].

- iii. Callbacks. Finally, there must be some way to perform an action if a user clicks a mouse on a button or types information on a keyboard. A mouse click or a key press is an event, and the MATLAB program must respond to each event if the program is to perform its function. For example, if a user clicks on a button, that event must cause the MATLAB code that implements the function of the button to be executed. The code executed in response to an event is known as a call back. There must be a callback to implement the function of each graphical component on the GUI [6] [7].

## **2.5 Similar software in Market**

A Windows-based Interactive and Graphic Package - For the Education and Training of Power System Analysis and operation. The authors of this software package are Joong-Rin Shin, Wook-Hwa Lee and Dong-hae Im from Department of Electrical Engineering (Kon-Kuk University, Seoul). The software represents a Windows-based Interactive Graphic Package developed for the education and training of the power system modeling, analysis and operation with user-friendly Graphical User Interface (GUI) and Visual Data Base Management System (VDBMS). Both GUI and VDBMS have been developed by the use of Borland C++ Builder and the application programs are written in C++. The application programs in this package include the Power Flow (PF) calculation, the Transient Stability Analysis (TSA), the Fault Analysis (FA), the Economic Dispatch (ED), and the Automatic Load-Frequency Control (ALFC) [2].



**Fig. 2.2:** A Windows-based Interactive and Graphic Package.

Figure 2 above shows screen shot for power flow calculation. The programming of power flow solution, the authors of this software choose to use full Jacobian Newton-Raphson method the most wide used method.



## **CHAPTER 3**

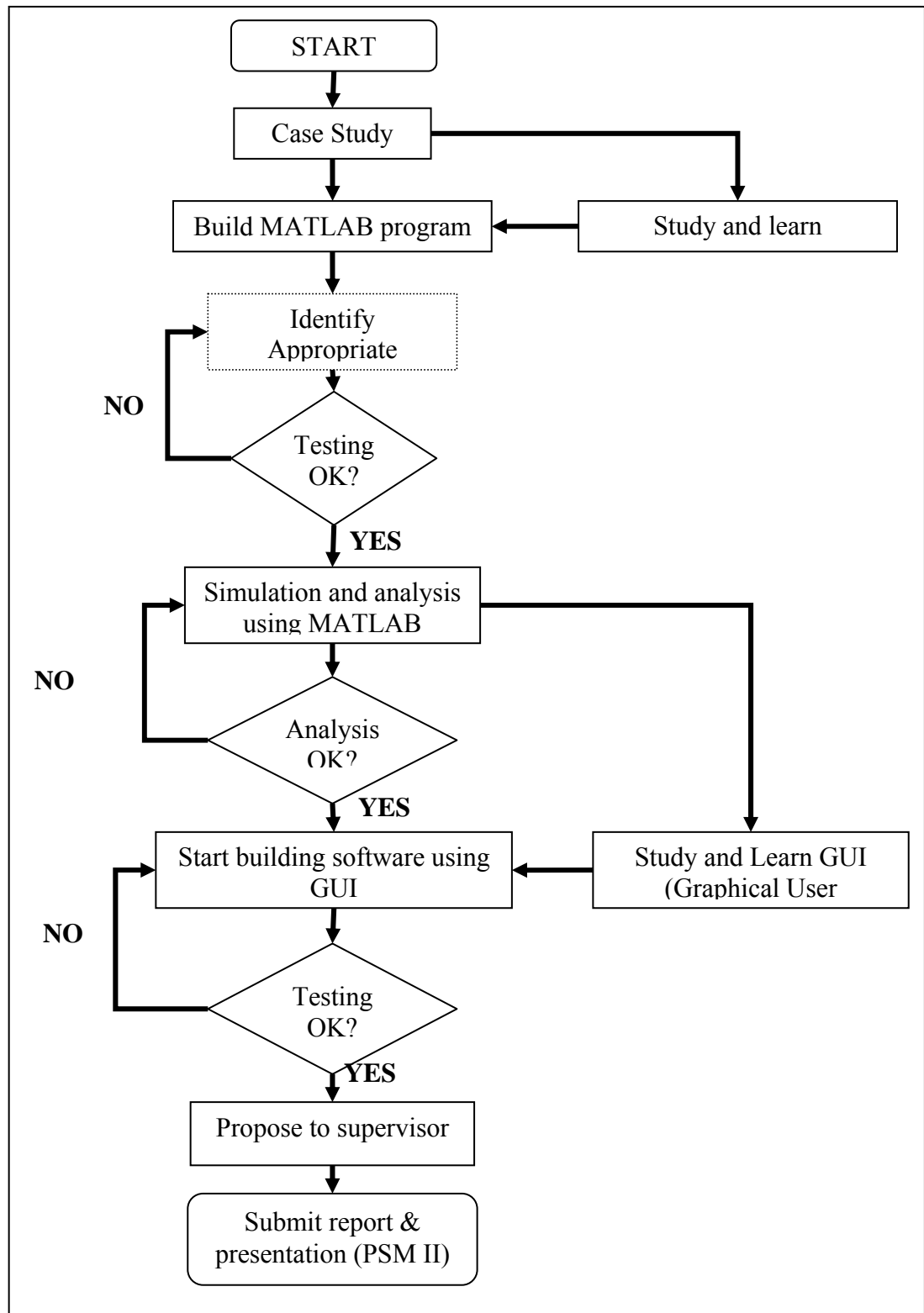
### **METHODOLOGY**

#### **3.1 Introduction**

This chapter presents the methodology of this project. It describes on how the project is organized and the flow of the steps in order to complete this project. The methodology is diverged in two parts, which is developing the programming for MATLAB and build graphical user interface in MATLAB GUIDE.

There are three mains method in order to develop this project. Before the project is developing using MATLAB, it is needed to do the study on MATLAB GUIDE and MATLAB program. The flowchart in Figure 3.1 illustrated the sequence of steps for this project. The first method is build MATLAB program of power flow solving. Secondly is run simulation and analysis using MATLAB. Lastly is developing GUI in MATLAB and programs every GUI component.

### 3.2 Project Flow Chart



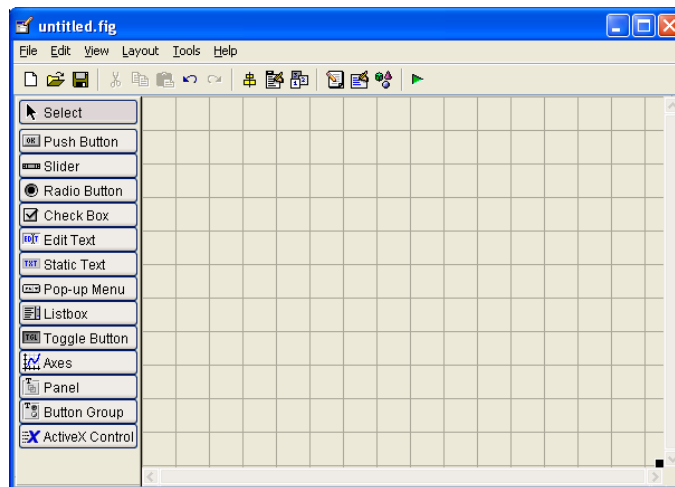
**Figure 3.1:** Flowchart for Whole Project

Figure 3.1 show the flow of the whole project. The project begins with doing case study about the project. First, power flow analysis programming develops using MATLAB. The second part is to run simulation of developed program in MATLAB for small, medium and large systems. After that design graphical for the software using MATLAB GUI and interface with the developed power flow analysis program. The main contribution of this project is to interface MATLAB GUI with the power flow analysis program.

### 3.3 Developing MATLAB GUI using MATLAB GUIDE

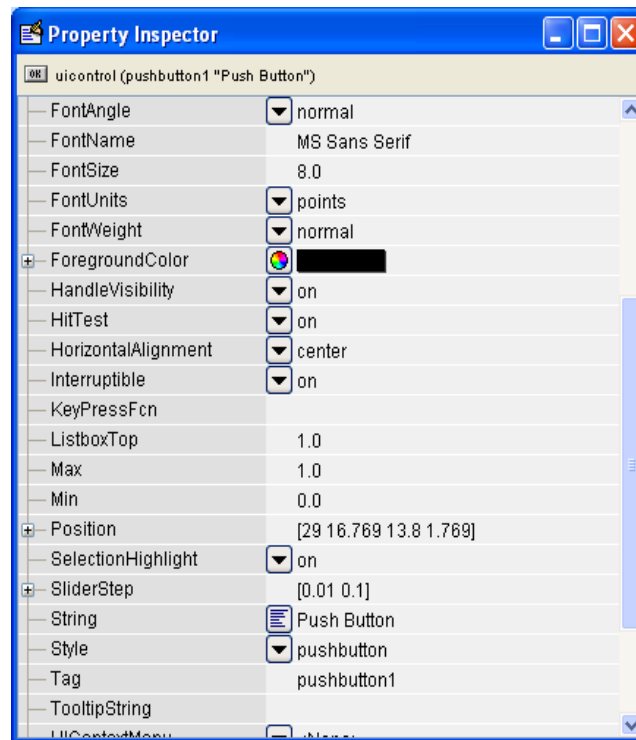
GUIDE, the MATLAB graphical user interface development environment, provides a set of tools for creating graphical user interfaces (GUIs). These tools simplify the process of laying out and programming GUIs. This tool allows a programmer to layout the GUI, selecting and aligning the GUI components to be placed in.

There are 5 steps in build the MATLAB GUI. Firstly the GUI's layout is used by placing the components in Layout Editor in MATLAB GUIDE. Figure 3.2 show MATLAB GUIDE Layout Editor. The basic component of the MATLAB GUI is shown in Table 3.1.



**Figure 3.2:** MATLAB GUIDE Layout Editor

Secondly, the MATLAB tool called the Property Inspector (built into guide) is used to give each component a name (a "tag") and to set the characteristics of each component, such as its color, the text it displays, and so on. Then, the figure is saved. When the figure is saved, two files will be created on disk with the same name but different extents. The fig file contains the actual GUI that has been created, and the M-file contains the code to load the figure and skeleton call backs for each GUI element.

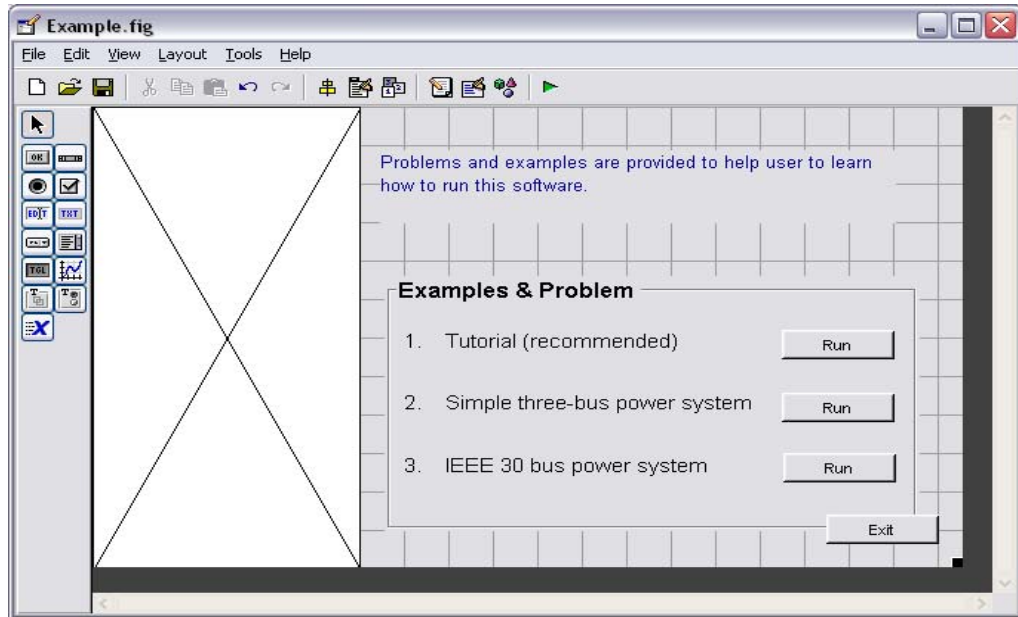


**Figure 3.3:** Property Inspector

**Table 3.1:** Basic MATLAB GUI Component [7]

Element	Created By	Description
<b>Graphical Controls</b>		
Pushbutton	<code>uicontrol</code>	A graphical component that implements a pushbutton. It triggers a callback when clicked with a mouse.
Toggle button	<code>uicontrol</code>	A graphical component that implements a toggle button. A toggle button is either “on” or “off,” and it changes state each time that it is clicked. Each mouse button click also triggers a callback.
Radio button	<code>uicontrol</code>	A radio button is a type of toggle button that appears as a small circle with a dot in the middle when it is “on.” Groups of radio buttons are used to implement mutually exclusive choices. Each mouse click on a radio button triggers a callback.
Check box	<code>uicontrol</code>	A check box is a type of toggle button that appears as a small square with a check mark in it when it is “on.” Each mouse click on a check box triggers a callback.
Edit box	<code>uicontrol</code>	An edit box displays a text string and allows the user to modify the information displayed. A callback is triggered when the user presses the <code>Enter</code> key.
List box	<code>uicontrol</code>	A list box is a graphical control that displays a series of text strings. A user can select one of the text strings by single- or double-clicking on it. A callback is triggered when the user selects a string.
Popup menus	<code>uicontrol</code>	A popup menu is a graphical control that displays a series of text strings in response to a mouse click. When the popup menu is not clicked on, only the currently selected string is visible.
Slider	<code>uicontrol</code>	A slider is a graphical control to adjust a value in a smooth, continuous fashion by dragging the control with a mouse. Each slider change triggers a callback.
<b>Static Elements</b>		
Frame	<code>uicontrol</code>	Creates a frame, which is a rectangular box within a figure. Frames are used to group sets of controls together. Frames never trigger callbacks.
Text field	<code>uicontrol</code>	Creates a label, which is a text string located at a point on the figure. Text fields never trigger callbacks.
<b>Menus and Axes</b>		
Menu items	<code>uimenu</code>	Creates a menu item. Menu items trigger a callback when a mouse button is released over them.
Context menus	<code>uicontextmenu</code>	Creates a context menu, which is a menu that appears over a graphical object when a user right-clicks the mouse on that object.
Axes	<code>axes</code>	Creates a new set of axes to display data on. Axes never trigger callbacks.

After laying out the GUI component and set the property, Figure 3.4 shows the GUI layout after laying out the GUI component and set the property.



**Figure 3.4:** Example of GUI

Finally, the code is programmed to implement the behavior associated with each callback function in m-files as shown in figure 3.5. This last step is the most difficult part and has to make an extra reading on how to write the coding before the GUI component can perform some task.

```

Editor - C:\MATLAB7\work\Powersolve\Database\Example.m
File Edit Text Cell Tools Debug Desktop Window Help
Stack: Base
% Executes just before Example is made visible.
50 function Example_OpeningFcn(hObject, eventdata, handles, varargin)
51 % This function has no output args, see OutputFcn.
52 % hObject handle to figure
53 % eventdata reserved - to be defined in a future version of MATLAB
54 % handles structure with handles and user data (see GUIDATA)
55 % varargin command line arguments to Example (see VARARGIN)
56 movegui('center')
57 % Choose default command line output for Example
58 handles.output = hObject;
59
60 % Update handles structure
61 guidata(hObject, handles);
62
63 % UIWAIT makes Example wait for user response (see UIRESUME)
64 % uiwait(handles.figure1);
65
66
67 % --- Outputs from this function are returned to the command line.
68 function varargout = Example_OutputFcn(hObject, eventdata, handles)
69 % varargout cell array for returning output args (see VARARGOUT);
70 % hObject handle to figure
71 % eventdata reserved - to be defined in a future version of MATLAB
72 % handles structure with handles and user data (see GUIDATA)
73
74 % Get default command line output from handles structure
75 varargout(1) = handles.output;
76
77
78 % --- Executes during object creation, after setting all properties.

```

Figure 3.5: Example of M-files for GUI

### 3.4 Problems on Power System

#### 3.4.1 Example of 3-Bus System

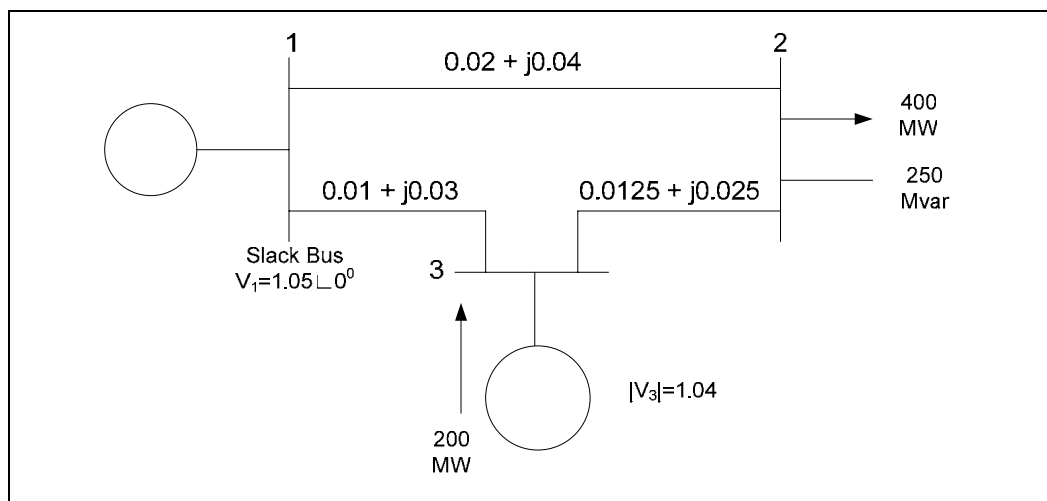


Figure 3.6: Example of 3 bus system

MVA base of system = 100MVA

**Table 3.2:** Bus data of 3-bus system

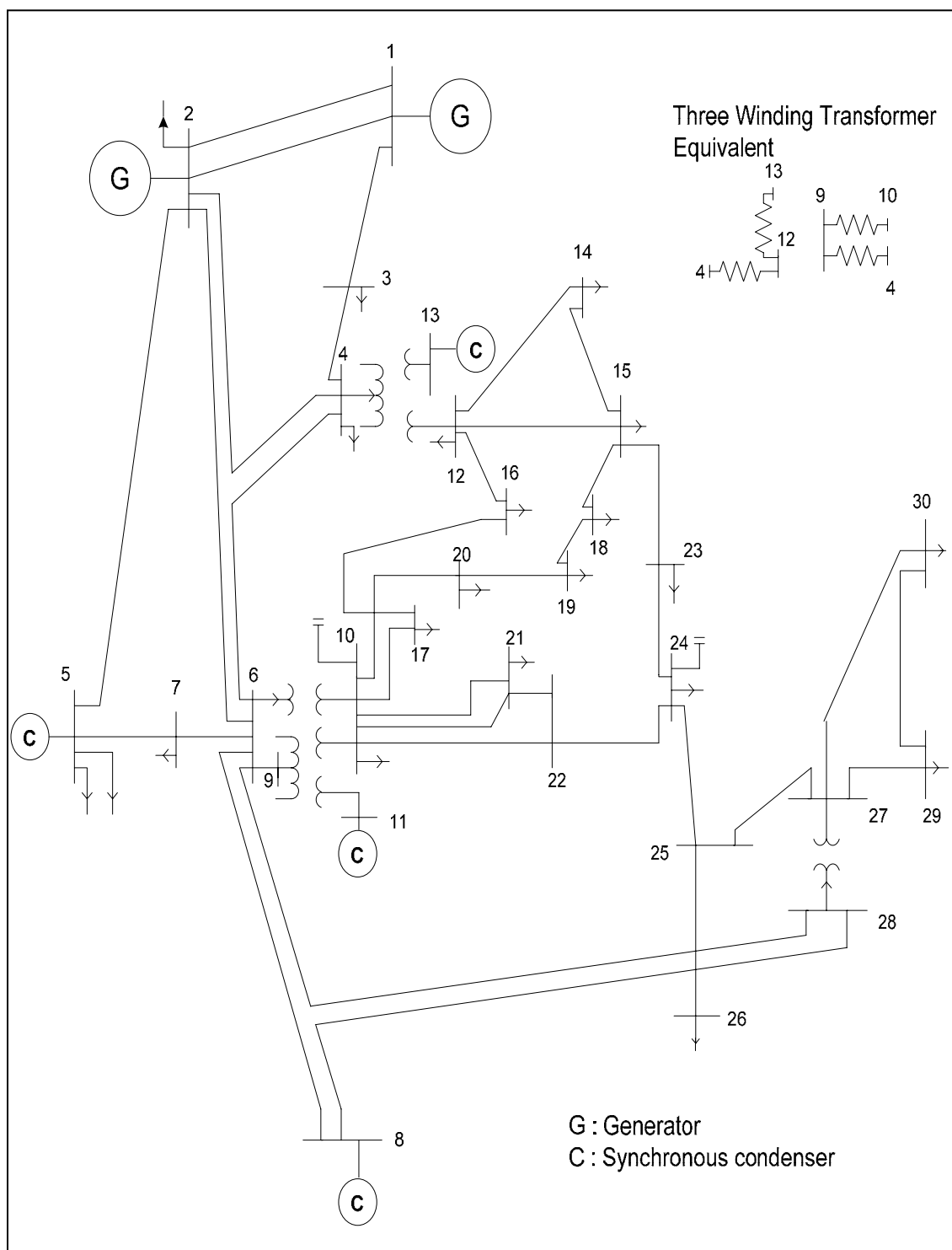
Bus No.	Bus Type	Voltage Magnitude	Angle Degree	load		Generator				Injected Mvar
				MW	Mvar	MW	Mvar	Qmin	Qmax	
1	Reference	1.05	0	0	0	0	0	0	0	0
2	Load	1	0	400	250	0	0	0	0	0
3	Generation	1.04	0	2.4	1.2	200	0	0	0	0

**Table 3.3:** Line data of 3-bus system

Bus From	Bus To	Resistance p.u	Reactance p.u	1/2 B p.u	Transformer Tap Setting Value
1	2	0.02	0.04	0	1
1	3	0.01	0.3	0	1
2	3	0.0125	0.025	0	1



### 3.4.2 Example of IEEE 30-Bus System



**Figure 3.7:** IEEE 30 bus system

MVA base of system = 100MVA

**Table 3.4:** Regulated Bus Data

<b>Regulated Bus Data</b>			
Bus No.	Voltage Magnitude	Min. Mvar Capacity	Max. Mvar Capacity
2	1.043	-40	50
5	1.01	-40	40
8	1.01	-10	40
11	1.082	-6	24
13	1.071	-6	24

**Table 3.5:** Transformer Data

<b>Transformer Data</b>	
Transformer Designation	Tap Setting Pu
4 – 12	0.932
6 - 9	0.978
6 – 10	0.969
28 – 27	0.968

**Table 3.6:** Q injection Data

<b>Injected Q due to Capacitors</b>	
Bus No.	Mvar
10	19
24	4.3

**Table 3.7:** Bus data for IEEE 30 bus system

Bus No.	Bus Type	Voltage Magnitude	Angle Degree	Load		Generator				Injected Mvar
				MW	Mvar	MW	Mvar	Qmin	Qmax	
1	Reference	1.060	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	Generation	1.043	0	21.70	12.70	40.00	0.00	-40.00	50.00	0.00
3	Load	1.000	0	2.40	1.20	0.00	0.00	0.00	0.00	0.00
4	Load	1.060	0	7.60	1.60	0.00	0.00	0.00	0.00	0.00
5	Generation	1.010	0	94.20	19.00	0.00	0.00	-40.00	40.00	0.00
6	Load	1.000	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	Load	1.000	0	22.80	10.90	0.00	0.00	0.00	0.00	0.00
8	Generation	1.010	0	30.00	30.00	0.00	0.00	-10.00	60.00	0.00
9	Load	1.000	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	Load	1.000	0	5.80	2.00	0.00	0.00	-6.00	24.00	19.00
11	Generation	1.082	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
12	Load	1.000	0	11.20	7.50	0.00	0.00	0.00	0.00	0.00
13	Generation	1.071	0	0.00	0.00	0.00	0.00	-6.00	24.00	0.00
14	Load	1.000	0	6.20	1.60	0.00	0.00	0.00	0.00	0.00
15	Load	1.000	0	8.20	2.50	0.00	0.00	0.00	0.00	0.00
16	Load	1.000	0	3.50	1.80	0.00	0.00	0.00	0.00	0.00
17	Load	1.000	0	9.00	5.80	0.00	0.00	0.00	0.00	0.00
18	Load	1.000	0	3.20	0.90	0.00	0.00	0.00	0.00	0.00
19	Load	1.000	0	9.50	3.40	0.00	0.00	0.00	0.00	0.00
20	Load	1.000	0	2.20	0.70	0.00	0.00	0.00	0.00	0.00
21	Load	1.000	0	17.50	11.20	0.00	0.00	0.00	0.00	0.00
22	Load	1.000	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
23	Load	1.000	0	3.20	1.60	0.00	0.00	0.00	0.00	0.00
24	Load	1.000	0	8.70	6.70	0.00	0.00	0.00	0.00	4.30
25	Load	1.000	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
26	Load	1.000	0	3.50	2.30	0.00	0.00	0.00	0.00	0.00
27	Load	1.000	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
28	Load	1.000	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
29	Load	1.000	0	2.40	0.90	0.00	0.00	0.00	0.00	0.00
30	Load	1.000	0	10.60	1.90	0.00	0.00	0.00	0.00	0.00

**Table 3.8:** Line data for IEEE 30 bus system

Bus From	Bus To	Resistance p.u	Reactance p.u	1/2 total charging p.u	Transformer Tap setting value
1	2	0.0920	0.0575	0.0640	1.0000
1	3	0.0452	0.1852	0.0204	1.0000
2	4	0.0570	0.1737	0.0184	1.0000
3	4	0.0132	0.0379	0.0042	1.0000
2	5	0.0472	0.1983	0.0209	1.0000
2	6	0.0581	0.1763	0.0187	1.0000
4	6	0.0119	0.0414	0.0045	1.0000
5	7	0.0460	0.1160	0.0102	1.0000
6	7	0.0267	0.0820	0.0085	1.0000
6	8	0.0120	0.0420	0.0045	1.0000
6	9	0.0000	0.2080	0.0000	0.9780
6	10	0.0000	0.5560	0.0000	0.9690
9	11	0.0000	0.2080	0.0000	1.0000
9	10	0.0000	0.1100	0.0000	1.0000
4	12	0.0000	0.2560	0.0000	0.9320
12	13	0.0000	0.1400	0.0000	1.0000
12	14	0.1231	0.2559	0.0000	1.0000
12	15	0.0662	0.1304	0.0000	1.0000
12	16	0.0945	0.1987	0.0000	1.0000
14	15	0.2210	0.1997	0.0000	1.0000
16	17	0.0824	0.1923	0.0000	1.0000
15	18	0.1073	0.2185	0.0000	1.0000
18	19	0.0639	0.1292	0.0000	1.0000
19	20	0.0340	0.0680	0.0000	1.0000
10	20	0.0936	0.2090	0.0000	1.0000
10	17	0.0324	0.0845	0.0000	1.0000
10	21	0.0348	0.0749	0.0000	1.0000
10	22	0.0727	0.1499	0.0000	1.0000
21	22	0.0116	0.0236	0.0000	1.0000
15	23	0.1000	0.2020	0.0000	1.0000
22	24	0.1150	0.1790	0.0000	1.0000
23	24	0.1320	0.2700	0.0000	1.0000
24	25	0.1885	0.3292	0.0000	1.0000
25	26	0.2544	0.3800	0.0000	1.0000
25	27	0.1093	0.2087	0.0000	1.0000
28	27	0.0000	0.3960	0.0000	0.9680
27	29	0.2198	0.4153	0.0000	1.0000
27	30	0.3202	0.6027	0.0000	1.0000
29	30	0.2399	0.4533	0.0000	1.0000
8	28	0.0636	0.2000	0.0214	1.0000
6	28	0.0169	0.0599	0.0650	1.0000

## **CHAPTER 4**

### **RESULTS AND DISCUSSIONS**

#### **4.1 Introduction**

This chapter consists three parts. First part will discuss on developed Power Flow Analysis software ver. 1.0. The second part will discuss on results, analysis and discussions of simulation on examples in Chapter 3 Section 3.4. This software design can be divided to four parts. The first part is main menu of the whole GUI. Second part is tutorial and examples for new user. The third part is run Power Flow Analysis Toolbox.

#### **4.2 Power Flow Analysis Toolbox Package ver. 1.0 (POWERSOLVE)**

##### **4.2.1 Introduction**

Power flow analysis software toolbox package ver. 1.0 (POWERSOLVE) is software to solve power flow analysis. MATLAB that have been use to develop this software is version 7.0. POWERSOLVE consists of three methods of power flow solution, Newton-Raphson method, Gauss-Seidel method and Fast Decoupled method.

POWERSOLVE have been saving in C:\MATLAB7\powersolve\database. Before that user need to copy and save powersolve folder in drive C (but make sure the folder is saved in MATLAB7 folder). Error will appear if it is saved in different drive. Powersolve folder consists of three subfolder; database, picture and formula. User need to 'add path' by click on 'path browser' as shown in Figure 4.1 or by type command 'addpath('c:\matlab\powersolve')' in MATLAB main frame. All folders must be put in MATLAB path list by clicking at 'Path' symbol and choose 'add to path'. Then user needs to choose 'File\save path'. If this step been ignored, user needs to repeat all above steps each time when user run powersolve software.

User can check all the contents in powersolve folder by type 'cd('d:\matlab7\powersolve')' followed by 'ls'. Same method can be use to check the contents of database and formula folders by type the command 'cd('d:\matlab7\powersolve\database')' followed by 'ls'. Type 'help cd' or 'help ls' in MATLAB main frame if user has difficulty with MATLAB command.

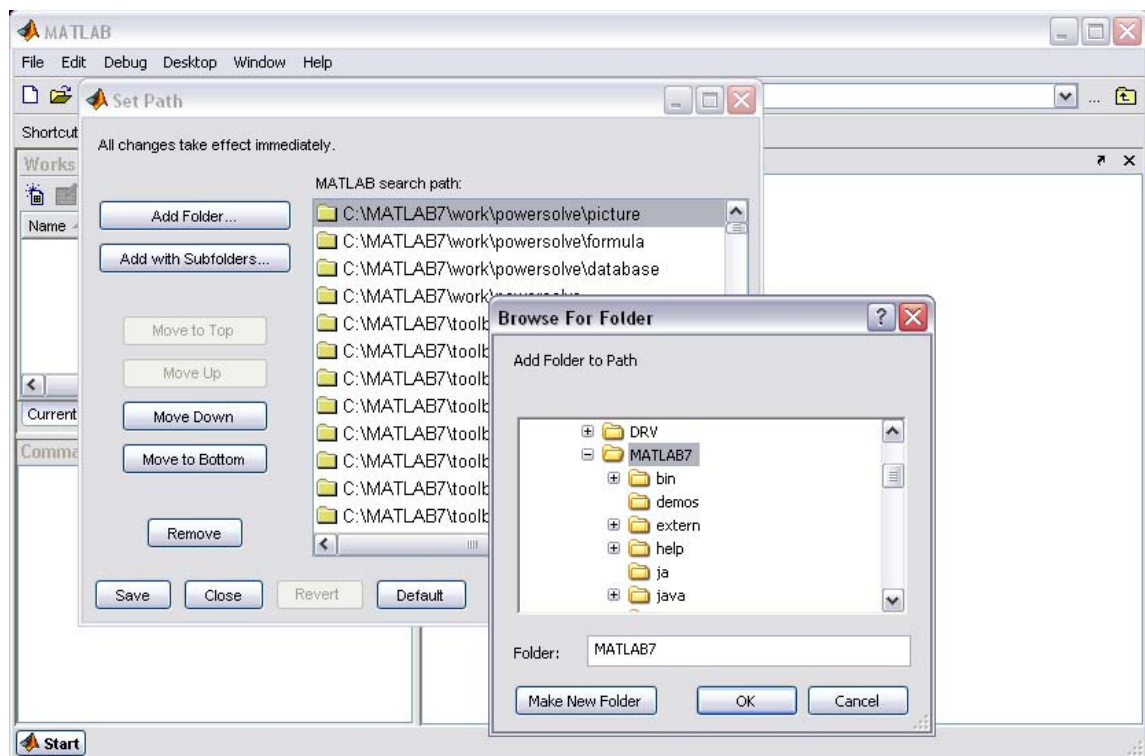
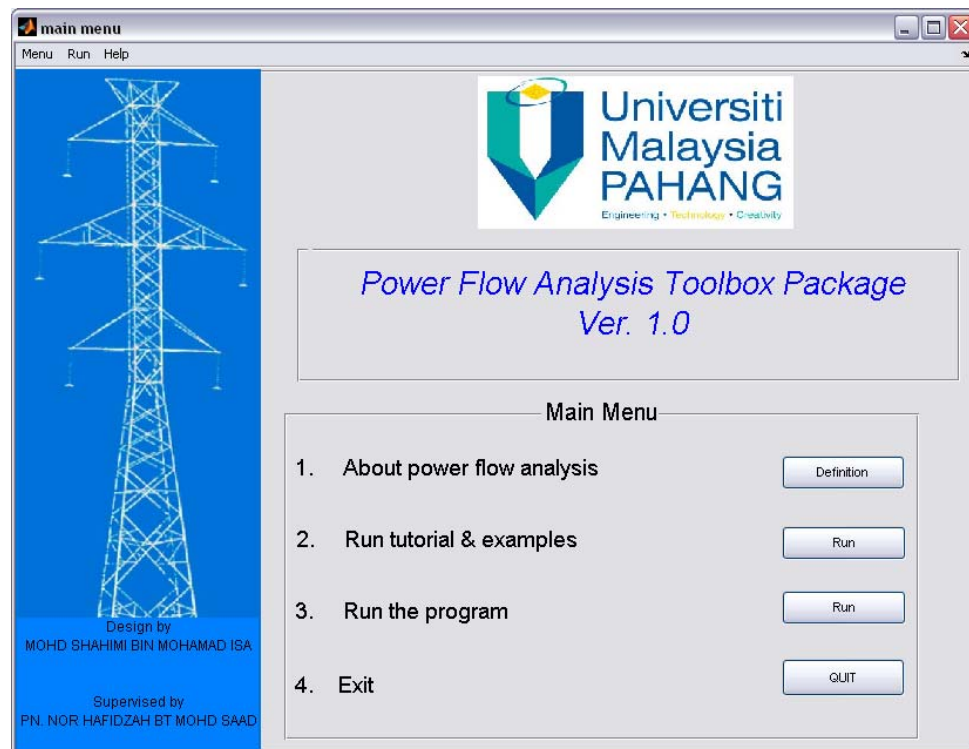


Figure 4.1: Addpath

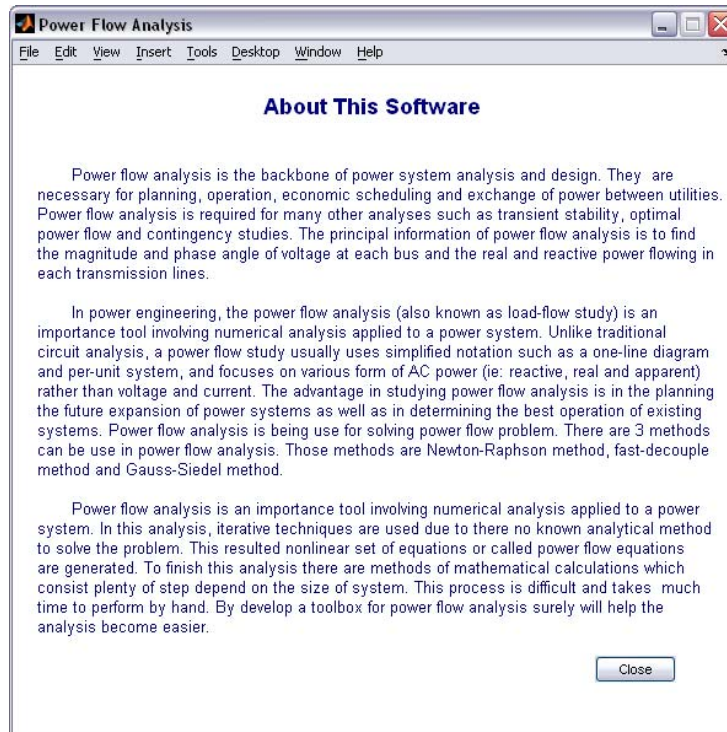
## 4.2.2 Regarding the Software

User may enter MATLAB main window by double click on MATLAB 7.0 icon. On MATLAB command window, type 'powersolve' (or capital letter POWERSOLVE), this step will run powersolve and main menu window will appear on screen as shown in Figure 4.2.

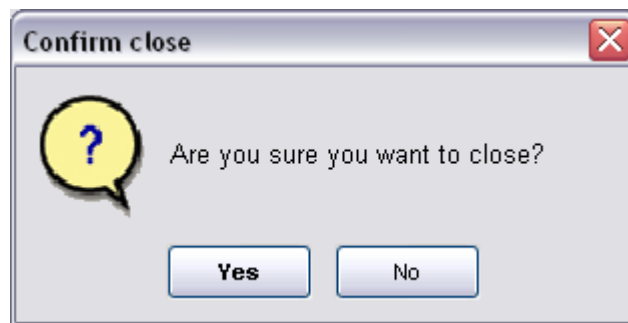
The main menu of the POWERSOLVE consist of four pushbuttons which link to general info about power flow analysis, run example and tutorial, run power flow analysis and lastly is exit button. The main menu of the GUI and info of the project is shown in Figure 4.2. Figure 4.3 shows the general information of Power Flow Analysis. It can be opened by choosing definition pushbutton in main menu. The detail of tutorial & examples and POWERSOLVE program will be discuss in next sub-chapter. For the exit button user will be asked about the confirmation either to exit the toolbox or not. The exit confirmation figure is shown in Figure 4.4.



**Figure 4.2:** Main menu of the GUI



**Figure 4.3:** General information of POWERSOLVE



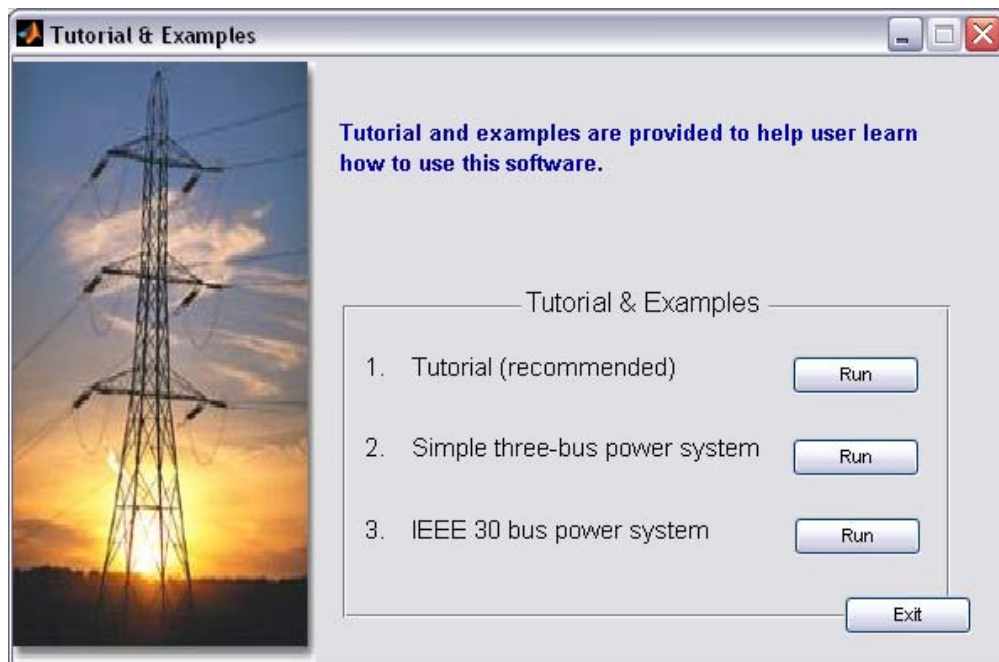
**Figure 4.4:** Exit Button Confirmation

#### 4.2.2.1 Tutorial & Examples

Tutorial and examples section is provided in order to help users to use Power Flow Analysis Toolbox version 1.0. Tutorial button is recommended because this



section will teach user how to use this software. More detail of tutorial will be discussed in next sub chapter. There are two examples provided to help users familiarize with the toolbox. First example is simple three bus power system. And for complex system this software provide user with IEEE 30 bus power system. Figure 4.4 shows the main menu of the tutorial and examples.



**Figure 4.5:** Tutorial & Examples menu

*a) Tutorial*

Tutorial section will teach user how to fill bus data and line data. Figure 4.6 shows the layout of the tutorial section. The bus data format and line data format contain the information data of each column. This will help user to prepare input data because it is important to obtain correct result. User need to click view button in order to view bus data format and line data format. The layout of both bus data format and line data format are shown in Figure 4.7 and Figure 4.8.

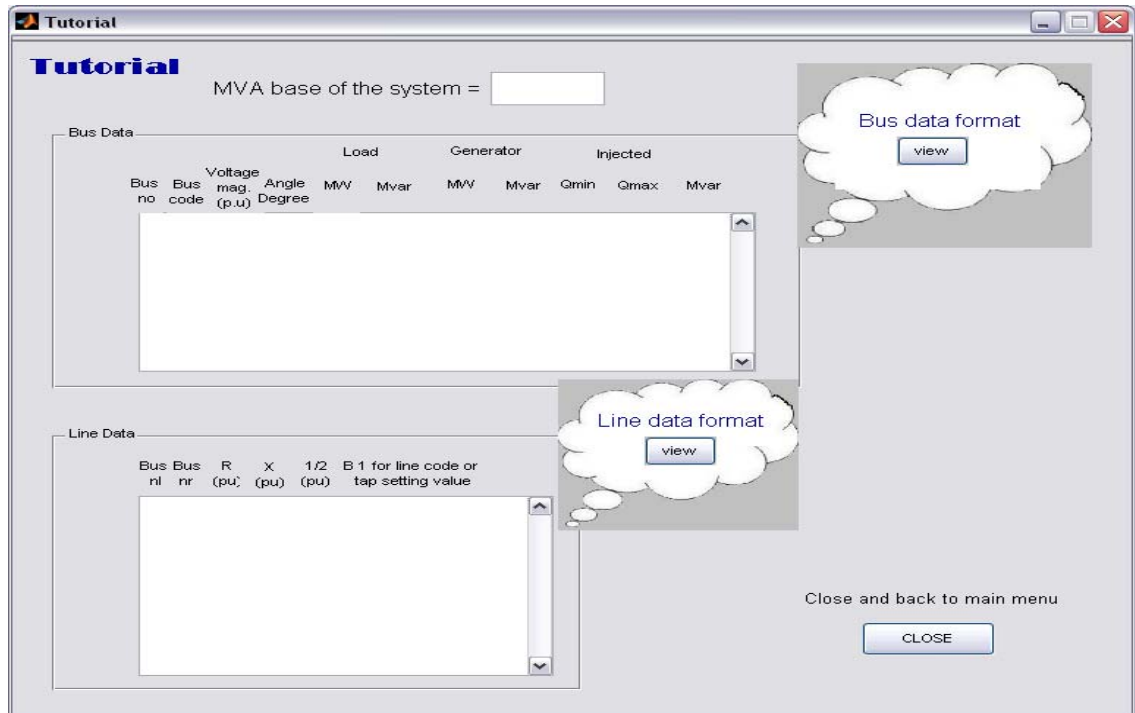


Figure 4.6: Tutorial

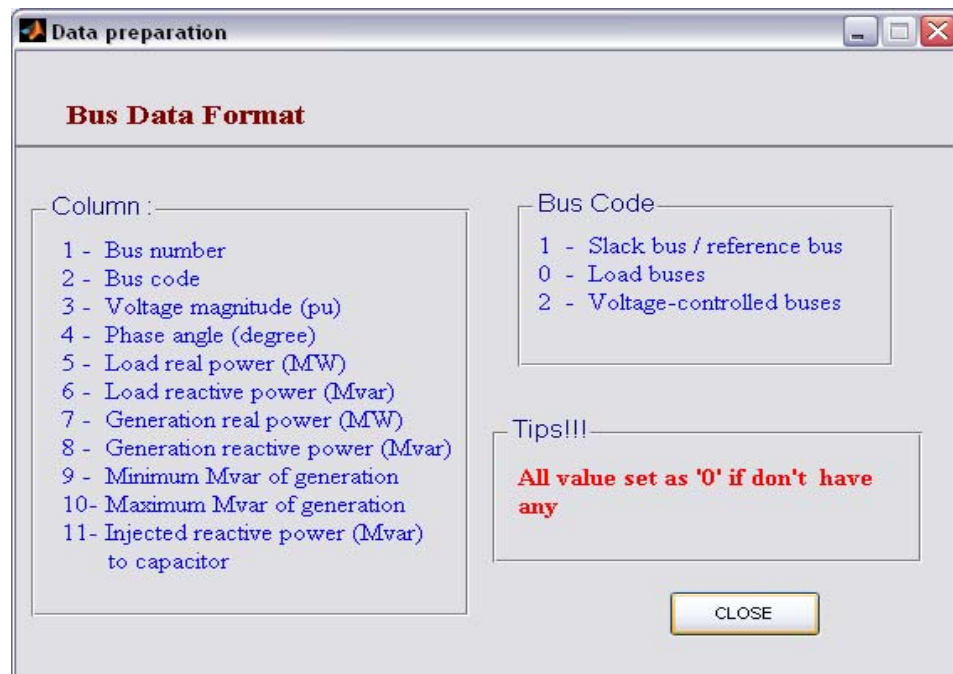
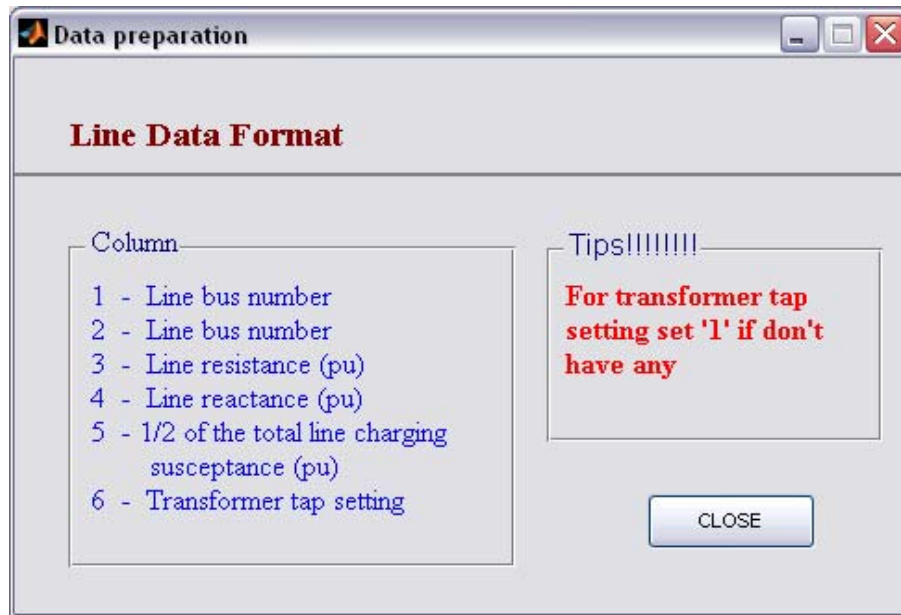


Figure 4.7: Bus data format



**Figure 4.8:** Line data format

*b) Examples*

Simple and complex power system provided in this section is to teach user the example of how to fill both bus data and line data. Both examples are display with one-line diagram to show the connection of the systems. Figure 4.9 to Figure 4.13 show the question, data preparation, analysis result, power flow analysis and line flow analysis of example 1 (simple 3 bus power system). Figure 4.14 to Figure 4.18 will show the question, data preparation, analysis result, power flow analysis and line flow analysis of example 2 (IEEE 30 bus power system).

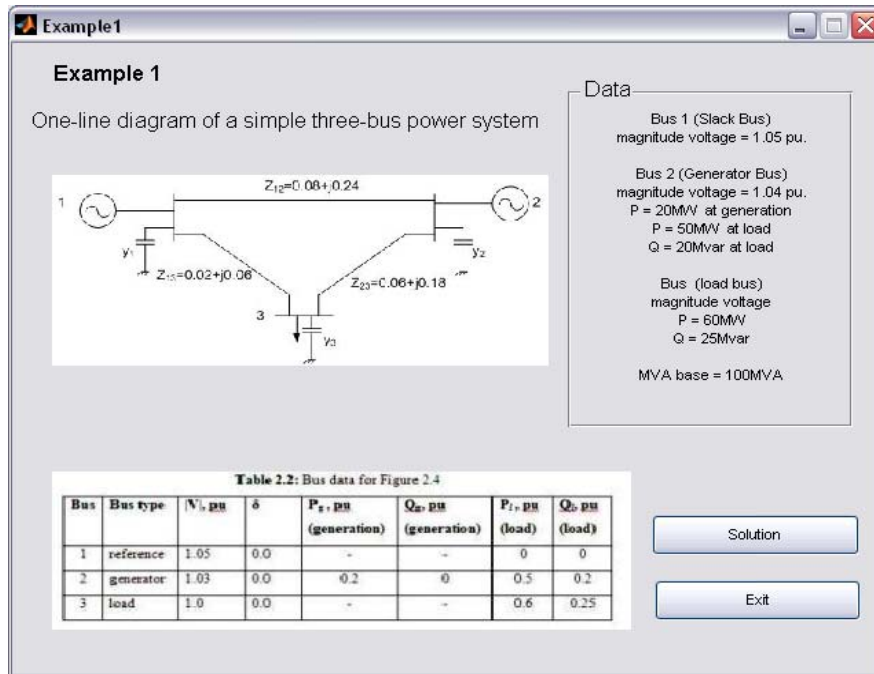


Figure 4.9: Example 1

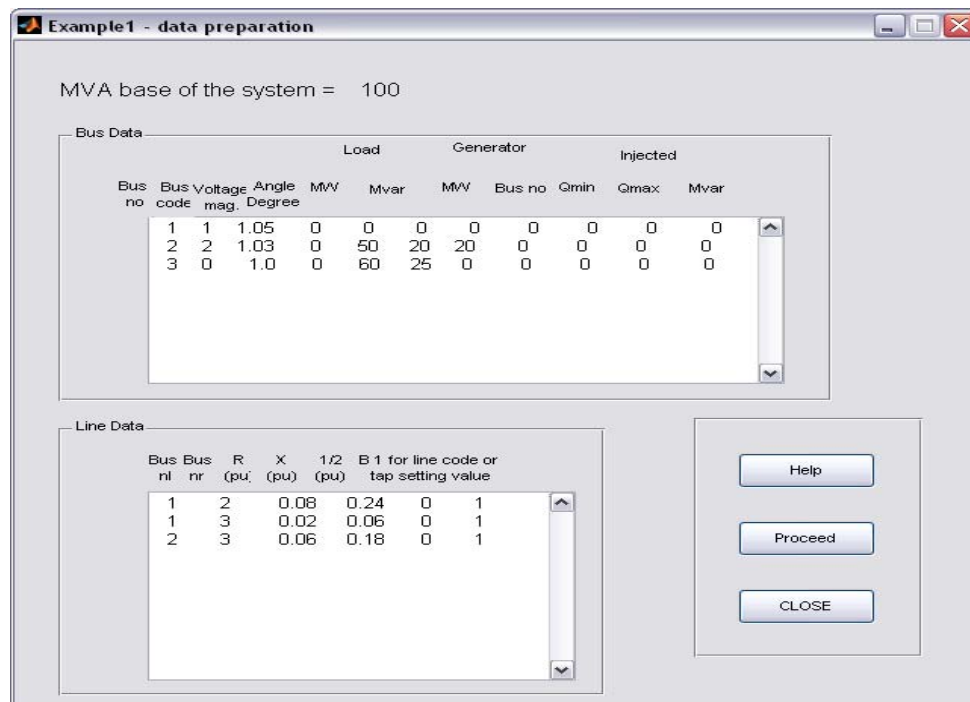
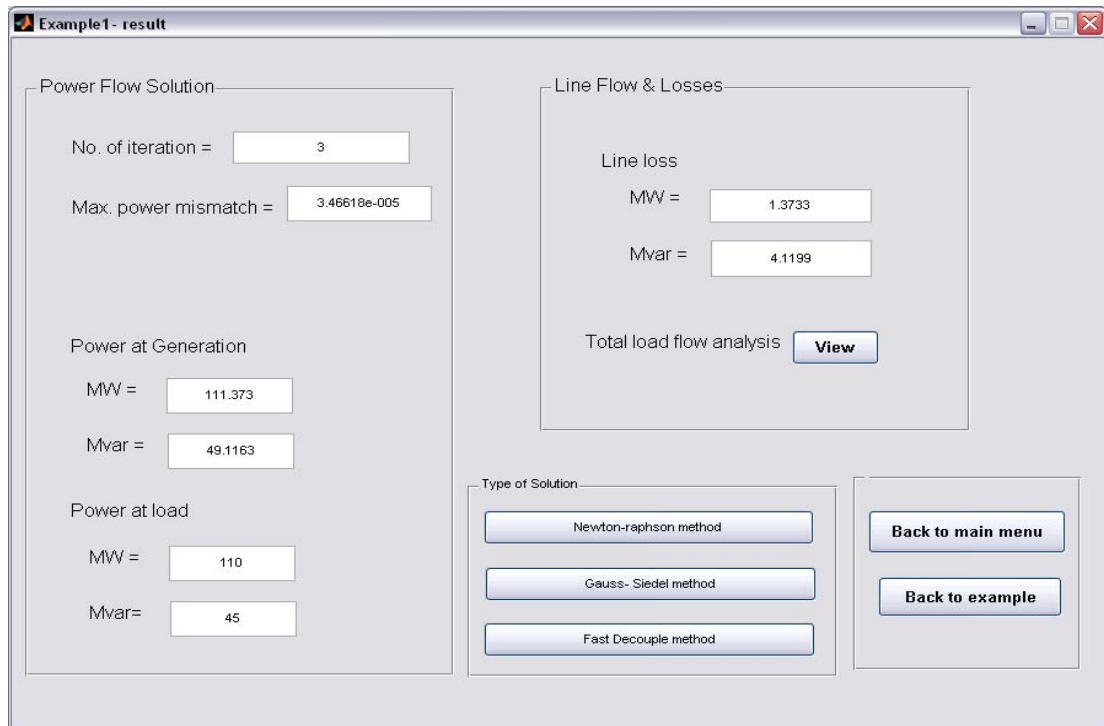
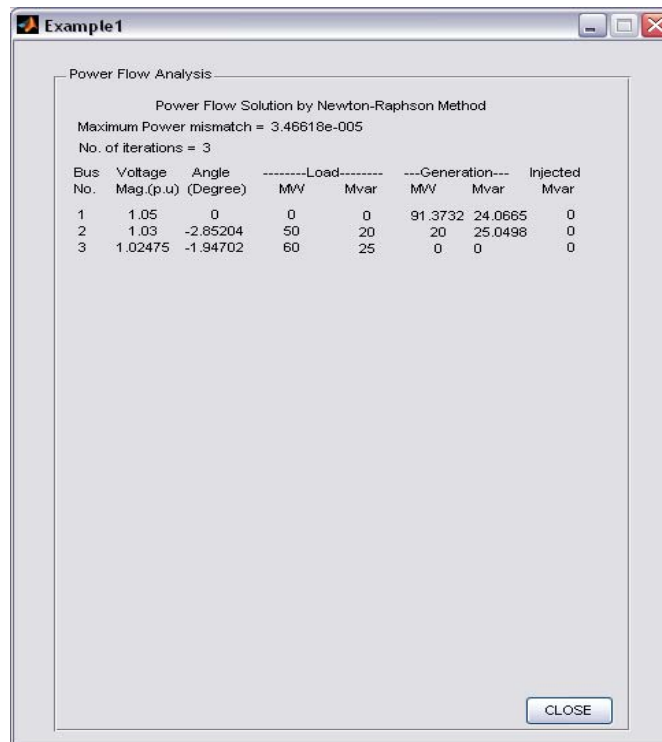


Figure 4.10: Example 1 – Data Preparation



**Figure 4.11:** Example 1 – Result



**Figure 4.12:** Example 1 - Power flow analysis

```

Command Window
File Edit Debug Desktop Window Help

Line Flow and Losses

--Line-- Power at bus & line flow --Line loss-- Transformer
from to MW Mvar MVA MW Mvar tap

1 91.373 24.067 94.490
2 22.972 1.651 23.031 0.385 1.155
3 68.401 22.418 71.981 0.940 2.820

2 -30.000 5.050 30.422
1 -22.587 -0.496 22.593 0.385 1.155
3 -7.413 5.547 9.259 0.048 0.145

3 -60.000 -25.000 65.000
1 -67.461 -19.598 70.251 0.940 2.820
2 7.461 -5.402 9.211 0.048 0.145

Total loss 1.373 4.120

>>

```

**Figure 4.13:** Example 1 - Line flow analysis

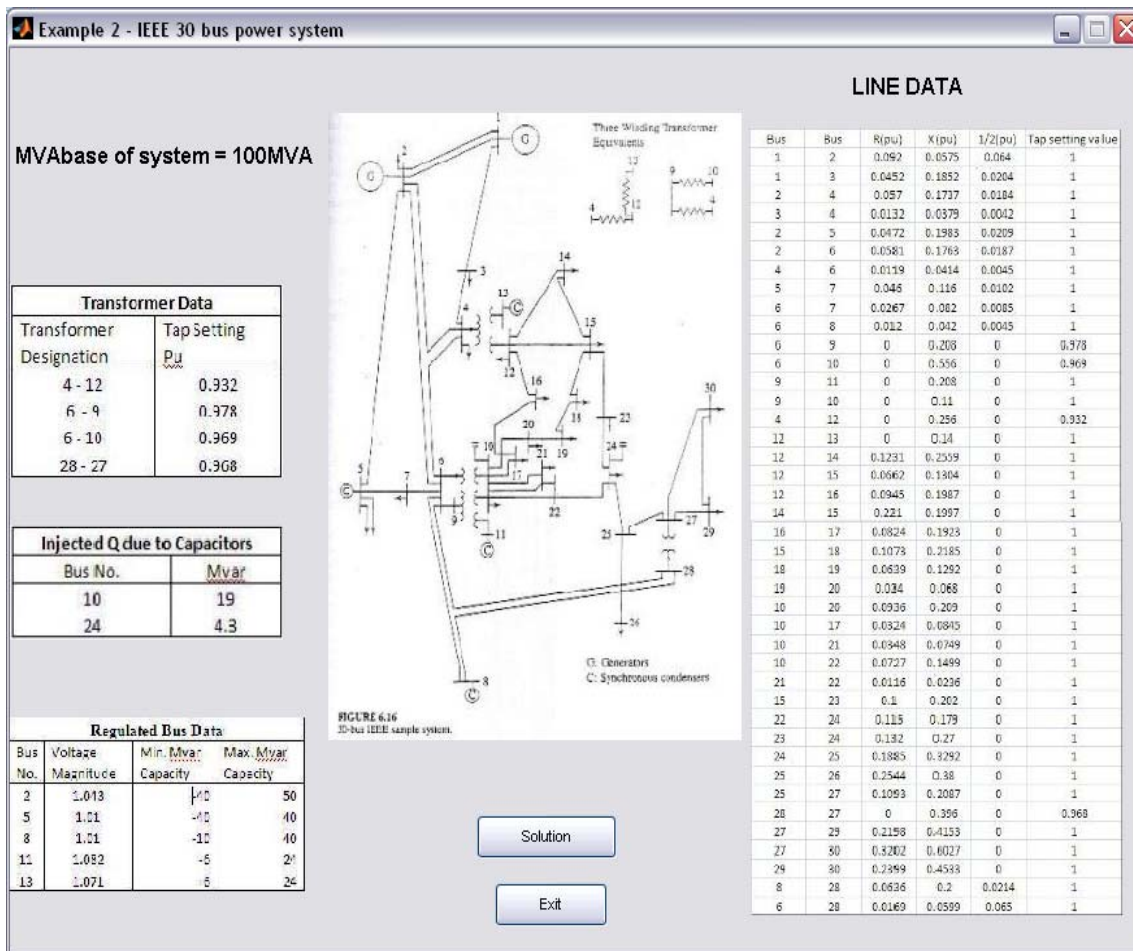


Figure 4.14: Example 2

**Example 2 - Data preparation**

MVA base of the system = 100

Bus Data

Bus no	Bus code	Voltage mag.	Angle Degree	Load		Generator			Injected	
				MW	Mvar	MW	Bus no	Qmin	Qmax	Mvar
1	1	1.06	0.0	0.0	0.0	0.0	0.0	0	0	0
2	2	1.043	0.0	21.70	12.7	40.0	0.0	-40	50	0
3	0	1.0	0.0	2.4	1.2	0.0	0.0	0	0	0
4	0	1.06	0.0	7.6	1.6	0.0	0.0	0	0	0
5	2	1.01	0.0	94.2	19.0	0.0	0.0	-40	40	0
6	0	1.0	0.0	0.0	0.0	0.0	0.0	0	0	0
7	0	1.0	0.0	22.8	10.9	0.0	0.0	0	0	0
8	2	1.01	0.0	30.0	30.0	0.0	0.0	-30	40	0
9	0	1.0	0.0	0.0	0.0	0.0	0.0	0	0	0

Line Data

Bus nr	Bus nr	R (pu)	X (pu)	1/2 (pu)	B1 for line code or tap setting value
1	2	0.0192	0.0575	0.02640	1
1	3	0.0452	0.1852	0.02040	1
2	4	0.0570	0.1737	0.01840	1
3	4	0.0132	0.0379	0.00420	1
2	5	0.0472	0.1983	0.02090	1
2	6	0.0581	0.1763	0.01870	1
4	6	0.0119	0.0414	0.00450	1
5	7	0.0460	0.1160	0.01020	1
6	7	0.0267	0.0820	0.00850	1
6	8	0.0120	0.0420	0.00450	1

Panel

Help

Proceed

CLOSE

**Figure 4.15:** Example 2 - Data preparation

**Example 2 - result**

Power Flow Solution

No. of iteration = 4

Max. power mismatch = 7.54899e-007

Power at Generation

MW = 300.998

Mvar = 125.144

Power at load

MW = 283.4

Mvar = 126.2

Line Flow & Losses

Line loss

MW = 17.5985

Mvar = 22.2444

Total load flow analysis [View](#)

Type of Solution

Newton-raphson method

Gauss- Siedel method

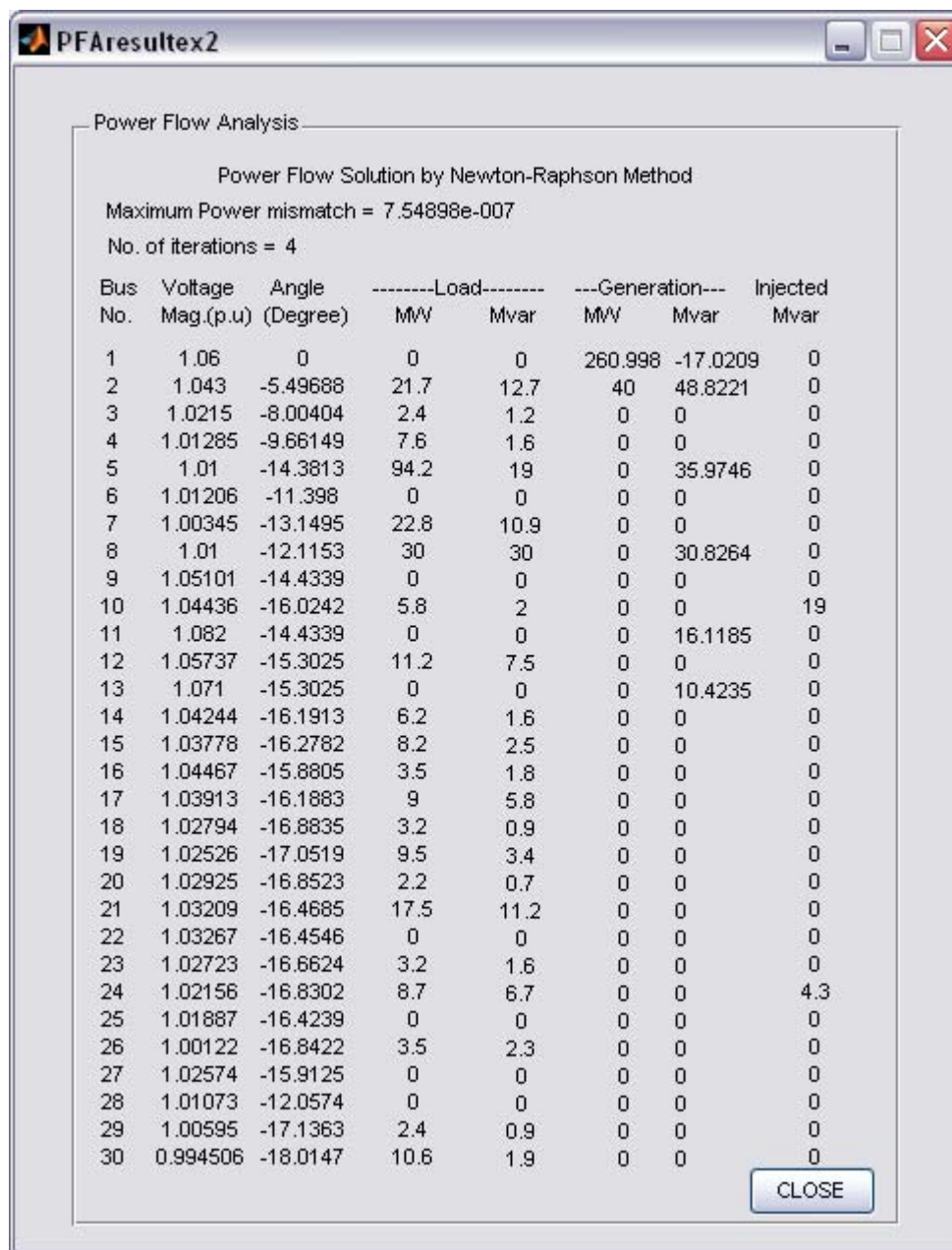
Fast Decouple method

[Back to main menu](#)

[Back to example](#)

**Figure 4.16:** Example 2 - Result





**Figure 4.17:** Example 2 - Power flow analysis

Line	Component 1	Component 2	Component 3	Component 4	Component 5	Component 6	Component 7
25	0.000	0.000	0.000				
24	1.333	-1.588	2.073	0.008	0.014		
26	3.545	2.366	4.262	0.045	0.066		
27	-4.877	-0.778	4.939	0.026	0.049		
26	-3.500	-2.300	4.188				
25	-3.500	-2.300	4.188	0.045	0.066		
27	0.000	0.000	0.000				
25	4.903	0.827	4.972	0.026	0.049		
28	-18.184	-4.157	18.653	0.000	1.309		
29	6.189	1.668	6.410	0.086	0.162		
30	7.091	1.661	7.283	0.161	0.304		
28	0.000	0.000	0.000				
27	18.184	5.466	18.987	0.000	1.309	0.968	
8	0.575	-1.999	2.080	0.000	-4.368		
6	-18.759	-3.467	19.077	0.060	-13.086		
29	-2.400	-0.900	2.563				
27	-6.104	-1.506	6.286	0.086	0.162		
30	3.704	0.606	3.753	0.033	0.063		
30	-10.600	-1.900	10.769				
27	-6.930	-1.358	7.062	0.161	0.304		
29	-3.670	-0.542	3.710	0.033	0.063		
Total loss				17.599	22.244		

**Figure 4.18:** Example 2 - Line flow analysis

#### 4.2.2.2 POWERSOLVE Program

Run pushbutton for third option on powersolve main menu will start the program by open inputdata or data preparation window as shown in figure 4.19. For this window user need to fill all input data before user click on 'PROCEED' button on assist panel.

Help button provided if user uncertain of filling all information. By click on this button, tutorial window will appear and user can refer bus data and line data format. Beside bus data and line data, user also required to fill the MVA base of the system.

The screenshot shows a software window titled "Data preparation". At the top, there is a text label "MVA base of the system =" followed by an empty input box. Below this, the window is divided into three main sections:

- Bus Data:** A table with columns for Bus no, Bus code, Voltage mag. (p.u.), Angle Degree, Load (MW, Mvar), Generator (MW, Mvar), and Injected (Gmin, Gmax, Mvar). The table is currently empty.
- Line Data:** A table with columns for Bus nr, Bus nr, R (pu), X (pu), 1/2 (pu), and B 1 for line code or tap setting value. The table is currently empty.
- Assist Panel:** A vertical panel on the right containing three buttons: "Help", "Proceed", and "CLOSE".

**Figure 4.19:** Input data window

After user finish fill in all the data required, click on 'proceed' button to proceed to result window as shown in Figure 4.20. The result window consists of all three method power flow analysis. User need to choose which method to be used to solve the problem. After user click on any of provided solution method, the analysis will run and the result will display on the same window. A window that shows complete detail of power flow analysis also will appear. Figure 4.21 shows the example of complete detail

of power when run on simple three bus system using Newton-Raphson method. To view result line flow analysis, user need to click on view button in 'Line Flow & Losses' section. Line flow result will appear in command window as shown in Figure 4.22.

The screenshot shows a software window titled "Result" with a standard Windows-style title bar (minimize, maximize, close buttons). The window is divided into several sections:

- Power Flow Solution:** This section contains input fields for:
  - No. of iteration = [ ]
  - Max. power mismatch = [ ]
  - Power at generation MW = [ ]
  - Power at load MW = [ ]
  - Mvar = [ ]
  - Mvar = [ ]
- Type of Solution:** A panel containing three buttons:
  - Newton-Raphson method
  - Gauss- Seidel method
  - Fast Decoupled method
- Line Flow & Losses:** A panel containing:
  - Total Line loss MW = [ ]
  - Mvar = [ ]
  - Total load flow analysis [View]
- Assist Panel:** A panel containing two buttons:
  - Back to main menu
  - Run another analysis

**Figure 4.20:** Result data window

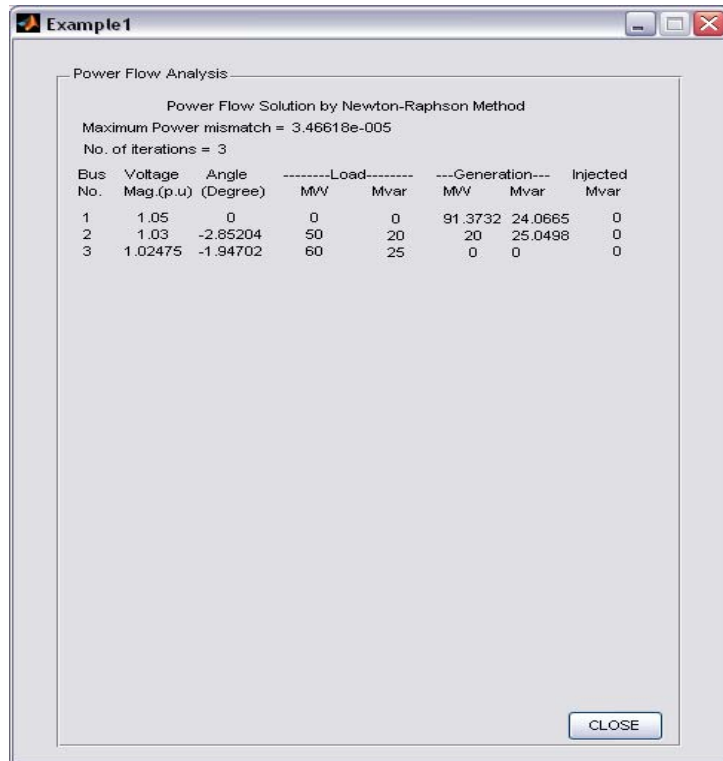


Figure 4.21: Power flow analysis window

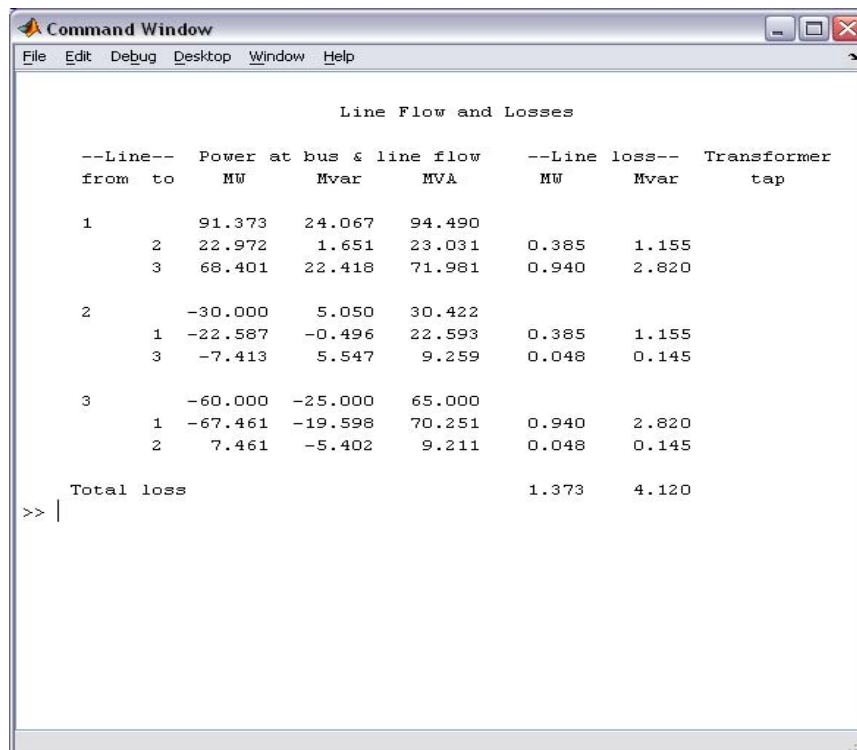


Figure 4.22: Line flow analysis window

User can perform the analysis using other method on the same problem by click another two methods. The result will display on the same window of previous method of analysis. User can compare the results of all three methods of power flow analysis.

### 4.3 Power Flow Input Data Formula

#### a) Bus data format

**Table 4.1:** Bus data format

Column no.	Item	Notes
1	Bus number	Integer
2	Bus code	Reference bus = 1 Load bus = 0 Regulated buses = 2
3	Voltage Magnitude	Per unit system (p.u)
4	Phase angle	Degree unit
5	Load MW	Real power
6	Load Mvar	Reactive power
7	Generator MW	Generated real power
8	Generator Mvar	Generated reactive power
9	Generator Qmin	Generated reactive power
10	Generator Qmax	Generated reactive power
11	Injected Mvar	Injected reactive power

**b) Line Data Format**

**Table 4.2:** Line data format

Column no.	Item	Notes
1	Line bus data	Integer
2	Line bus data	Integer
3	Line resistance	Per unit system (p.u)
4	Line reactance	Per unit system (p.u)
5	1/2 of the total line charging susceptance	Per unit system (p.u)
6	Transformer tap setting	Per unit system (p.u)

**4.4 Power Flow Analysis Simulation using MATLAB**

This sub-chapter will discuss the result of simulation for 3 bus system and 30 bus system using Newton-Raphson method, Gauss-Seidel method and Fast Decoupled method. One-line diagram for both system have been shown in Chapter 3 section 3.4.

**4.4.1 Result on 3-Bus System Analysis**

**4.4.1.1 Newton-Raphson method**

**Table 4.3:** Newton-Raphson analysis on 3 bus system

Bus no.	Voltage Magnitude	Angle degree	Load		Generation		Injected Mvar
			MW	Mvar	MW	Mvar	
1	1.050	0.000	0.000	0.000	221.270	109.269	0.000
2	0.972	-3.519	400.000	250.000	0.000	0.000	0.000
3	1.040	-1.839	2.400	1.200	200.000	180.083	0.000

**Table 4.4:** Line loss on 3 bus system

Line		Power at bus and line flow			Line loss		Transformer
From	To	Mw	Mvar	MVA	MW	Mvar	Tap
1		221.270	109.269	246.779	-	-	-
	2	209.510	105.979	234.789	10.000	20.000	-
	3	11.790	3.294	12.241	0.014	0.408	-
2		-400.000	-250.000	471.699	-	-	-
	1	-199.510	-85.978	217.248	10.000	20.000	-
	3	-200.490	-164.022	259.035	8.886	17.772	-
3		197.600	178.883	266.542	-	-	-
	1	-11.776	-2.887	12.125	0.014	0.408	-
	2	209.376	181.793	277.285	8.886	17.772	-

Maximum Power Mismatch using Newton-Raphson method is 0.000266997. To obtain the result the software performed 3 iterations. Table 4.3 and Table 4.4 show full analysis of the system. From simulation:

Total generation =  $421.27 + j289.352$  MVA

Total load =  $402.4 + j251.2$  MVA

Total loss =  $18.900 + j38.180$  MVA

#### 4.4.1.2 Gauss-Seidel method

**Table 4.5:** Analysis on 3 bus system

Bus no.	Voltage Magnitude	Angle degree	Load		Generation		Injected
			MW	Mvar	MW	Mvar	Mvar
1	1.050	0.000	0.000	0.000	221.235	109.135	0.000
2	0.972	-3.517	400.000	250.000	0.000	0.000	0.000
3	1.040	-1.838	2.400	1.200	200.000	180.151	0.000



**Table 4.6:** Line loss on 3 bus system

Line		Power at bus and line flow			Line loss		Transformer
From	To	Mw	Mvar	MVA	MW	Mvar	Tap
1		221.235	109.135	246.689	-	-	-
	2	209.479	106.033	234.786	10.000	20.000	-
	3	11.784	3.294	12.236	0.014	0.407	-
2		-400.000	-250.000	471.699	-	-	-
	1	-199.480	-86.033	217.241	10.000	20.000	-
	3	-200.489	-164.084	259.074	8.889	17.778	-
3		197.600	178.951	266.588	-	-	-
	1	-11.771	-2.887	12.120	0.014	0.407	-
	2	209.378	181.861	277.331	8.889	17.778	-

Maximum Power Mismatch using Gauss-Seidel method is 0.000704109. To obtain the result the software performed 5 iterations. Table 4.5 and Table 4.6 show full analysis of the system. From simulation:

Total generation =  $421.235 + j289.286$  MVA

Total load =  $402.4 + j251.2$  MVA

Total loss =  $18.902 + j38.185$  MVA

#### 4.4.1.3 Fast Decouple method

**Table 4.7:** Analysis on 3 bus system

Bus no.	Voltage Magnitude	Angle Degree	Load		Generation		Injected
			MW	Mvar	MW	Mvar	Mvar
1	1.050	0.000	0.000	0.000	221.285	109.275	0.000
2	0.972	-3.519	400.000	250.000	0.000	0.000	0.000
3	1.040	-1.839	2.400	1.200	200.000	180.114	0.000

**Table 4.8:** Line loss on 3 bus system

Line		Power at bus and line flow			Line loss		Transformer
From	To	Mw	Mvar	MVA	MW	Mvar	Tap
1		221.285	109.275	246.796	-	-	-
	2	209.508	105.968	234.783	10.000	19.999	-
	3	11.790	3.294	12.241	0.014	0.408	-
2		-400.000	-250.000	471.699	-	-	-
	1	-199.509	-85.969	217.242	10.000	19.999	-
	3	-200.486	-164.007	259.0230	8.885	17.770	-
3		197.600	178.914	266.563	-	-	-
	1	-11.776	-2.887	12.125	0.014	0.408	-
	2	209.371	181.776	277.271	8.885	17.770	-

Maximum Power Mismatch using Fast Decoupled method is 0.000455801. To obtain the result the software performed 13 iterations. Table 4.7 and Table 4.8 show full analysis of the system. From simulation:

Total generation =  $421.285 + j289.389$  MVA

Total load =  $402.4 + j251.2$  MVA

Total loss =  $18.898 + j38.177$  MVA

#### 4.4.1.4 Discussion on 3 system bus analysis

**Table 4.9:** Simulation result for 3-bus system

	<b>Power Flow Solution</b>		
	Newton-Raphson	Gauss-Seidel	Fast decoupled
No. of iteration	3	5	13
max. power mismatch	0.000266997	0.000704109	0.000455801
Total line loss	18.900 + j38.180 MVA	18.902 + j38.185 MVA	18.898 + j38.177MVA

Table 4.9 shows the result from simulation of 3 bus using MATLAB programming. For Newton-Raphson method, number of iteration is equal to 3 and the maximum power mismatch equal to 0.000266997. For Gauss-Seidel method, number of iteration equal to 5 and 0.000704109 maximum power mismatch. The last method, fast decoupled method needed 13 iterations to obtain the result. The maximum power mismatch for this method is 0.000455801. The result shows that the total generation and line losses almost same for all methods.

#### 4.4.2 Result on IEEE 30-Bus System Analysis

##### 4.4.2.1 Newton-Raphson method

**Table 4.10:** Newton-Raphson Analysis on 30 bus system

Bus no.	Voltage magnitude	Angle Degree	Load		Generation		Injected
			MW	Mvar	MW	Mvar	Mvar
1	1.060	0.000	0.000	0.000	260.998	-17.021	0.000
2	1.043	-5.497	21.700	12.700	40.000	48.822	0.000
3	1.022	-8.004	2.400	1.200	0.000	0.000	0.000
4	1.013	-9.661	7.600	1.600	0.000	0.000	0.000
5	1.010	-14.381	94.200	19.000	0.000	35.975	0.000
6	1.012	-11.398	0.000	0.000	0.000	0.000	0.000
7	1.003	-13.150	22.800	10.900	0.000	0.000	0.000
8	1.010	-12.115	30.000	30.000	0.000	30.826	0.000
9	1.051	-14.434	0.000	0.000	0.000	0.000	0.000
10	1.044	-16.024	5.800	2.000	0.000	0.000	19.000
11	1.082	-14.434	0.000	0.000	0.000	16.119	0.000
12	1.057	-15.302	11.200	7.500	0.000	0.000	0.000
13	1.071	-15.302	0.000	0.000	0.000	10.423	0.000
14	1.042	-16.191	6.200	1.600	0.000	0.000	0.000
15	1.038	-16.278	8.200	2.500	0.000	0.000	0.000
16	1.045	-15.880	3.500	1.800	0.000	0.000	0.000
17	1.039	-16.188	9.000	5.800	0.000	0.000	0.000
18	1.028	-16.884	3.200	0.900	0.000	0.000	0.000
19	1.025	-17.052	9.500	3.400	0.000	0.000	0.000
20	1.029	-16.852	2.200	0.700	0.000	0.000	0.000
21	1.032	-16.468	17.500	11.200	0.000	0.000	0.000
22	1.033	-16.455	0.000	0.000	0.000	0.000	0.000
23	1.027	-16.662	3.200	1.600	0.000	0.000	0.000
24	1.022	-16.830	8.700	6.700	0.000	0.000	4.300
25	1.019	-16.424	0.000	0.000	0.000	0.000	0.000
26	1.001	-16.842	3.500	2.300	0.000	0.000	0.000
27	1.026	-15.912	0.000	0.000	0.000	0.000	0.000
28	1.011	-12.057	0.000	0.000	0.000	0.000	0.000
29	1.006	-17.136	2.400	0.900	0.000	0.000	0.000
30	0.995	-18.015	10.600	1.900	0.000	0.000	0.000

**Table 4.11:** Line loss on 30 bus system

Line		Power at bus and line flow			Line loss		Transformer
From	To	Mw	Mvar	MVA	MW	Mvar	Tap
1		260.998	-17.021	261.553	-	-	-
	2	177.778	-22.148	179.152	5.464	10.524	-
	3	83.221	5.127	83.378	2.808	7.085	-
2		18.300	36.122	40.493	-	-	-
	1	-172.314	32.671	175.384	5.464	10.524	-
	4	45.712	2.705	45.792	1.106	-0.517	-
	5	82.990	1.703	83.008	2.995	8.178	-
3	6	61.912	-0.958	61.920	2.048	2.264	-
		-2.400	-1.200	2.683	-	-	-
3	1	-80.412	1.958	80.436	2.808	7.085	-
	4	78.012	-3.158	78.076	0.771	1.344	-
4		-7.600	-1.600	7.767	-	-	-
	2	-44.605	-3.222	44.722	1.106	-0.517	-
	3	-77.242	4.503	77.373	0.771	1.344	-
	6	70.126	-17.526	72.282	0.604	1.179	-
	12	44.121	14.646	46.489	0.000	4.685	0.932
5		-94.200	16.975	95.717	-	-	-
	2	-79.995	6.475	80.257	2.995	8.178	-
	7	-14.205	10.500	17.664	0.151	-1.687	-
6		0.000	0.000	0.000	-	-	-
	2	-59.864	3.222	59.951	2.048	2.264	-
	4	-69.521	18.705	71.994	0.604	1.179	-
	7	37.523	-1.885	37.570	0.367	-0.598	-
	8	29.528	-3.754	29.766	0.103	-0.558	-
	9	27.693	-7.322	28.644	0.000	1.594	0.978
	10	15.823	0.653	15.836	0.000	1.278	0.969
28	18.819	-9.618	21.134	0.060	-13.086	-	
7		-22.800	-10.900	25.272	-	-	-
	5	14.356	-12.187	18.831	0.151	-1.687	-
	6	-37.156	1.287	37.178	0.367	-0.598	-

8		-30.000	0.826	30.011	-	-	-
	6	-29.425	3.196	29.598	0.103	-0.558	-
	28	-0.575	-2.37	2.438	0.000	-4.368	-
9		0.000	0.000	0.000	-	-	-
	6	-27.693	8.916	29.093	0.000	1.594	-
	11	0.000	-15.657	15.657	0.000	0.462	-
	10	27.693	6.741	28.501	0.000	0.809	-
10		-5.800	17.000	17.962	-	-	-
	6	-15.823	0.626	15.835	0.000	1.278	-
	9	-27.693	-5.932	28.321	0.000	0.809	-
	20	9.027	3.560	9.704	0.081	0.180	-
	17	5.372	4.414	6.953	0.014	0.037	-
	21	15.733	9.842	18.558	0.110	0.237	-
	22	7.583	4.490	8.813	0.052	0.107	-
11		0.000	16.119	16.119	-	-	-
	9	0.000	16.119	16.119	0.000	0.462	-
12		-11.200	-7.500	13.479	-	-	-
	4	-44.121	-9.961	45.232	0.000	4.685	-
	13	0.000	-10.291	10.291	0.000	0.133	-
	14	7.856	2.442	8.227	0.075	0.155	-
	15	17.857	6.947	19.161	0.217	0.428	-
	16	7.208	3.363	7.954	0.053	0.112	-
13		0.000	10.423	10.423	-	-	-
	12	0.000	10.424	10.424	0.000	0.133	-
14		-6.200	-1.600	6.403	-	-	-
	12	-7.782	-2.287	8.111	0.075	0.155	-
	15	1.582	0.687	1.724	0.006	0.005	-
15		-8.200	-2.500	8.573	-	-	-
	12	-17.64	-6.519	18.806	0.217	0.428	-
	14	-1.576	-0.681	1.717	0.006	0.005	-
	18	6.014	1.744	6.262	0.039	0.080	-
	23	5.001	2.956	5.810	0.031	0.063	-
16		-3.500	-1.800	3.936	-	-	-
	12	-7.154	-3.251	7.858	0.053	0.112	-
	17	3.654	1.451	3.932	0.012	0.027	-

17		-9.000	-5.800	10.707	-	-	-
	16	-3.643	-1.424	3.911	0.012	0.027	-
	10	-5.357	-4.376	6.918	0.014	0.037	-
18		-3.200	-0.900	3.324	-	-	-
	15	-5.975	-1.665	6.203	0.039	0.08	-
	19	2.775	0.765	2.879	0.005	0.01	-
19		-9.500	-3.400	10.090	-	-	-
	18	-2.770	-0.755	2.871	0.005	0.010	-
	20	-6.730	-2.645	7.231	0.017	0.034	-
20		-2.200	-0.700	2.309	-	-	-
	19	6.747	2.679	7.259	0.017	0.034	-
	10	-8.947	-3.379	9.564	0.081	0.180	-
21		-17.500	-11.200	20.777	-	-	-
	10	-15.623	-9.606	180.340	0.110	0.237	-
	22	-1.877	-1.594	2.462	0.001	0.001	-
22		0.000	0.000	0.000	-	-	-
	10	-7.531	-4.384	8.714	0.052	0.107	-
	21	1.877	1.596	2.464	0.001	0.001	-
23		-3.200	-1.600	3.578	-	-	-
	15	-4.970	-2.893	5.751	0.031	0.063	-
	24	1.770	1.293	2.192	0.006	0.012	-
24		-8.700	-2.400	9.025	-	-	-
	22	-5.611	-2.721	6.236	0.043	0.067	-
	23	-1.764	-1.280	2.180	0.006	0.012	-
25		-1.325	1.602	2.079	0.008	0.014	-
	24	0.000	0.000	0.000	-	-	-
	26	1.333	-1.588	2.073	0.008	0.014	-
26		3.545	2.366	4.262	0.045	0.066	-
	27	-4.877	-0.778	4.939	0.026	0.049	-
	25	-3.500	-2.300	4.188	-	-	-
		-3.500	-2.300	4.188	0.045	0.066	-

27		0.000	0.000	0.000	-	-	-
	25	4.903	0.827	4.972	0.026	0.049	-
	28	-18.184	-4.157	18.653	0.000	1.309	-
	29	6.189	1.668	6.410	0.086	0.162	-
	30	7.091	1.661	7.283	0.161	0.304	-
28		0.000	0.000	0.000	-	-	-
	27	18.184	5.466	18.987	0.000	1.309	0.968
	8	0.575	-1.999	2.080	0.000	-4.368	-
	6	18.759	-3.467	19.077	0.060	-13.086	-
29		-2.400	-0.900	2.563	-	-	-
	27	-6.104	-1.506	6.286	0.086	0.162	-
	30	3.704	0.606	3.753	0.033	0.063	-
30		-10.600	-1.900	10.769	-	-	-
	27	-6.930	-1.358	7.062	0.161	0.304	-
	29	-3.670	-0.542	3.710	0.033	0.063	-

Maximum Power Mismatch using Newton-Raphson method is  $7.54898e-007$ . To obtain the result the software performed 4 iterations. Table 4.10 and Table 4.11 show full analysis of the system. From simulation:

Total generation =  $300.998 + j125.144$  MVA

Total load =  $283.4 + j126.2$  MVA

Total loss =  $17.599 + j22.244$  MVA



#### 4.4.2.2 Gauss-Seidel method

**Table 4.12:** Gauss-Seidel Analysis on 30 bus system

Bus no.	Voltage Magnitude	Angle Degree	Load		Generation		Injected Mvar
			MW	Mvar	MW	Mvar	
1	1.060	0.000	0.000	0.000	260.950	-17.010	0.000
2	1.043	-5.496	21.700	12.700	40.000	48.826	0.000
3	1.022	-8.002	2.400	1.200	0.000	0.000	0.000
4	1.013	-9.659	7.600	1.600	0.000	0.000	0.000
5	1.010	-14.380	94.200	19.000	0.000	35.995	0.000
6	1.012	-11.396	0.000	0.000	0.000	0.000	0.000
7	1.003	-13.149	22.800	10.900	0.000	0.000	0.000
8	1.010	-12.114	30.000	30.000	0.000	30.759	0.000
9	1.051	-14.432	0.000	0.000	0.000	0.000	0.000
10	1.044	-16.024	5.800	2.000	0.000	0.000	19.000
11	1.082	-14.432	0.000	0.000	0.000	16.113	0.000
12	1.057	-15.301	11.200	7.500	0.000	0.000	0.000
13	1.071	-15.300	0.000	0.000	0.000	10.406	0.000
14	1.043	-16.190	6.200	1.600	0.000	0.000	0.000
15	1.038	-16.276	8.200	2.500	0.000	0.000	0.000
16	1.045	-15.879	3.500	1.800	0.000	0.000	0.000
17	1.039	-16.187	9.000	5.800	0.000	0.000	0.000
18	1.028	-16.881	3.200	0.900	0.000	0.000	0.000
19	1.025	-17.049	9.500	3.400	0.000	0.000	0.000
20	1.029	-16.851	2.200	0.700	0.000	0.000	0.000
21	1.032	-16.468	17.500	11.200	0.000	0.000	0.000
22	1.033	-16.455	0.000	0.000	0.000	0.000	0.000
23	1.027	-16.660	3.200	1.600	0.000	0.000	0.000
24	1.022	-16.829	8.700	6.700	0.000	0.000	4.300
25	1.019	-16.423	0.000	0.000	0.000	0.000	0.000
26	1.001	-16.835	3.500	2.300	0.000	0.000	0.000
27	1.026	-15.913	0.000	0.000	0.000	0.000	0.000
28	1.011	-12.056	0.000	0.000	0.000	0.000	0.000
29	1.006	-17.133	2.400	0.900	0.000	0.000	0.000
30	0.994	-18.016	10.600	1.900	0.000	0.000	0.000

**Table 4.13:** Line loss on 30 bus system

Line		Power at bus and line flow			Line loss		Transformer
From	To	Mw	Mvar	MVA	MW	Mvar	tap
1		260.950	-17.010	261.504	-	-	-
	2	177.743	-22.140	179.117	5.461	10.517	-
	3	83.197	5.125	83.354	2.807	7.079	-
2		18.300	36.126	40.497	-	-	-
	1	-172.282	32.657	175.350	5.461	10.517	-
	4	45.702	2.720	45.783	1.106	-0.519	-
	5	82.990	1.704	83.008	2.995	8.179	-
3	6	61.905	-0.966	61.913	2.047	2.263	-
		-2.400	-1.200	2.683	-	-	-
3	1	-80.390	1.954	80.414	2.807	7.079	-
	4	78.034	-3.087	78.095	0.771	1.345	-
4		-7.600	-1.600	7.767	-	-	-
	2	-44.596	-3.239	44.713	1.106	-0.519	-
	3	-77.263	4.432	77.390	0.771	1.345	-
	6	70.132	-17.624	72.313	0.605	1.181	-
	12	44.131	14.627	46.492	0.000	4.686	0.932
5		-94.200	16.995	95.721	-	-	-
	2	-79.995	6.474	80.256	2.995	8.178	-
	7	-14.210	10.467	17.649	0.151	-1.687	-
6		0.000	0.000	0.000	-	-	-
	2	-59.858	3.229	59.945	2.047	2.263	-
	4	-69.527	18.805	72.026	0.605	1.181	-
	7	37.537	-1.915	37.586	0.368	-0.598	-
	8	29.534	-3.712	29.766	0.103	-0.558	-
	9	27.687	-7.318	28.638	0.000	1.593	0.978
	10	15.828	0.656	15.842	0.000	1.279	0.969
28	18.840	-9.575	21.134	0.000	-13.085	-	
7		-22.800	-10.900	25.272	-	-	-
	5	14.361	-12.154	18.814	0.151	-1.687	-
	6	-37.170	1.317	37.193	0.368	-0.598	-

8		-30.000	0.759	30.01	-	-	-
	6	-29.431	3.154	29.599	0.103	-0.558	-
	28	-0.570	-2.366	2.433	0.00	-4.368	-
9		0.000	0.000	0.000	-	-	-
	6	-27.687	8.911	29.086	0.000	1.593	-
	11	0.003	-15.653	15.653	0.000	0.461	-
	10	27.731	6.747	28.540	0.000	0.811	-
10		-5.800	17.000	17.962	-	-	-
	6	-15.828	0.623	15.840	0.000	1.279	-
	9	-27.731	-5.936	28.359	0.000	0.811	-
	20	9.018	3.569	9.698	0.081	0.180	-
	17	5.347	4.393	6.920	0.014	0.037	-
	21	15.723	9.846	18.551	0.110	0.236	-
	22	7.582	4.487	8.811	0.052	0.107	-
11		0.000	16.113	16.113	-	-	-
	9	-0.003	16.114	16.114	0.000	0.461	-
12		-11.200	-7.500	13.479	-	-	-
	4	-44.131	-9.941	45.237	0.000	4.686	-
	13	-0.021	-10.274	10.274	0.000	0.132	-
	14	7.852	2.428	8.219	0.074	0.155	-
	15	17.852	6.968	19.164	0.217	0.428	-
	16	7.206	3.370	7.955	0.053	0.112	-
13		0.000	10.406	10.406	-	-	-
	12	0.021	10.406	10.406	0.000	0.132	-
14		-6.200	-1.600	6.403	-	-	-
	12	-7.778	-2.273	8.103	0.074	0.155	-
	15	1.592	0.708	1.742	0.006	0.006	-
15		-8.200	-2.500	8.573	-	-	-
	12	-17.634	-6.540	18.808	0.217	0.428	-
	14	-1.586	-0.702	1.734	0.006	0.006	-
	18	6.009	1.741	6.256	0.039	0.079	-
	23	5.004	2.963	5.815	.0310	0.063	-
16		-3.500	-1.800	3.936	-	-	-
	12	-7.152	-3.257	7.859	0.053	0.112	-
	17	3.658	1.440	3.931	0.012	0.027	-

17		-9.000	-5.800	10.707	-	-	-
	16	-3.646	-1.413	3.910	0.012	0.027	-
	10	-5.332	-4.355	6.885	0.014	0.037	-
18		-3.200	-0.900	3.324	-	-	-
	15	-5.970	-1.661	6.197	0.039	0.079	-
	19	2.779	0.787	2.888	0.005	0.010	-
19		-9.500	-3.400	10.090	-	-	-
	18	-2.774	-0.777	2.881	0.005	0.010	-
	20	-6.703	-2.675	7.217	0.017	0.034	-
20		-2.200	-0.700	2.309	-	-	-
	19	6.720	2.709	7.245	0.017	0.034	-
	10	8.937	-3.389	9.558	0.081	0.180	-
21		-17.500	-11.200	20.777	-	-	-
	10	-15.613	-9.609	18.333	0.110	0.236	-
	22	-1.849	-1.627	2.463	0.001	0.001	-
22		0.000	0.000	0.000	-	-	-
	10	-7.531	-4.380	8.712	0.052	0.107	-
	21	1.850	1.628	2.464	0.001	0.001	-
	24	5.643	2.795	6.297	0.043	0.067	-
23		-3.200	-1.600	3.578	-	-	-
	15	-4.972	-2.900	5.756	0.031	0.063	-
	24	1.771	1.282	2.186	0.060	0.012	-
24		-8.700	-2.400	9.025	-	-	-
	22	-5.601	-2.728	6.230	0.043	0.067	-
	23	-1.765	-1.270	2.174	0.006	0.012	-
	25	-1.322	1.604	2.079	0.008	0.014	-
25		0.000	0.000	0.000	-	-	-
	24	1.330	-1.590	2.073	0.008	0.014	-
	26	3.520	2.372	4.244	0.044	0.066	-
	27	-4.866	-0.786	4.929	0.026	0.049	-
26		-3.500	-2.300	4.188	-	-	-
	25	-3.476	-2.306	4.171	0.044	0.066	-

27		0.000	0.000	0.000	-	-	-
	25	4.892	0.835	4.963	0.026	0.049	-
	28	-18.192	-4.152	18.66	0.000	1.310	-
	29	6.178	1.675	6.401	0.086	0.162	-
	30	7.093	1.663	7.286	0.162	0.304	-
28		0.000	0.000	0.000	-	-	-
	27	18.192	5.463	18.994	0.000	1.310	0.968
	8	0.570	-2.003	2.082	0.000	-4.368	-
	6	-18.78	-3.510	19.106	0.060	-13.085	-
29		-2.400	-0.900	2.563	-	-	-
	27	-6.093	-1.513	6.278	0.086	0.162	-
	30	3.716	0.601	3.764	0.034	0.063	-
30		-10.600	-1.900	10.769	-	-	-
	27	-6.932	-1.359	7.064	0.162	0.304	-
	29	-3.683	-0.537	3.722	0.034	0.063	-

Maximum Power Mismatch using Gauss-Seidel method is 0.000951884. To obtain the result the software performed 34 iterations. Table 4.12 and Table 4.13 show full analysis of the system. From simulation:

Total generation =  $300.950 + j125.089$  MVA

Total load =  $283.4 + j126.2$  MVA

Total loss =  $17.594 + j22.233$  MVA

### 4.4.2.3 Fast-Decoupled method

**Table 4.14:** Fast Decoupled Analysis on 30 bus system

Bus no.	Voltage magnitude	Angle degree	Load		Generation		Injected
			MW	Mvar	MW	Mvar	Mvar
1	1.060	0.000	0.000	0.00	260.998	-17.021	0.000
2	1.043	-5.497	21.700	12.700	40.000	48.822	0.000
3	1.022	-8.004	2.400	1.200	0.000	0.000	0.000
4	1.013	-9.662	7.600	1.600	0.000	0.000	0.000
5	1.010	-14.381	94.200	19.000	0.000	35.975	0.000
6	1.012	-11.398	0.000	0.000	0.000	0.000	0.000
7	1.003	-13.149	22.800	10.900	0.000	0.000	0.000
8	1.010	-12.115	30.000	30.000	0.000	30.828	0.000
9	1.051	-14.434	0.000	0.000	0.000	0.000	0.000
10	1.044	-16.024	5.800	2.000	0.000	0.000	19.000
11	1.082	-14.434	0.000	0.000	0.000	16.120	0.000
12	1.057	-15.303	11.200	7.500	0.000	0.000	0.000
13	1.071	-15.303	0.000	0.000	0.000	10.421	0.000
14	1.042	-16.198	6.200	1.600	0.000	0.000	0.000
15	1.038	-16.276	8.200	2.500	0.000	0.000	0.000
16	1.045	-15.881	3.500	1.800	0.000	0.000	0.000
17	1.039	-16.188	9.000	5.800	0.000	0.000	0.000
18	1.028	-16.882	3.200	0.900	0.000	0.000	0.000
19	1.025	-17.051	9.500	3.400	0.000	0.000	0.000
20	1.029	-16.852	2.200	0.700	0.000	0.000	0.000
21	1.032	-16.468	17.500	11.200	0.000	0.000	0.000
22	1.033	-16.454	0.000	0.000	0.000	0.000	0.000
23	1.027	-16.661	3.200	1.600	0.000	0.000	0.000
24	1.022	-16.829	8.700	6.700	0.000	0.000	4.300
25	1.019	-16.423	0.000	0.000	0.000	0.000	0.000
26	1.001	-16.840	3.500	2.300	0.000	0.000	0.000
27	1.026	-15.912	0.000	0.000	0.000	0.000	0.000
28	1.011	-12.057	0.000	0.000	0.000	0.000	0.000
29	1.006	-17.136	2.400	0.900	0.000	0.000	0.000
30	0.995	-18.014	10.600	1.900	0.000	0.000	0.000

**Table 4.15:** Line loss on 30 bus system

Line		Power at bus and line flow			Line loss		Transformer
From	To	Mw	Mvar	MVA	MW	Mvar	tap
1		260.998	-17.021	261.553	-	-	-
	2	177.778	-22.148	179.152	5.464	10.524	-
	3	83.221	5.127	83.379	2.808	7.085	-
2		18.300	36.122	40.493	-	-	-
	1	-172.315	32.672	175.385	5.464	10.524	-
	4	45.712	2.705	45.792	1.107	-0.517	-
	5	82.990	1.703	83.008	2.995	8.178	-
3	6	61.912	-0.958	61.919	2.048	2.264	-
		-2.400	-1.200	2.683	-	-	-
	1	-80.413	1.959	80.437	2.808	7.085	-
4	4	78.013	-3.159	78.077	0.771	1.344	-
		-7.600	-1.600	7.767	-	-	-
	2	-44.606	-3.222	44.722	1.107	-0.517	-
	3	-77.242	4.503	77.373	0.771	1.344	-
	6	70.123	-17.525	72.280	0.604	1.179	-
5	12	44.124	14.646	46.491	0.000	4.685	0.932
		-94.200	16.975	95.717	-	-	-
	2	-79.995	6.475	80.257	2.995	8.178	-
	7	-14.205	10.500	17.664	0.151	-1.687	-
6		0.000	0.000	0.000	-	-	-
	2	-59.864	3.222	59.951	2.048	2.264	-
	4	-69.519	18.704	71.991	0.604	1.179	-
	7	37.523	-1.885	37.571	0.367	-0.598	-
	8	29.529	-3.755	29.766	0.103	-0.558	-
	9	27.692	-7.322	28.644	0.000	1.594	0.978
	10	15.822	0.653	15.836	0.000	1.278	0.969
	28	18.818	-9.618	21.133	0.060	-13.086	-
7		-22.800	-10.900	25.272	-	-	-
	5	14.356	-12.187	18.831	0.151	-1.687	-
	6	-37.156	1.287	37.178	0.367	-0.598	-

8		-30.000	0.828	30.011	-	-	-
	6	-29.425	3.197	29.598	0.103	-0.558	-
	28	-0.575	-2.369	2.438	0.000	-4.368	-
9		0.000	0.000	0.000	-	-	-
	6	-27.692	8.916	29.092	0.000	1.594	-
	11	0.000	-15.657	15.657	0.000	0.462	-
	10	27.692	6.741	28.501	0.000	0.809	-
10		-5.800	17.000	17.962	-	-	-
	6	-15.822	0.625	15.835	0.000	1.278	-
	9	-27.692	-5.933	28.320	0.000	0.0809	-
	20	9.023	3.563	9.701	0.081	0.180	-
	17	5.373	4.413	6.953	0.014	0.037	-
	21	15.732	9.843	18.558	0.110	0.237	-
	22	7.583	4.491	8.813	0.052	0.107	-
11		0.000	16.120	16.120	-	-	-
	9	0.000	16.119	16.119	0.000	0.462	-
12		-11.200	-7.500	13.479	-	-	-
	4	-44.124	-9.961	45.234	0.000	4.685	-
	13	0.000	-10.291	10.291	0.000	0.133	-
	14	7.891	2.418	8.253	0.075	0.156	-
	15	17.828	6.969	19.142	0.217	0.427	-
	16	7.206	3.364	7.953	0.053	0.112	-
13		0.000	10.421	10.421	-	-	-
	12	0.000	10.423	10.423	0.000	0.133	-
14		-6.200	-1.600	6.403	-	-	-
	12	-7.816	-2.262	8.137	0.075	0.156	-
	15	1.551	0.734	1.716	0.006	0.005	-
15		-8.200	-2.500	8.573	-	-	-
	12	-17.611	-6.541	18.787	0.217	0.427	-
	14	-1.545	-0.729	1.709	0.006	0.005	-
	18	6.020	1.740	6.266	0.039	0.08	-
	23	5.005	2.954	5.812	0.031	0.063	-
16		-3.500	-1.800	3.936	-	-	-
	12	-7.152	-3.252	7.857	0.053	0.112	-
	17	3.653	1.452	3.931	0.012	0.027	-



17		-9.000	-5.800	10.707	-	-	-
	16	-3.641	-1.424	3.910	0.012	0.027	-
	10	-5.359	-4.376	6.918	0.014	0.037	-
18		-3.200	-0.900	3.324	-	-	-
	15	-5.981	-1.660	6.207	0.039	0.080	-
	19	2.780	0.761	2.882	0.005	0.010	-
19		-9.500	-3.400	10.09	-	-	-
	18	-2.775	-0.751	2.875	0.005	0.010	-
	20	-6.725	-2.649	7.228	0.017	0.034	-
20		-2.200	-0.700	2.309	-	-	-
	19	6.742	2.683	7.256	0.017	0.034	-
	10	-8.943	-3.382	9.561	0.081	0.180	-
21		-17.500	-11.200	20.777	-	-	-
	10	-15.622	-9.606	18.340	0.110	0.237	-
	22	-1.878	-1.593	2.463	0.001	0.001	-
22		0.000	0.000	0.000	-	-	-
	10	-7.531	-4.384	8.714	0.052	0.107	-
	21	1.878	1.595	2.464	0.001	0.001	-
	24	5.648	2.795	6.301	0.043	0.067	-
23		-3.200	-1.600	3.578	-	-	-
	15	-4.974	-2.890	5.753	0.031	0.063	-
	24	1.773	1.291	2.193	0.006	0.012	-
24		-8.700	-2.400	9.025	-	-	-
	22	-5.605	-2.728	6.234	0.043	0.067	-
	23	-1.767	-1.278	2.181	0.006	0.012	-
	25	-1.322	1.600	2.075	0.008	0.014	-
25		0.000	0.000	0.000	-	-	-
	24	1.330	-1.586	2.070	0.008	0.014	-
	26	3.539	2.374	4.261	0.044	0.066	-
	27	-4.875	-0.781	4.937	0.026	0.049	-
26		-3.500	-2.300	4.188	-	-	-
	25	-3.494	-2.307	4.187	0.044	0.066	-

27		0.000	0.000	0.000	-	-	-
	25	4.901	0.830	4.970	0.026	0.049	-
	28	-18.183	-4.157	18.652	0.000	1.309	-
	29	6.189	1.668	6.410	0.086	0.162	-
	30	7.091	1.662	7.283	0.161	0.304	-
28		0.000	0.000	0.000	-	-	-
	27	18.183	5.466	18.987	0.000	1.309	0.968
	8	0.576	-1.999	2.080	0.000	-4.368	-
	6	-18.758	-3.468	19.076	0.060	-13.086	-
29		-2.400	-0.900	2.563	-	-	-
	27	-6.103	-1.506	6.286	0.086	0.162	-
	30	3.704	0.605	3.753	0.033	0.063	-
30		-10.600	-1.900	10.769	-	-	-
	27	-6.930	-1.358	7.062	0.161	0.304	-
	29	-3.670	-0.542	3.710	0.033	0.063	-

Maximum Power Mismatch using Fast-Decoupled method is 0.000919582. To obtain the result the software performed 15 iterations. Table 4.14 and Table 4.15 show full analysis of the system. From simulation:

Total generation =  $300.998 + j125.145$  MVA

Total load =  $283.4 + j126.2$  MVA

Total loss =  $17.598 + j22.245$  MVA

#### 4.4.2.4 Discussion on 30 bus system analysis

**Table 4.16:** Simulation result for IEEE 30-bus system

	<b>Power Flow Solution</b>		
	Newton-Raphson	Gauss-Seidel	Fast decouple
No. of iteration	4	34	15
max. power mismatch	7.55E-07	0.000951884	0.000919582
Total line loss	17.599 + j22.244 MVA	17.594 + j22.233 MVA	17.598 + j22.245MVA

Table 4.16 shows the result from simulation of 30 bus using MATLAB programming. For Newton-Raphson method, number of iteration is equal to 4 and the maximum power mismatch equal to 7.55E-07. For Gauss-Seidel method, number of iteration equal to 34 and 0.000951884 maximum power mismatch. The last methods, fast decoupled method needed 15 iterations to obtain the result. The maximum power mismatch for this method is 0.000919582. The result shows that the total generation and line losses almost same for all methods.

#### 4.5 Discussion and Analysis

From the simulation result, all the methods are accepted. All three methods produced almost the same result. However the differences between the methods are the number of iteration and maximum power mismatch. In choosing the best methods of solving power flow analysis there are factors need to be considered. One of the factors is maximum power mismatch. The maximum power mismatch shows the difference between power that we specified and the power that we obtained. So if the value closer to zero the result will be more accurate.

After analyses the results we can conclude that the best method to be used to solve power flow problem is Newton-Raphson method due to the maximum power mismatch closest to value zero.

## CHAPTER 5

### CONCLUSION AND RECOMMENDATION

#### 5.1 Conclusion

The design and program of Power Flow Analysis Toolbox Package ver. 1.0 (POWERSOLVE) has been presented in this project. The development of the power flow analysis program and graphical user interface (GUI) using MATLAB was done after detail study and analysis. Through the development of this project it has conclude that to solve power flow problem there are three methods can be used. Those three methods are Gauss-Seidel method, Newton-Raphson method and Fast-decouple method. These three methods have differences number of iteration and maximum power mismatch however produce the same result. The best method of solving power flow problem is Newton-Raphson method due to low number of iteration and low value of maximum power mismatch.

Designing and programming GUI seems easy however to do advanced programming needs more knowledge and effort. The objective of this project which is to develop software package and to simulate power flow analysis is successfully achieved. The main contribution of this project is designing the load flow analysis software package using MATLAB 7 for three types of analysis; Newton-Raphson power flow analysis, Gauss-Seidel power flow analysis and Fast Decoupled power flow analysis.

## **5.2 Future Recommendations**

For the future recommendations, to improve this project, others function on POWERSOLVE can be added example of performing optimal dispatch, fault analysis or transient study and power system stability analysis. By adding those analyses, POWERSOLVE will be multipurpose power system analysis software.

Beside the function, appearance of the POWERSOLVE can be improved like replacing certain pushbutton with other component in GUI. This will help the software turn to more attractive. Another recommendation is change POWERSOLVE designing so the software only runs on one window. This will avoid many GUI window appear at same time.

### **5.2.1 Commercialization**

The objective of developing POWERSOLVE toolbox is for education and training used. Lecturer of UMP have a permission of the author to use this software through out the teaching session for electrical and electronics course. Particularly this software will be very useful in learning of Power System & Analysis subject. The author gives full authority of this project and software to my supervisor Mrs. Norhafidzah binti Mohd Saad to handle further matter.

## REFERENCE

- [1] Saadat, H. *Power System Analysis*: Mc Graw Hill
- [2] Shin, J, R., (1999). *A Windows-based Interactive and Graphic Package For the Education and Training of Power System Analysis and Operation*, Kon-Kuk University, Seoul.
- [3] Nguyen, H, L. *Newton-Raphson Method in Complex Form*, The Vienna University of Technology, Austria.
- [4] M.S. Sachdev, T.K.P. Medicherla. *A Second order load flow technique*. IEEE Trans. On Power Apparatus and Systems, Vol. PAS-96, No 1, pp. 189-197, JanReb 1977
- [5] <http://www.bellevuelinux.org/gui.html>
- [6] <http://www.ewh.ieee.org/r8/uae/GUI.pdf>
- [7] [http://www.mathworks.com/access/helpdesk/help/pdf\\_doc/matlab/buildgui.pdf](http://www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/buildgui.pdf)
- [8] [http://www.rclsgi.eng.ohio-state.edu/~gateway/docs/pres\\_and\\_prog\\_reports/MatlabGUI.ppt](http://www.rclsgi.eng.ohio-state.edu/~gateway/docs/pres_and_prog_reports/MatlabGUI.ppt)
- [9] <http://myweb.msoe.edu/~saadat/psabook.htm>
- [10] Duane Hanselman, Bruce LITTEFIELD, *MASTERING MATLAB 7*: Pearson/Prentice Hall, 2005

## Appendix A

### M-file of main page of the software

```

1  function varargout = page1b(varargin)
2
3  % Begin initialization code - DO NOT EDIT
4  gui_Singleton = 1;
5  gui_State = struct('gui_Name',       mfilename, ...
6                    'gui_Singleton',  gui_Singleton, ...
7                    'gui_OpeningFcn', @page1b_OpeningFcn, ...
8                    'gui_OutputFcn',  @page1b_OutputFcn, ...
9                    'gui_LayoutFcn',  [] , ...
10                   'gui_Callback',   []);
11  if nargin && ischar(varargin{1})
12      gui_State.gui_Callback = str2func(varargin{1});
13  end
14
15  if nargin
16      [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
17  else
18      gui_mainfcn(gui_State, varargin{:});
19  end
20  % End initialization code - DO NOT EDIT
21
22
23  % --- Executes just before page1b is made visible.
24  function page1b_OpeningFcn(hObject, eventdata, handles, varargin)
25
26  % Choose default command line output for page1b
27  handles.output = hObject;
28
29  % Update handles structure
30  guidata(hObject, handles);
31
32  % UIWAIT makes page1b wait for user response (see UIRESUME)
33  % uiwait(handles.figure1);
34  axes(handles.axes_logo)
35  diagram = imread('logo.jpg');
36  imshow(diagram);
37
38  axes(handles.axes11)
39  diagram = imread('tower2.jpg');
40  imshow(diagram);
41  % --- Outputs from this function are returned to the command line.
42  function varargout = page1b_OutputFcn(hObject, eventdata, handles)
43
44  varargout{1} = handles.output;
45
46  % --- Executes on button press in pushbutton1.
47  function pushbutton1_Callback(hObject, eventdata, handles)
48
49  % --- Executes on button press in pushbutton2.
50  function pushbutton2_Callback(hObject, eventdata, handles)
51
52  % --- Executes on button press in pushbutton3.
53  function pushbutton3_Callback(hObject, eventdata, handles)
54
55  % --- Executes on button press in pushbutton4.
56  function varargout = pushbutton4_Callback(h, eventdata, handles, varargin)
57  figure(definition)
58
59  % --- Executes on button press in pushbutton5.
60  function pushbutton5_Callback(hObject, eventdata, handles)
61  figure(input_test)
62
63  % --- Executes on button press in close_pushbutton.
64  function close_pushbutton_Callback(hObject, eventdata, handles)
65

```

```

66 % Get the current position of the GUI from the handles structure
67 % to pass to the modal dialog.
68 pos_size = get(handles.figure1,'Position');
69 % Call modaldlg with the argument 'Position'.
70 user_response = modaldlg('Title','Confirm close');
71 switch user_response
72 case {'No'}
73 % take no action
74 case 'Yes'
75 % Prepare to close GUI application window
76 % .
77 % .
78 % .
79 delete(handles.figure1)
80 end
81
82 % -----
83 function Menu_Callback(hObject, eventdata, handles)
84
85 % -----
86 function Exit_Callback(hObject, eventdata, handles)
87
88 pos_size = get(handles.figure1,'Position');
89 % Call modaldlg with the argument 'Position'.
90 user_response = modaldlg('Title','Confirm close');
91 switch user_response
92 case {'No'}
93 % take no action
94 case 'Yes'
95 % Prepare to close GUI application window
96 % .
97 % .
98 % .
99 delete(handles.figure1)
100 end
101
102 % -----
103 function Credit_Callback(hObject, eventdata, handles)
104 figure(credits)
105
106 % -----
107 function Help_Callback(hObject, eventdata, handles)
108
109 % -----
110 function About_this_software_Callback(hObject, eventdata, handles)
111 figure(definition)
112
113
114
115 % --- Executes on button press in pushbutton7.
116 function pushbutton7_Callback(hObject, eventdata, handles)
117 figure(example)
118
119 % -----
120 function PFA_Callback(hObject, eventdata, handles)
121 figure(input_test)
122
123 % -----
124 function ET_Callback(hObject, eventdata, handles)
125
126 % -----
127 function Run_Callback(hObject, eventdata, handles)
128
129 % -----
130 function Tutorial_Callback(hObject, eventdata, handles)
131 figure(Tutorial)
132
133

```

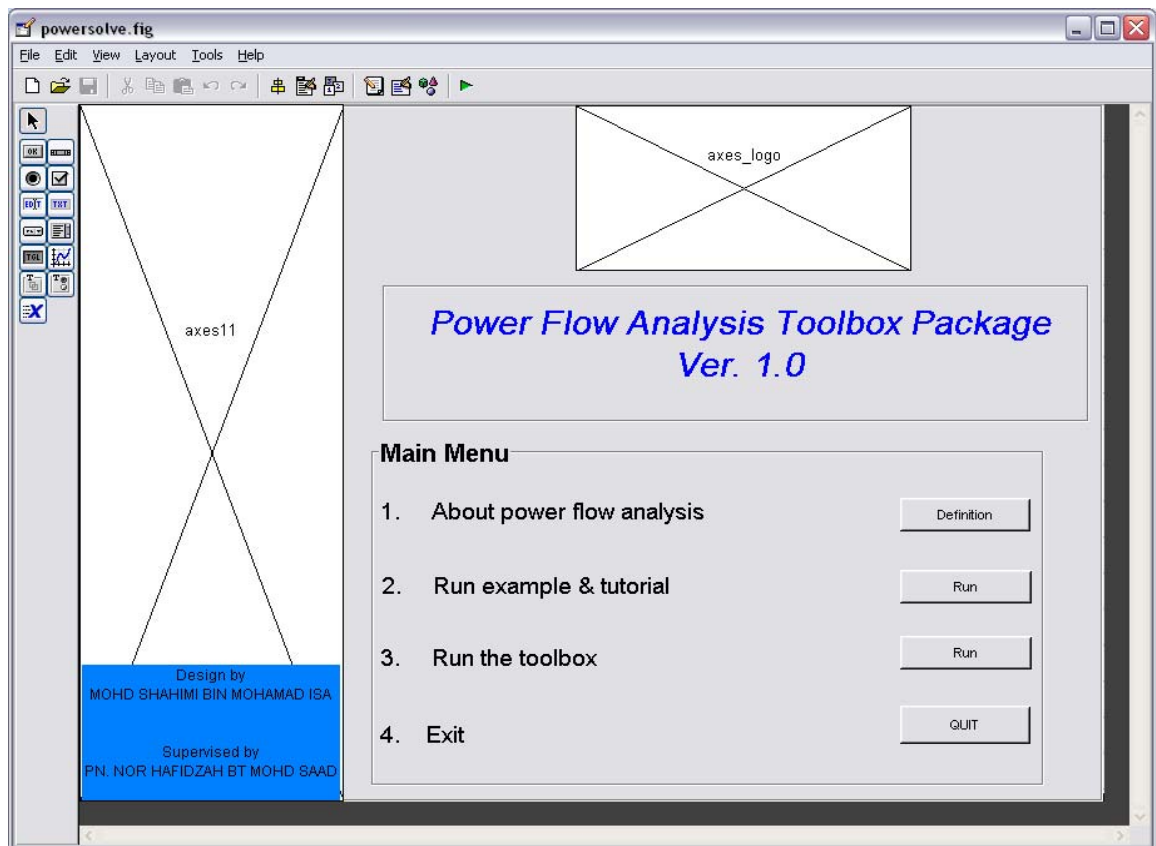


```

134
135 % -----
136 function Example1_Callback(hObject, eventdata, handles)
137 figure(example3)
138
139 % -----
140 function Example2_Callback(hObject, eventdata, handles)
141 figure(example2)
142

```

## Powersolve.fig



## Appendix B

### M-files for Newton-Raphson power flow analysis

```

1
2 - ns=0; ng=0; Vm=0; delta=0; yload=0; deltad=0;
3 - nbus = length(busdata(:,1));
4 - nbus1 = busdata(:,1);
5 - kb=[];Vm=[]; delta=[]; Pd=[]; Qd=[]; Pg=[]; Qg=[]; Qmin=[]; Qmax=[];
6 - PK=[]; P=[]; Qk=[]; Q=[]; S=[]; V=[]; % Added (6-8-00)
7 - for k=1:nbus
8 - n=busdata(k,1);
9 - kb(n)=busdata(k,2); Vm(n)=busdata(k,3); delta(n)=busdata(k, 4);
10 - Pd(n)=busdata(k,5); Qd(n)=busdata(k,6); Pg(n)=busdata(k,7); Qg(n) = busdata(k,8);
11 - Qmin(n)=busdata(k, 9); Qmax(n)=busdata(k, 10);
12 - Qsh(n)=busdata(k, 11);
13 -     if Vm(n) <= 0 Vm(n) = 1.0; V(n) = 1 + j*0;
14 -     else delta(n) = pi/180*delta(n);
15 -         V(n) = Vm(n)*(cos(delta(n)) + j*sin(delta(n)));
16 -         P(n)=(Pg(n)-Pd(n))/basemva;
17 -         Q(n)=(Qg(n)-Qd(n)+ Qsh(n))/basemva;
18 -         S(n) = P(n) + j*Q(n);
19 -     end
20 - end
21 - for k=1:nbus
22 -     if kb(k) == 1, ns = ns+1; else, end
23 -     if kb(k) == 2 ng = ng+1; else, end
24 -     ngs(k) = ng;
25 -     nss(k) = ns;
26 - end
27 - Ym=abs(Ybus); t = angle(Ybus);
28 - m=2*nbus-ng-2*ns;
29 - maxerror = 1; converge=1;
30 - iter = 0;
31
32 - mline=ones(nbr,1);
33 - for k=1:nbr
34 -     for m=k+1:nbr
35 -         if((nl(k)==nl(m)) & (nr(k)==nr(m)));
36 -             mline(m)=2;
37 -         elseif ((nl(k)==nr(m)) & (nr(k)==nl(m)));
38 -             mline(m)=2;
39 -         else, end
40 -     end
41 - end
42
43
44 - % Start of iterations
45 - clear A DC J DX
46 - while maxerror >= accuracy & iter <= maxiter % Test for max. power mismatch
47 - for ii=1:m
48 - for k=1:m
49 -     A(ii,k)=0; %Initializing Jacobian matrix
50 - end, end
51 - iter = iter+1;
52 - for n=1:nbus
53 - nn=n-nss(n);
54 - lm=nbus+n-ngs(n)-nss(n)-ns;
55 - J11=0; J22=0; J33=0; J44=0;
56 - for ii=1:nbr
57 -     if mline(ii)==1
58 -         if nl(ii) == n | nr(ii) == n
59 -             if nl(ii) == n , l = nr(ii); end
60 -             if nr(ii) == n , l = nl(ii); end

```

```

61 -         J11=J11+ Vm(n)*Vm(l)*Ym(n,l)*sin(t(n,l)- delta(n) + delta(l));
62 -         J33=J33+ Vm(n)*Vm(l)*Ym(n,l)*cos(t(n,l)- delta(n) + delta(l));
63 -         if Kb(n)~=1
64 -             J22=J22+ Vm(l)*Ym(n,l)*cos(t(n,l)- delta(n) + delta(l));
65 -             J44=J44+ Vm(l)*Ym(n,l)*sin(t(n,l)- delta(n) + delta(l));
66 -         else, end
67 -         if kb(n) ~= 1 & kb(l) ~=1
68 -             lk = nbus+1-ngs(l)-nss(l)-ns;
69 -             ll = 1 -nss(l);
70 -             % off diagonalelements of J1
71 -             A(nn, ll) =-Vm(n)*Vm(l)*Ym(n,l)*sin(t(n,l)- delta(n) + delta(l));
72 -             if kb(l) == 0 % off diagonal elements of J2
73 -                 A(nn, lk) =Vm(n)*Ym(n,l)*cos(t(n,l)- delta(n) + delta(l));end
74 -             if kb(n) == 0 % off diagonal elements of J3
75 -                 A(lm, ll) =-Vm(n)*Vm(l)*Ym(n,l)*cos(t(n,l)- delta(n)+delta(l)); end
76 -             if kb(n) == 0 & kb(l) == 0 % off diagonal elements of J4
77 -                 A(lm, lk) =-Vm(n)*Ym(n,l)*sin(t(n,l)- delta(n) + delta(l));end
78 -         else end
79 -         else , end
80 -     else, end
81 - end
82 - Pk = Vm(n)^2*Ym(n,n)*cos(t(n,n))+J33;
83 - Qk = -Vm(n)^2*Ym(n,n)*sin(t(n,n))-J11;
84 - if Kb(n) == 1 P(n)=Pk; Q(n) = Qk; end % Swing bus P
85 - if Kb(n) == 2 Q(n)=Qk;
86 -     if Qmax(n) ~= 0
87 -         Qgc = Q(n)*basemva + Qd(n) - Qsh(n);
88 -         if iter <= 7
89 -             if iter > 2
90 -                 if Qgc < Qmin(n),
91 -                     Vm(n) = Vm(n) + 0.01;
92 -                 elseif Qgc > Qmax(n),
93 -                     Vm(n) = Vm(n) - 0.01;end
94 -             else, end
95 -         else,end
96 -     else,end
97 - end
98 - if Kb(n) ~= 1
99 -     A(nn,nn) = J11; %diagonal elements of J1
100 -     DC(nn) = P(n)-Pk;
101 - end
102 - if kb(n) == 0
103 -     A(nn, lm) = 2*Vm(n)*Ym(n,n)*cos(t(n,n))+J22; %diagonal elements of J2
104 -     A(lm,nn)= J33; %diagonal elements of J3
105 -     A(lm, lm) =-2*Vm(n)*Ym(n,n)*sin(t(n,n))-J44; %diagonal of elements of J4
106 -     DC(lm) = Q(n)-Qk;
107 - end
108 - end
109 - DX=A\DC';
110 - for n=1:nbus
111 -     nn=n-nss(n);
112 -     lm=nbus+n-ngs(n)-nss(n)-ns;
113 -     if kb(n) ~= 1
114 -         delta(n) = delta(n)+DX(nn); end
115 -     if kb(n) == 0
116 -         Vm(n)=Vm(n)+DX(lm); end
117 - end
118 - maxerror=max(abs(DC));
119 - if iter == maxiter & maxerror > accuracy
120 -     fprintf('\nWARNING: Iterative solution did not converged after ')

```

```

121 - fprintf('%g', iter), fprintf(' iterations.\n\n')
122 - fprintf('Press Enter to terminate the iterations and print the results \n')
123 - converge = 0; pause, else, end
124
125 - end
126
127 - if converge ~= 1
128 -     tech= ('                ITERATIVE SOLUTION DID NOT CONVERGE'); else,
129 -     tech=('                Power Flow Solution by Newton-Raphson Method');
130 - end
131 - V = Vm.*cos(delta)+j*Vm.*sin(delta);
132 - deltad=180/pi*delta;
133 - i=sqrt(-1);
134 - k=0;
135 - for n = 1:nbus
136 -     if kb(n) == 1
137 -         k=k+1;
138 -         S(n)= P(n)+j*Q(n);
139 -         Pg(n) = P(n)*basemva + Pd(n);
140 -         Qg(n) = Q(n)*basemva + Qd(n) - Qsh(n);
141 -         Pgg(k)=Pg(n);
142 -         Qgg(k)=Qg(n);
143 -     elseif kb(n) ==2
144 -         k=k+1;
145 -         S(n)=P(n)+j*Q(n);
146 -         Qg(n) = Q(n)*basemva + Qd(n) - Qsh(n);
147 -         Pgg(k)=Pg(n);
148 -         Qgg(k)=Qg(n);
149 -     end
150 -     yload(n) = (Pd(n)- j*Qd(n)+j*Qsh(n))/(basemva*Vm(n)^2);
151 - end
152 - busdata(:,3)=Vm'; busdata(:,4)=deltad';
153 - Pgt = sum(Pg); Qgt = sum(Qg); Pdt = sum(Pd); Qdt = sum(Qd); Qsht = sum(Qsh);
154
155 %clear A DC DX J11 J22 J33 J44 Qk delta lk ll lm
156 %clear A DC DX J11 J22 J33 Qk delta lk ll lm
157

```

## Appendix C

### M-files for tutorial & examples GUI

```

1  function varargout = Example(varargin)
2
3  - gui_Singleton = 1;
4  - gui_State = struct('gui_Name',       mfilename, ...
5                    'gui_Singleton',  gui_Singleton, ...
6                    'gui_OpeningFcn', @Example_OpeningFcn, ...
7                    'gui_OutputFcn',  @Example_OutputFcn, ...
8                    'gui_LayoutFcn',   [] , ...
9                    'gui_Callback',    []);
10 - if nargin && ischar(varargin{1})
11 -     gui_State.gui_Callback = str2func(varargin{1});
12 - end
13
14 - if nargin
15 -     [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
16 - else
17 -     gui_mainfcn(gui_State, varargin{:});
18 - end
19 % End initialization code - DO NOT EDIT
20
21
22 % --- Executes just before Example is made visible.
23 function Example_OpeningFcn(hObject, eventdata, handles, varargin)
24
25 - movegui('center')
26 % Choose default command line output for Example
27 - handles.output = hObject;
28
29 % Update handles structure
30 - guidata(hObject, handles);
31
32 % UIWAIT makes Example wait for user response (see UIRESUME)
33 % uiwait(handles.figure1);
34
35
36 % --- Outputs from this function are returned to the command line.
37 function varargout = Example_OutputFcn(hObject, eventdata, handles)
38
39 % Get default command line output from handles structure
40 - varargout{1} = handles.output;
41
42 % --- Executes during object creation, after setting all properties.
43 function axes1_CreateFcn(hObject, eventdata, handles)
44
45 % Hint: place code in OpeningFcn to populate axes1
46
47 - [x,map] = imread('tower','jpg');
48 - image(x)
49 - set(gca,'visible','off')
50
51
52 % --- Executes on button press in pushbutton1.
53 function pushbutton1_Callback(hObject, eventdata, handles)
54
55 - figure(example3)
56
57 % --- Executes on button press in pushbutton2.
58 function pushbutton2_Callback(hObject, eventdata, handles)
59
60 - figure(example2)

```

```
61
62 % --- Executes on button press in pushbutton3.
63 function pushbutton3_Callback(hObject, eventdata, handles)
64
65 - Close
66
67 % --- Executes on button press in pushbutton4.
68 function pushbutton4_Callback(hObject, eventdata, handles)
69
70 - figure(Tutorial)
71
```

## Example.fig

