GRAPHICS-DRIVEN ELECTRIC POWER SYSTEMS PROGRAMS FOR THE PERSONAL COMPUTER

by

Kai Fat Wong

A Thesis
presented to the University of Manitoba
in partial fulfillment of the
requirements for the degree of
Master of Science
in
Department of Electrical Engineering

Winnipeg, Manitoba, 1988

ISBN 0-315-47853-5

GRAPHICS-DRIVEN ELECTRIC POWER SYSTEMS PROGRAMS FOR THE PERSONAL COMPUTER

BY

KAI FAT WONG

A thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

I hereby declare that I am the sole author of this thesis. I authorize the University of Manitoba to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Kai Fat Wong

I further authorize the University of Manitoba to reproduce this thesis by photocopying or by other means, in whole or in part, at the resquest of the other institutions or individuals for the purpose of scholarly research.

Kai Fat Wong

The University of Manitoba requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

ABSTRACT

This paper presents an educational power system software package designed for introductory power system analysis course. The software is called PC-POWER. It is an interactve and graphics-driven personal computer package. It can be run on an IBM PC, AT, XT or compatible personal computer with a CGA card or equivalent. The software is capable of solving transmission line representation, load flow, symmetrical three-phase fault, unsymmetrical faults and transient stability for predefined small power systems. The objective is for students to learn the principles and the characteristics of power system and not the numerical analysis. The software adopts small size predefined power systems to demonstrate power system principles since the principles are best learned by studying small systems.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to Professor G. W. Swift for suggesting this topic, and for his valuable guidance and advice throughout the development of the software package. I would also like to thank Manitoba Hydro for the financial support. Thanks are also due to Professor M. Z. Tranawecky for his useful comments.

TABLE OF CONTENTS

ABSTRACT iv

ACKNOWLEDGEMENT v

Chapter Page

I. INTRODUCTION 1

1.1 Objective 1
1.2 Background 1
1.3 Problems 3
1.4 Scope 5

II. DESCRIPTION OF THE PACKAGE 7

2.1 Selection of Programming Language 9
2.2 Hardware Requirements 11
2.3 Module Organization 11
2.4 Input/Output Facilities 14
2.5 Analytical Facilities 14

III. TRANSMISSION LINE REPRESENTATION 16

3.1 Input Requirements 17
3.2 Output Results 18
3.3 Analytical Method 20
1. The Short Line Representation 21
2. The Medium Line Representation 21
3. The Long Line Representation 22

VI. LOAD FLOW PROGRAM 24

4.1 Input Requirements 26
4.2 Output Results 27
4.3 Analytical Method 29

V. SYMMETRICAL THREE-PHASE FAULT PROGRAM 37

5.1 Input requirements 37
5.2 Output Results 38
5.3 Analytical Method 40

VI. UNSYMMETRICAL FAULTS PROGRAM 42

6.1 Input Requirements 43
6.2 Output Results 46
6.3 Analytical Method 46

VII. TRANSIENT STABILITY PROGRAM 52

7.1 Input Requirements 54
7.2 Output Results 56
7.3 Analytical Method 56

VIII. CONCLUSIONS 61

BIBLIOGRAPHY 63

Appendix

A. Listing of Source Programs 65

LIST OF FIGURES

Figure		Page
1.1	The One-Line Diagram of IEEE 39 Bus System	6
1.2	The Curves of Electrical Acceleration Angles of Generators 1-9	6
2.1	Flow Chart of the Module Organization	12
2.2	Flow Chart of Sub-Program	13
3.1	Single-Line Diagram of the Test System	16
3.2	Output Results of the Transmission Line Program	19
3.3	Voltage Profile of a Transmission Line	19
3.4	The Short Line Representation	21
3.5	The Medium Line Representation	22
4.1	The Two Bus System in the Load Flow Program	25
4.2	The Three Bus System in the Load Flow Program	25
4.3	Input Mode of the Two Bus System	28
4.4	Output Results of the Two Bus System	28
4.5	Flow Chart of the Newton-Raphson Method	36
5.1	Two Bus System of the 3-Phase Fault Program	39
5.2	Output Results of the 3-Phase Fault Program	39
6.1	Single-Line Diagram of the Power System in the Unsymmetrical Faults Program	42
6.2	Connections & Zero-Sequence Equivalent Circuits of Three-Phase Transformer Banks	44

6.3 Connection of the Hypothetical Stubs for Various Faults through Impedance 45

6.4 Input Mode of the Line-to-line Fault 47

6.5 Output Results of a Line-to-line Fault 47

6.6 Sequence Connections of Various Types of Faults Through Fault Impedance 49

7.1 One-Line Diagram of the Multimachine System 53

7.2 Input Mode of the Multimachine System 55

7.3 Output Plot of the Rotor Angles Vs Time Graph for the Two Machines 55

7.4 Flow Chart of the Transient Stability 60

LIST OF TABLES

Table Page

2.1 Benchmark Results : Run Time Comparisons of the Five Compilers(interpreter) for Computing 10,000 Times of Combinations of Transcendental Functions 10

4.1 Summary of Bus Type in the Load Flow Study 27

Chapter I

INTRODUCTION

1.1 Objective

The purpose of this research was to develop a user friendly and highly portable personal computer software package for power system study. The software package is to be used as an educational tool and as part of the laboratory facilities for a introductory power system analysis course.

1.2 Background

Computers have been used as powerful tools in the power engineering profession for a long time. The first special purpose analog computer, called an ac network analyzer was designed in 1929 to study the system operating conditions under the existing and proposed conditions.

In the 1950s, large scale digital computers provided sufficient capacity and speed to handle major power system problems. In 1957 the American Electric Power Services Corporation completed a large scale load flow program for the IBM

704. Because of the success of that program and the great technical advancement in the design and production of new digital computers all subsequent studies have employed a digital computer.(Stagg, pp.1-2)

The changes in the industry have direct impact on the educational environment. In the 1950s and 1960s, experiments were usually performed with a miniature system comprising signal generators, RLC components and an ac analyser.(Wachal, p.445) The set up time was excessively long and the measured results were less "pure" than those that appeared in textbooks.(Semlyen, p.2290)

In the 1970s computer assignments tended to replace the physical power system laboratories. However, the large scale computer often applied batch processing which tended to "move the emphasis from basic understanding of physical principles to numerical analysis." (Hatziargyriou, p.2296) Moreover, the results could not be seen immediately and the output format was so poor that the results could not be visualized as meaningful by students because of the lack of graphics ability. (Hatziargyriou, p.2296)

In the 1980s the great technical advances in the personal computer's production and design had made economically feasible for academic institutions to employ personal computers in their laboratories. Since then personal computer simulation software for power system studies were under development. Interactive systems were widely employed to eliminate the disadvantages of the batch processing common for large scale computers. Some of the programs even used graphics visual display to improve the man-machine interface.

1.3 Problems

Physical lab experiments are not adequate for a power system course because the set-up time is long and the models are hard to overview and handle. (Semlyen, p.2290) On the other hand, computer simulations can eliminate these problems since it does not require to set up any physical model, only mathematical equations are required. Therefore, it can very specifically model one phenomena without the interference of the others.

It is, however, very difficult to find a suitable software package for educational use in power system study especially for personal computer. Most pioneered research studies and developments have focused their attention toward utilities. Many of these can only be used in large scale computers or mini-computers. Therefore, these programs have very poor user-computer communication and are not highly portable.

Only a few educational purpose micro-computer programs have been developed recently. They use interactive programming techniques to improve the man-machine interface. Unfortunately, there is still no graphics-driven power system analysis software commercially available for personal computers in 1987, according to Norma J. Haakonstad, Sales Manager of the Electrocon International INC. (letter to Professor G.W. Swift). The software available on the market at that time was called the Electrocon IBM Personal Computer Distribution Primary Analysis System Program (DPAS). However, DPAS was not suitable for teaching purposes for the following reasons:

1. The preparation time was long. The student had to go through 200 pages of user's manuals and tutorials before he could manage the software.

2. The man-machine interface was poor. There was no graphical display to help the student visualize the whole system during the data acquisition process. Moreover, there was no error checking on each entry for syntax errors.

3. There was too much data and file transferring. The student could not go to the power system analysis programs directly. It was distracting.

4. The software was expensive. "The single-copy price for DPAS 4.0 is $5000" according to the IBM Personal Computer (PC) Series Price List - July 1986. (Electrocon International, Inc.) It was very difficult for a student

to afford one if he wanted to use it at home.

Some researches do include graphics visual display facilities to further improve the man-machine interface. However, their packages are too complicated for an undergraduate student. For example, Figure 1.1 and Figure 1.2 show the output from the Color Graphical Power System Simulator (GPSS) described by a paper presented in IEEE/PES 1988 Winter Meeting. (Yu) The system is too complex for an undergraduate student. The result plotting is confusing. Furthermore, it is not yet commercially available.

Therefore, it is necessary to design new software to meet the academic goals. Professor G. W. Swift has given concise objectives and specifications for the educational software from the beginning of this development.

1.4 Scope

This report contains 8 chapters and 1 appendix. Chapter II gives an overview of the package including the special features and the hardware requirements. Chapter III to Chapter VII are detailed descriptions of each program. The sequence is: transmission line representation, load flow, symmetrical three-phase fault, unsymmetrical faults and transient stability. Chapter VIII is the conclusion of the thesis. Appendix A contains the listing of each program.

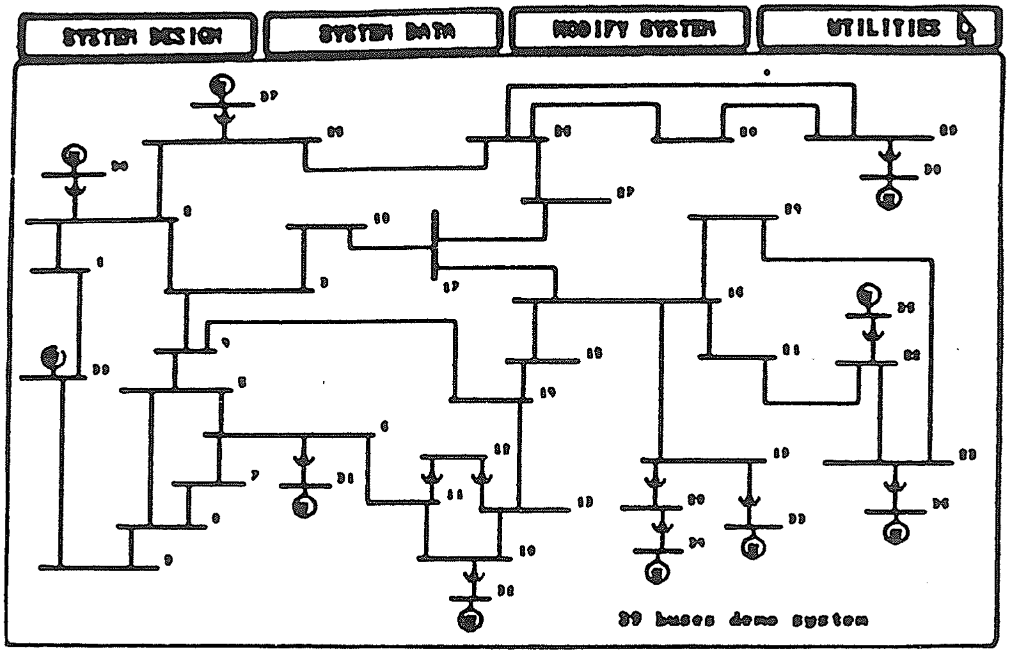


Figure 1.1 : The One-Line Diagram of IEEE 39 Bus System

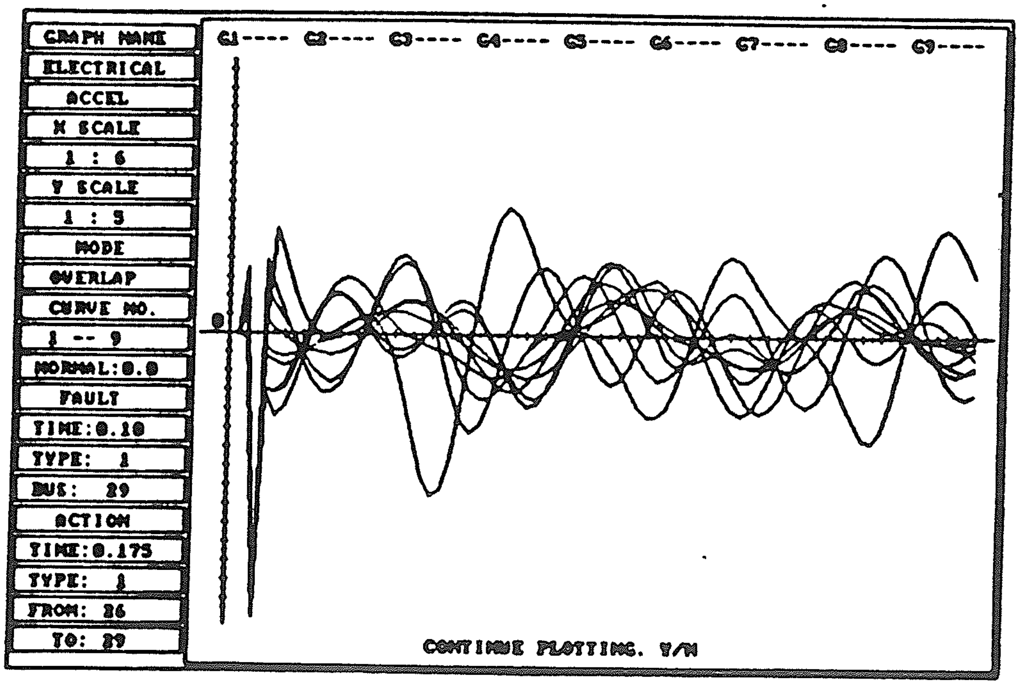


Figure 1.2 : The Curves of Electrical Acceleration Angles of Generator 1-9

Source for Fig. 1.1 and Fig. 1.2 : "A PC Oriented Interactive and Graphical Simulation Package for Power System Study." IEEE/PES 1988 Winter Meeting, New York, New York, January 31 - February 5, 1988

Chapter II

DESCRIPTION OF THE PACKAGE

The package, called PC-POWER, is an interactive, graphics-driven package. It provides a portable and user friendly environment for the power system study. The ultimate goal is to help the student understand the physical principles and the characteristics of power system. The package consists of a set of programs and a set of corresponding exercises. The programs include transmission line representation, load flow, symmetrical three-phase fault, unsymmetrical faults and transient stability studies for small power systems. Each program can be run independently according to the need. The corresponding exercises were written by Professor G. W. Swift. Easy-to-use features have been implemented in PC-POWER so that it can be easily fitted into the tight undergraduate curriculum without further overburdening the student's work load. In order to achieve the objectives, specifications were predetermined as follows:

1. The programs should be easy to use. Users should not be required to have any programming knowledge.

2. The programs should be graphics driven. They should have

graphical input and graphical output. Appropriate instructions should be clearly shown on the screen at all times.

3. The programs should not "hang-up" if special case data is entered (transmission line impedance equals zero, for example).

4. The programs should not accept data which would cause syntax errors during the data acquisition process.

5. The programs should use simple fixed configurations but should provide the ability for users to modify the system conditions easily.

6. The program should be portable. They should be easy to carry and should be able to run with a broad class of personal computers.

7. Corresponding exercises should be set up to assist the student to understand the physical principles rather than the numerical analysis.

8. The software should be affordable for the student.

This chapter describes the selection of a programming language, the operating requirements, and the input/output and analytical facilities.

2.1 Selection of Programming Language

After the specifications have been defined, the programming language should be chosen so that it can perform and produce the expected results. Under these circumstances, the following requirements should be met:

1. The language should have good graphics facilities since the whole package is graphics-driven.

2. The language should have plenty of arithmetic functions and high speed calculation performance since many arithmetic operations are always involved in obtaining answers from a set of linear or non-linear equations.

3. The software developed from that language should not be copyright protected since the package should be easily available to the student.

Five compilers (interpreters) were investigated: Quick Basic 3.0, GW-Basic, Turbo Pascal 3.01A, WATFOR-77, and Turbo C 1. It was found out that Quick Basic 3.0 and GW-Basic have the best graphics functions among the five compilers (interpreters). (Microsoft QuickBASIC 3.0 Manual), (GW-BASIC Interpreter User's Reference), (Turbo Pascal 3.01A Manual), (WATFOR-77 User's Guide),

(Turbo C 1 User's Guide)

In an arithmetic performance test, a test program computing 10,000 repetition of a combination of the transcendental functions, such as sines, cosines, logs, and exponentials, was implemented on the same computer (without math coprocessor) using the five compilers (interpreters). The benchmark results are shown in Table 2.1. The result shows that Quick Basic 3.0 has the fastest run time. Furthermore, the software developed from Quick Basic 3.0 does not have copyright restrictions. Therefore, it is the most suitable language for this research.

Table 2.1: Benchmark results. Run time comparisons Of the five compilers(interpreter) for computing 10,000 times of combinations of the transcendental functions, such as sines, cosines, logs, and exponentials functions.

Language	Run Time (seconds)
Quick Basic 3.0	164
GW-Basic (interpreter)	632
Turbo Pascal 3.01A	694
WATFOR-77	960
Turbo C 1	650

2.2 Hardware Requirements

PC-POWER is designed to run on an IBM PC, XT, AT or compatible personal computers using PC DOS or MS DOS version 2.0 or higher operating system. It requires 256K RAM and a CGA graphics card or equivalent. It needs a dot matrix printer if printing is required, but the GRAPHICS command on the DOS must be executed before using this software.

2.3 Module Organization

PC-POWER consists of 5 independent programs written in Quick-Basic 3.0 language. They are:

1. Transmission line representation
2. Load-flow analysis
3. Symmetrical three-phase fault study
4. Unsymmetrical fault study
5. Transient Stability analysis.

These programs are linked together by a main menu as shown in Figure 2.1. Each program contains a set of concise predefined power systems which are designed to demonstrate a particular principle. They can be run according to the student's choices. Figure 2.2 shows the structure of each program. Each program has its own input/output subroutines and analytical procedures. Therefore, each of them can be run independently. This module structure provides good maintainability and adaptability as well as run time efficiency for the software.

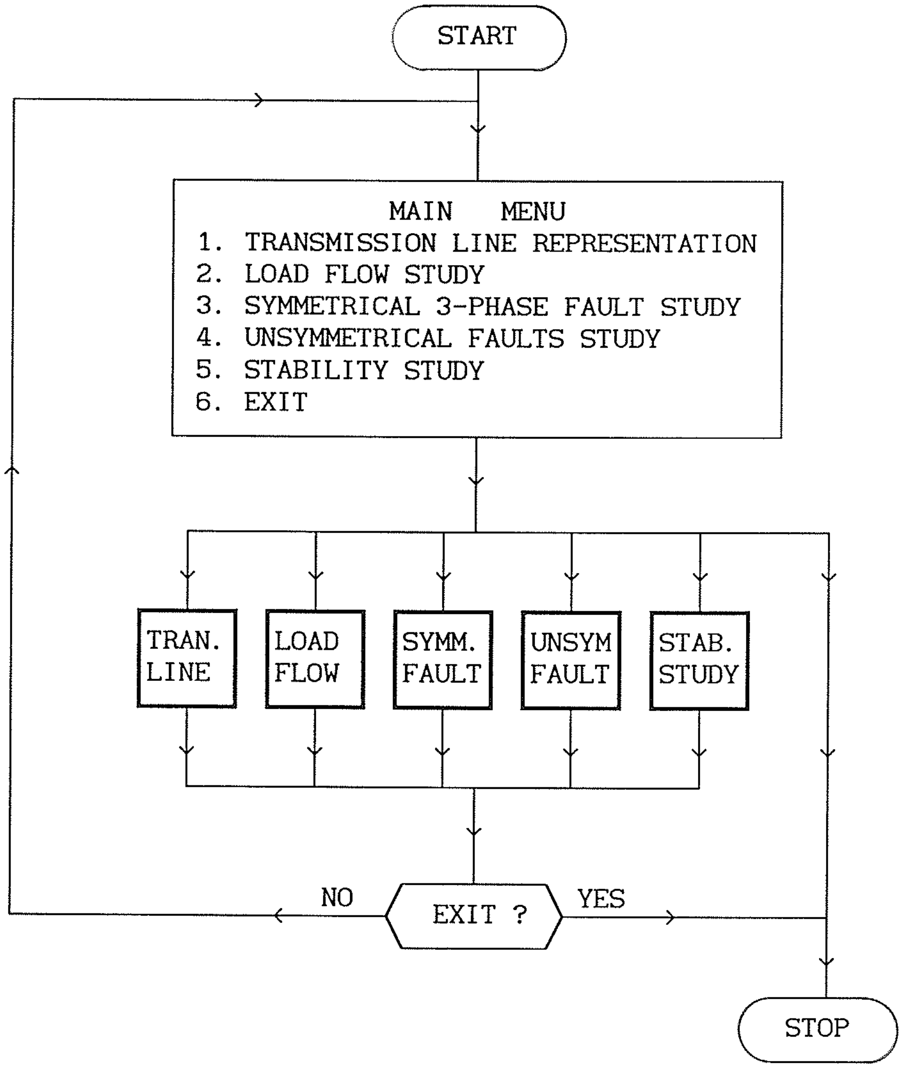


Figure 2.1 : Flow Chart of the Module Organization

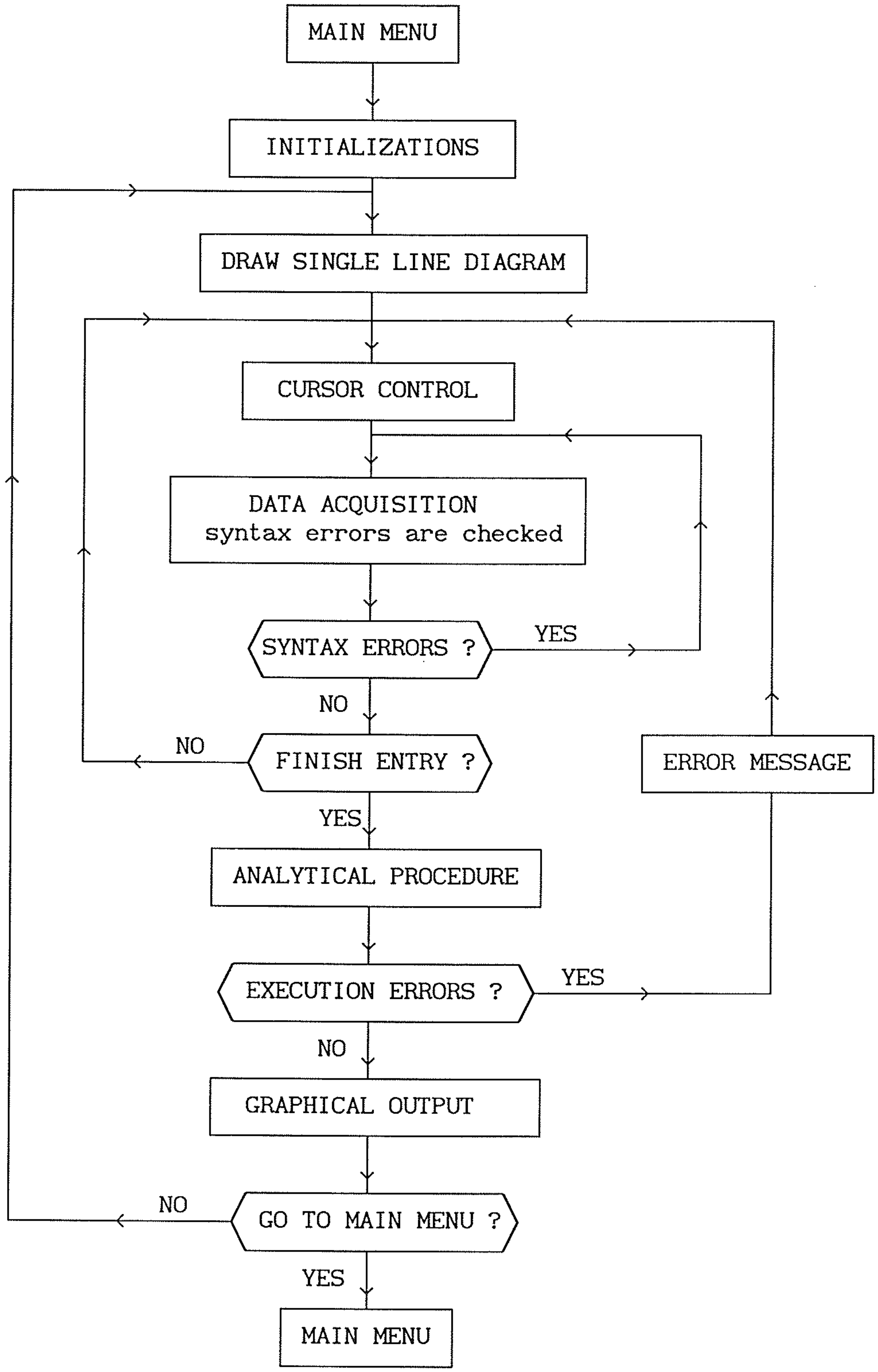


Figure 2.2 : Flow Chart of Sub-Program

2.4 Input/Output Facilities

The key feature of this interactive software is the graphics visual display facility. This feature promotes the learning process immensely. It is used in the data acquisition process. A drawing of a single-line diagram of a predefined power network is shown on the screen. Data is entered at the network elements on the single-line diagram by selecting an appropriate position on the diagrams using a cursor and typing in an appropriate data. Every input digit is checked on entry for syntax errors and will be disregarded when an error is detected. This checking can effectively eliminate most typing errors. On-screen instructions are also shown at the bottom of the diagram to guide the student. The output is also displayed pictorially so that the results can be visualized more easily and meaningfully. The above features are implemented to achieve a user friendly environment for the student.

2.5 Analytical Facilities

Fast solution time is important in an interactive environment. There are two ways to improve the calculation speed. The first way is to use a high-performance compiler. However, in this case, it should also provide good graphics functions. This has been considered in section 2.1. The second way is to choose fast and reliable algorithms to tackle the problems, since "in most of the cases, the mathematical model (ie. mathematical equations) of the power system problems, are a set of linear or nonlinear algebraic equations and or differential equations" (Singh, p.2). Solutions cannot usually be obtained directly and numerical approaches must be applied. Chapter III to Chapter VII will describe the data requirements and the solution techniques for each program in detail. Besides the speed, PC-POWER also handles execution errors caused by overflow or underflow during the calculations to prevent abnormal termination. If an error occurs, the programs will prompt error messages and will direct the student to go back to the input mode.

Chapter III

TRANSMISSION LINE REPRESENTATION

The objective of this program is to compare the accuracy of three transmission line representations, namely the short, medium and long transmission line representation, for an overhead line. The single-line system shown in Figure 3.1 is used to investigate the accuracy of the three transmission line models. Figure 3.1 is a single-line diagram representation of a typical 3-phase system where the generator supplies a balanced-Y load through a 3-phase transmission system.

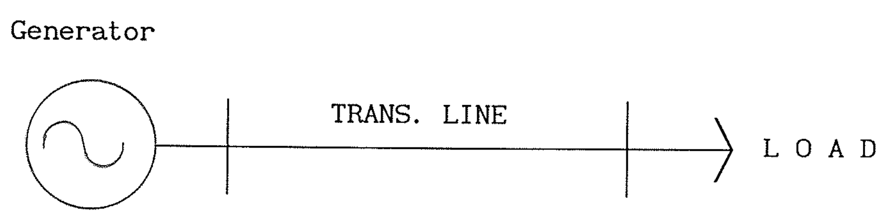


Figure 3.1 : Single-Line Diagram of the Test System

3.1 Input Requirements

The performance of an overhead line is determined by three distributed parameters: series resistance, series inductance, and shunt capacitance. Shunt conductance is neglected because the line loss due to leakage current between conductors or between conductors and ground is very small. (Glover p.135)

The performance is usually measured in terms of efficiency and regulation of the line.

The efficiency of a line is defined as:

$$\% \text{ efficiency} = \frac{\text{Power delivered at the receiving end}}{\text{Power sent from the sending end}} * 100$$

The mathematical expression for the regulation of a transmission line is:

$$\% \text{ regulation} = \frac{|V_{RNL}| - |V_{RFL}|}{|V_{RFL}|} * 100 \qquad (3.1)$$

where $|V_{RNL}|$ = the magnitude of the no-load receiving end voltage

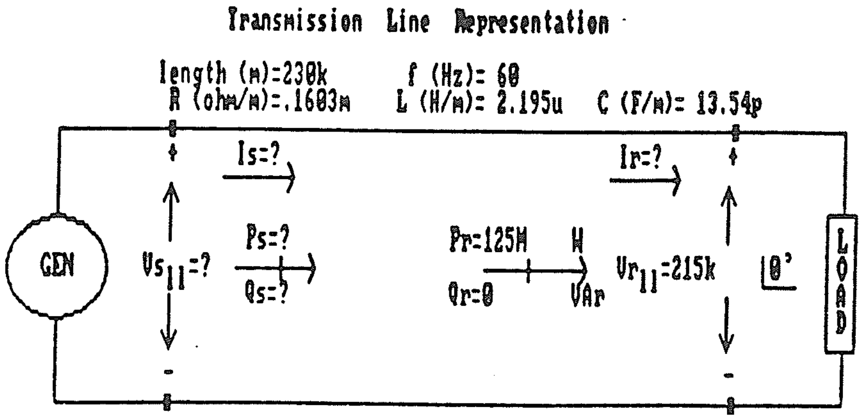
$|V_{RFL}|$ = the magnitude of the full-load receiving end voltage

Normally, the receiving end voltage, power, operating frequency and line length of a given system are known. The sending end voltage, power, and the performance of the line are then determined. Therefore, the data required in this program are:

1. Series resistance, series inductance, shunt capacitance, and the length of the transmission line,
2. Voltage and power at the receiving end, and
3. Operating frequency of the system.

3.2 Output Results

In order to compare the accuracy of the three models, sending end voltage, current and power will be calculated using the mathematical formulations of the three models. The results are printed in a tabular form for easy comparison as shown in Figure 3.2. A voltage profile showing that the voltage varies along the line can also be generated easily if the student is interested in it. Figure 3.3 is a computer printout of a voltage profile of a transmission line.



	Short Line Rep.	Medium line Rep.	Long Line Rep.
Ps (MW)	137.46	138.05	137.36
Qs (MVAr)	64.333	5.9395	3.4396
Vsll (kV)	261.05 ∠ 25.080°	241.69 ∠ 28.495°	238.75 ∠ 27.761°
Is (A)	335.67 ∠ .00000°	330.07 ∠ 26.031°	332.27 ∠ 26.327°
Ir (A)	335.67 ∠ .00000°	335.67 ∠ .00000°	335.67 ∠ .00000°
γ (/km)			.00020 +j .00206

<P>: plot the voltage profile, or other keys to the input mode ...

Figure 3.2 : Output Results of the Transmission Line Program

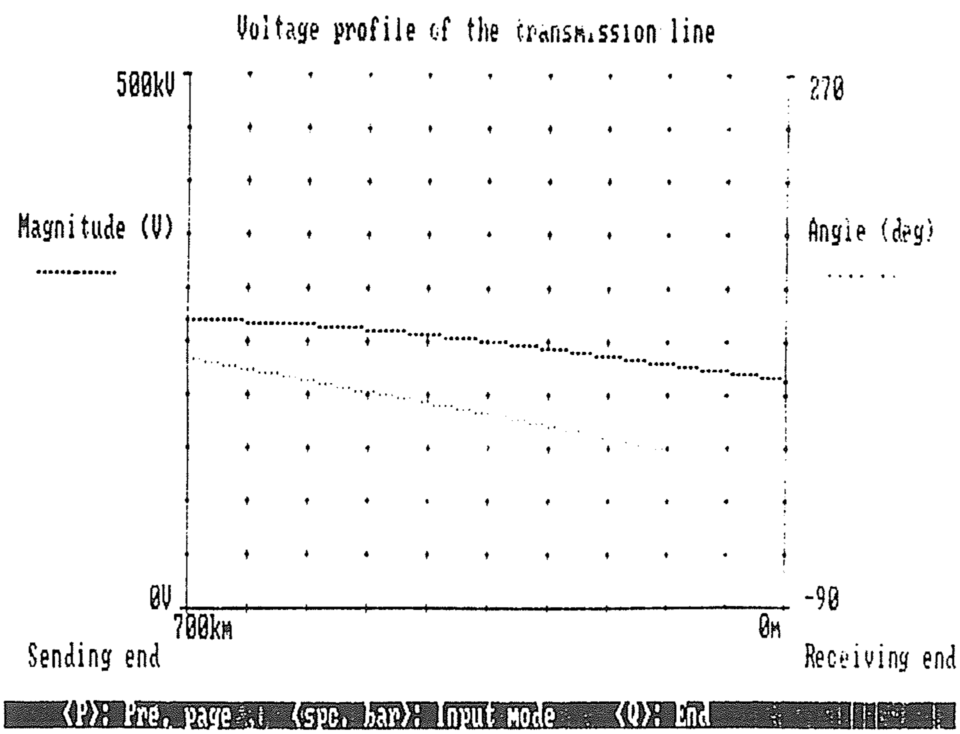


Figure 3.3 : Voltage Profile of a Transmission Line

3.3 Analytical Method

The short and medium line models assume lumped parameters for actually uniformly distributed parameters. The short line model further neglects the effect of the shunt capacitance. Although the assumptions simplify the calculations, they can only be applied for line less than 240km with good accuracy for a 60Hz system. (Stevenson pp.90-91) The most accurate model is the long line representation since it models the parameters uniformly along the line.

The following nomenclature is adopted in the discussion of this chapter:

z = series impedance per unit length

y = shunt admittance per unit length

l = length of the line

$Z = zl$ = total series impedance

$Y = yl$ = total shunt admittance

V_S = sending end line-to-neutral voltage

V_R = receiving end line-to-neutral voltage

I_S = sending end line current

I_R = receiving end line current

ω = operating frequency in radian

1. <u>The Short Line Representation</u>

The equivalent circuit of a short line is shown in Fig. 3.4. Only the series impedance is included.

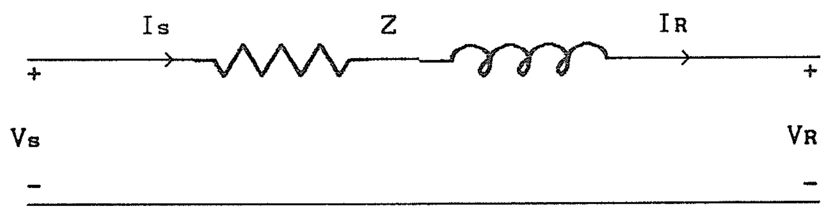


Figure 3.4 : The Short Line Representation

The relationship between the sending end and the receiving end quantities are:

$$V_S = V_R + I_R Z \qquad (3.2)$$

$$I_S = I_R \qquad (3.3)$$

2. <u>The Medium Line Representation</u>

The medium line can be represented by either nominal-T or nominal-Π representation. This program uses the nominal-Π representation as shown in Fig. 3.5. Half of the total shunt capacitance is lumped and located at each end of the line.

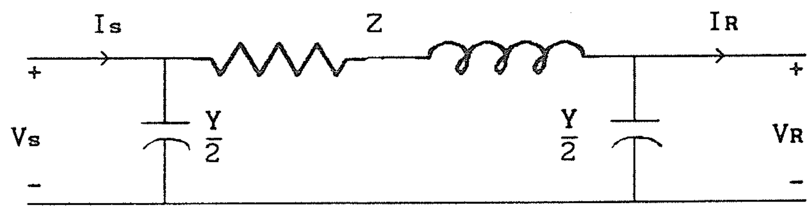


Figure 3.5 : The Medium Line Representation

The sending end voltage and current can be written as:

$$V_S = (\frac{Z\,Y}{2} + 1)\ V_R + Z\ I_R \qquad (3.4)$$

$$I_S = Y\ (\frac{Z\,Y}{4} + 1)\ V_R + (\frac{Z\,Y}{2} + 1)\ I_R \qquad (3.5)$$

3. <u>The Long Line Representation</u>

The long line representation is the most accurate model and it is also the most difficult one to calculate because it distributes the parameters evenly along the line. The mathematical expressions for the sending end voltage and current in terms of the receiving end voltage and current are:

$$V_S = V_R \cosh(\gamma l) + I_R Z_C \sinh(\gamma l) \qquad (3.6)$$

$$I_S = \frac{V_R}{Z_C} \sinh(\gamma l) + I_R \cosh(\gamma l) \qquad (3.7)$$

where $\gamma = \sqrt{y\,z}$ m^{-1}

$Zc = \sqrt{z/y}$ Ω

γ and Zc are called the propagation constant and the characteristic impedance, respectively.

The voltage profile of the line is generated using the following equations:

$$V = V_R \cosh(\gamma x) + I_R Zc \sinh(\gamma x) \qquad (3.6)$$

where x = the distance measured from the receiving end of the line

Chapter IV

LOAD FLOW PROGRAM

Load flow calculations determine the steady-state operating conditions of a power system. A load flow solution provides the voltage magnitude and phase angle at each bus, and hence the real and reactive power flow in each line. The information is essential for evaluating and improving the performance of an existing power system, and for designing a new power system. Load flow solutions also give the initial conditions for other studies such as stability studies. The mathematical formulation of the load flow problem is a system of nonlinear algebraic equations. Therefore, numerical techniques are needed in order to solve the equations using a digital computer.

There are both a two bus and a three bus systems in this program. The two bus system has two generators supplying a load. Generator 2 and the load are both connected to bus 2 and, generator 1 is connected to bus 1. Bus 1 and bus 2 are connected together through a three-phase transmission line.

The three bus system has two generators supplying two loads in the following connection. Generator 1 is connected to the

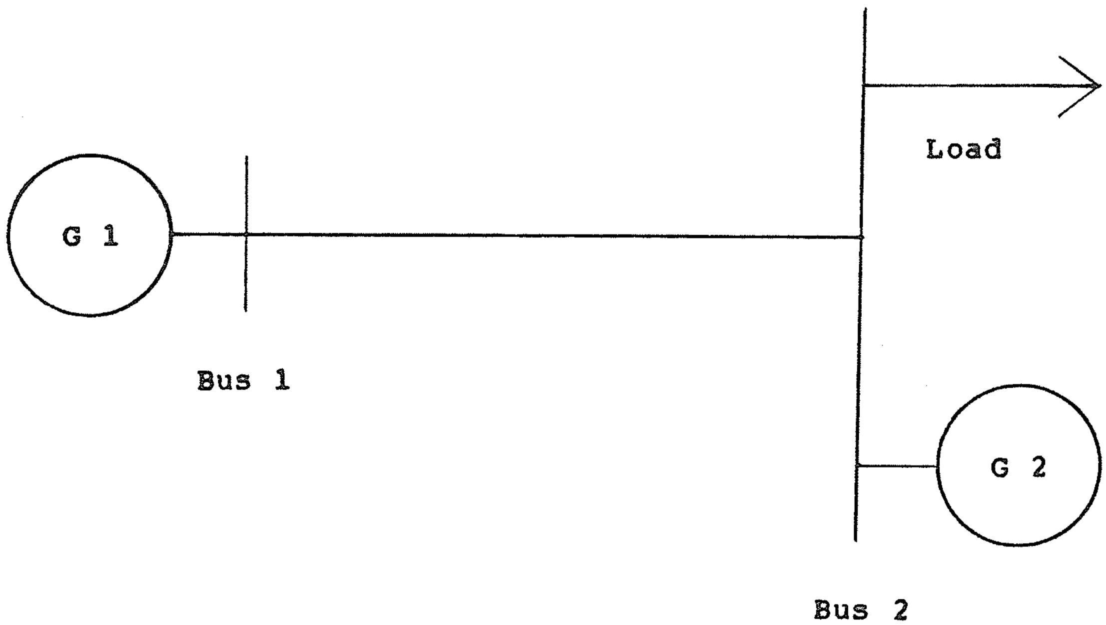


Figure 4.1 : The Two Bus System in the Load Flow Program

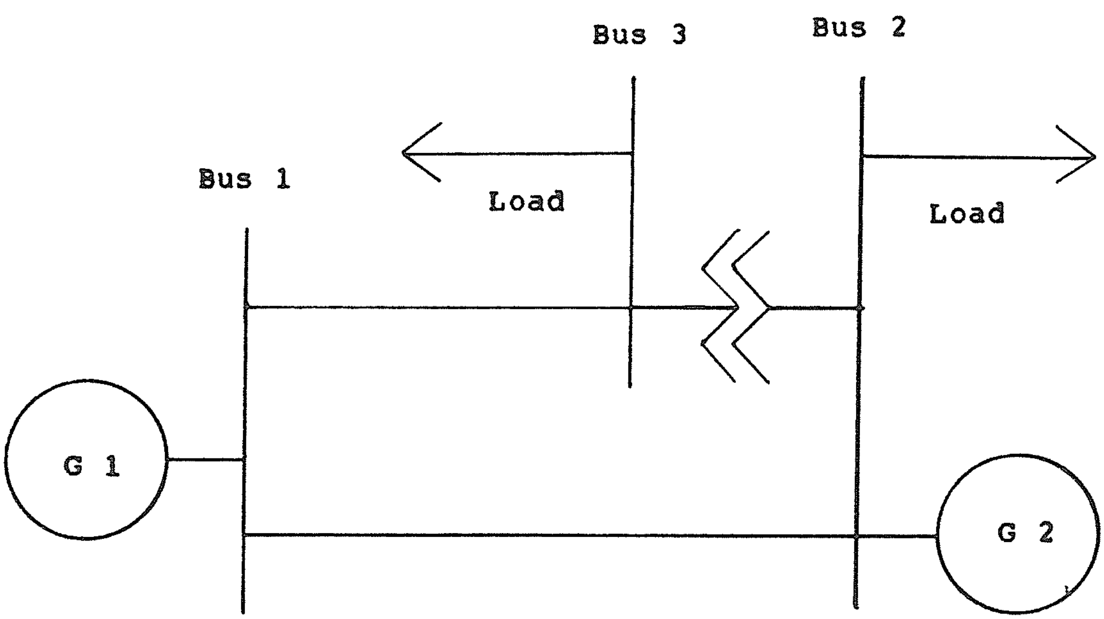


Figure 4.2 : The Three Bus System in the Load Flow Program

swing bus (bus 1). Generator 2 and one load are connected to bus 2. The other load is connected to bus 3. Bus 1 is connected to bus 2 and bus 3 through a three-phase transmission line. Bus 2 and bus 3 are connected through a tapped changing transformer. The systems are shown in Figure. 4.1 and Figure. 4.2, respectively.

4.1 Input Requirements

Input data consist of bus, transmission line and transformer data. Similar to the transmission line representation program, the input data are entered above the elements of the predetermined single-line diagrams but using per-unit quantities instead. Figure 4.3 shows the input mode of the two bus system.

Each bus in this study is associated with four quantities: voltage magnitude $|V|$, its phase angle δ, and real and reactive power $P+jQ$. Two of these quantities are specified as input data and the remaining two quantities are to be computed by the program. The buses are classified according to the quantities specified. Table 4.1 summarizes the three types of buses in the load flow study.

Bus Type	Quantities	
	Specified	Unknown
Load bus	P, Q	$\|V\|$, δ
Generator Bus (Voltage controlled bus)	P, $\|V\|$	Q, δ
Swing Bus (slack bus)	$\|V\|$, δ	P, Q

Table 4.1 : Summary of Bus Type in the Load Flow Study

In this program, a transmission line is represented by an equivalent Π circuit. Therefore, the required data for a transmission line is the series impedance and shunt capacitance. The transformer data includes an off-nominal turns ratio and a series "leakage" winding inductance.

If the student purposely enters a very large series impedance for a line, say 100000 pu, and sets zero to its shunt capacitance, the program can simulate the event of losing a line. Therefore, contingency studies can also be done with this program.

4.2 Output Results

As mentioned earlier in this chapter, the program will compute the magnitude of voltage and phase angle for each bus, and the real and reactive power flow in each line. Results will be shown on a single-line diagram with the input data shown at the

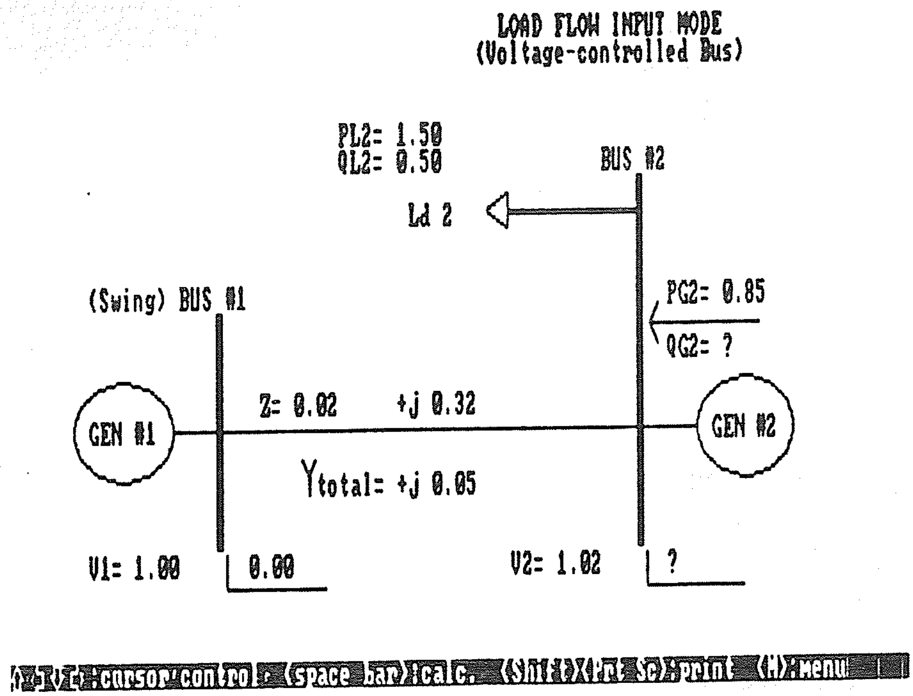


Figure 4.3 : Input Mode of the Two Bus System

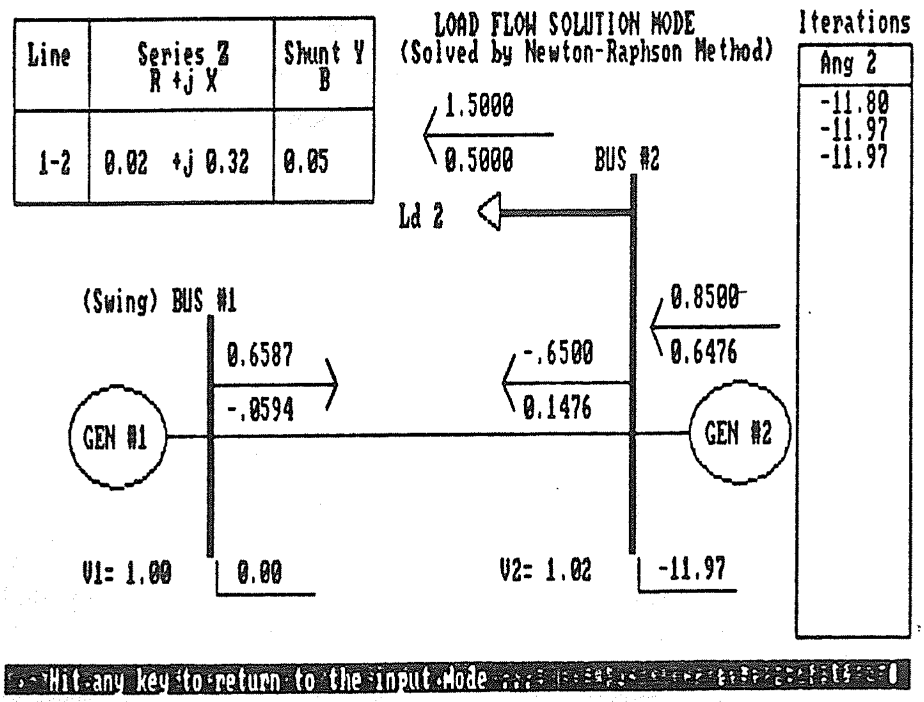


Figure 4.4 : Output Results of the Two Bus System

upper left corner of the diagram to provide complete information for the student. An example of a output results for the two bus system is shown in Figure 4.4.

In the ideal situation, the total real power and reactive power entering a bus will be equal to the real power and reactive power leaving the same bus. However, due to the round-off error of the computer and the unavoidable error in the numerical methods such as truncation error, a very small difference in the total powers entering and leaving a bus, in some cases, does appear in the output.

4.3 Analytical Method

The system under consideration is assumed to be operating in a balanced steady-state condition. The analysis requires mainly the following two steps:

1. Formulation of the network equations

   In this step, the bus admittance matrix Y_{bus} and the power equations are found. Firstly, the swing bus of an N bus system is taken as reference. Secondly, the (N-1) bus admittance matrix is assembled as follows:

   a. The diagonal elements:

      Y_{kk} = sum of admittances connected to bus k

b. The off-diagonal elements:

Y_{kn} = -(sum of admittances connected between bus k and n), $k \neq n$

Finally, the power equations are derived as follows:

$$P_k - jQ_k = \sum_{n=1}^{N} |V_k V_n Y_{kn}| \angle \theta_{kn} + \delta_n - \delta_k \qquad (4.1)$$

$$\therefore \quad P_k = \sum_{n=1}^{N} |V_k V_n Y_{kn}| \cos(\theta_{kn} + \delta_n - \delta_k) \qquad (4.2)$$

$$Q_k = -\sum_{n=1}^{N} |V_k V_n Y_{kn}| \sin(\theta_{kn} + \delta_n - \delta_k) \qquad (4.3)$$

where $V_k = |V_k| \angle \delta_k$ = voltage at bus k

$V_n = |V_n| \angle \delta_n$ = voltage at bus n

$Y_{kn} = |Y_{kn}| \angle \theta_{kn}$ = (k,n) element of Y_{bus}

P_k = total real power entering bus k

Q_k = total reactive power entering bus k

2. Numerical technique to solve the set of equations

This program uses the Newton-Raphson iterative method to solve the set of power equations. The method is fast and reliable. Moreover, the method can be easily modified to the decoupled Newton method or the "fast decoupled" Newton method.

The Newton-Raphson method is based on the Taylor's series expansion for a function of two or more variables. The object is to find out the voltage magnitude and phase angle for unknown buses and use the power flow to test whether the results are acceptable or not. The method is summarized in the following steps:

a. Guess the initial values for the unknown buses voltages and phase angles, setting them to the swing bus value, and start the iteration counter.

b. Calculate the power mismatch:

ΔP_k for generator bus,

ΔP_k and ΔQ_k for load bus.

where

$$\Delta P_k = P_{k,spec} - P_{k,calc} \quad (4.4)$$

$$\Delta Q_k = Q_{k,spec} - Q_{k,calc} \quad (4.5)$$

Both $P_{k,spec}$ and $Q_{k,spec}$ are specified as the input data and defined as the real and reactive powers entering bus k. $P_{k,calc}$ and $Q_{k,calc}$ are the values calculated from the power equations (4.2) and (4.3) with the guessed voltage magnitude and phase angle.

c. Form the Jacobian matrix J.

$$J = \left[\begin{array}{c|c} J_1 & J_2 \\ \hline J_3 & J_4 \end{array}\right] \qquad (4.6)$$

$$= \left[\begin{array}{ccc|ccc} \frac{\partial P_2}{\partial \delta_2} & \cdots & \frac{\partial P_2}{\partial \delta_N} & \frac{\partial P_2}{\partial |V_2|} & \cdots & \frac{\partial P_2}{\partial |V_N|} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial P_N}{\partial \delta_2} & \cdots & \frac{\partial P_N}{\partial \delta_N} & \frac{\partial P_N}{\partial |V_2|} & \cdots & \frac{\partial P_N}{\partial |V_N|} \\ \hline \frac{\partial Q_2}{\partial \delta_2} & \cdots & \frac{\partial Q_2}{\partial \delta_N} & \frac{\partial Q_2}{\partial |V_2|} & \cdots & \frac{\partial Q_2}{\partial |V_N|} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial Q_N}{\partial \delta_2} & \cdots & \frac{\partial Q_N}{\partial \delta_N} & \frac{\partial Q_N}{\partial |V_2|} & \cdots & \frac{\partial Q_N}{\partial |V_N|} \end{array}\right]$$

Where

$$\frac{\partial P_k}{\partial \delta_k} = \sum_{\substack{n=1 \\ n\neq k}}^{N} |V_k V_n Y_{kn}| \sin(\theta_{kn} + \delta_n - \delta_k) \qquad (4.7)$$

$$\frac{\partial P_k}{\partial |V_k|} = |V_k V_{kk}| \cos(\theta_{kk}) + \sum_{n=1}^{N} |Y_{kn} V_n| \cos(\theta_{kn} + \delta_n - \delta_k) \qquad (4.9)$$

$$\frac{\partial Q_k}{\partial \delta_k} = \sum_{\substack{n=1 \\ n\neq k}}^{N} |V_k V_n Y_{kn}| \cos(\theta_{kn} + \delta_n - \delta_k) \qquad (4.10)$$

$$\frac{\partial Q_k}{\partial |V_k|} = -|V_k V_{kk}| \sin(\theta_{kk}) - \sum_{n=1}^{N} |Y_{kn} V_n| \sin(\theta_{kn} + \delta_n - \delta_k) \qquad (4.11)$$

$$\frac{\partial P_k}{\partial \delta_n} = -|V_k V_n Y_{kn}| \sin(\theta_{kn} + \delta_n - \delta_k), \quad (n \neq k) \tag{4.12}$$

$$\frac{\partial P_k}{\partial |V_n|} = |V_k Y_{kn}| \cos(\theta_{kn} + \delta_n - \delta_k), \quad (n \neq k) \tag{4.13}$$

$$\frac{\partial Q_k}{\partial \delta_n} = -|V_k V_n Y_{kn}| \cos(\theta_{kn} + \delta_n - \delta_k), \quad (n \neq k) \tag{4.14}$$

$$\frac{\partial Q_k}{\partial |V_n|} = -|V_k Y_{kn}| \sin(\theta_{kn} + \delta_n - \delta_k), \quad (n \neq k) \tag{4.15}$$

d. Use Gauss elimination and back substitution to solve (4.16) and find the magnitude and phase correction (ΔV & $\Delta\delta$) of the unknown buses voltages.

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \left[\begin{array}{c|c} J_1 & J_2 \\ \hline J_3 & J_4 \end{array}\right] \begin{bmatrix} \Delta \delta \\ \Delta V \end{bmatrix} \tag{4.16}$$

e. Estimate the new bus voltages as follow:

$$\begin{bmatrix} \delta_{i+1} \\ V_{i+1} \end{bmatrix} = \begin{bmatrix} \delta_i \\ V_i \end{bmatrix} + \begin{bmatrix} \Delta \delta_i \\ \Delta V_i \end{bmatrix} \tag{4.17}$$

f. Go to step b. and continue the process until all the power mismatches are less than the predefined tolerance (0.0001) or until the iteration is greater than predetermined number (20) in this program. If convergence is obtained, then the answers are accepted and the power flow at each line is calculated. Otherwise, the program will print a message indicating that no acceptable answer is obtained.

The flow chart of the Newton-Raphson algorithm is shown in Figure 4.5.

The real power is mostly affected by the voltage phase angle and the reactive power is greatly affected by the voltage magnitude at the bus. The rate of change of real power with respect to voltage magitude and the rate of change of reactive power with respect to phase angle are very small and can be ignored. The decoupled Newton method, therefore, sets J_2 and J_3 in the Jacobian matrix equal to zero i.e. $\frac{\partial P}{\partial |V|} = 0$, $\frac{\partial Q}{\partial \delta} = 0$. The set of equations in (4.16) can be decoupled to:

$$[\ \Delta P\ ] = [\ J_1\ ]\ [\ \Delta\delta\ ] \quad (4.18)$$

$$[\ \Delta Q\ ] = [\ J_4\ ]\ [\ \Delta V\ ] \quad (4.19)$$

A lot of time can be saved for solving (4.18) and (4.19) compared to (4.16) for a large system.

The process can be speeded up even further, if the program assumes all buses voltage magnitudes are very close to the swing bus value and bus phase angles are almost the same. Or, mathematically:

$V_k \approx V_n \approx 1.0$ pu and

$\delta_k \approx \delta_n$

Then the system of equations can be simplified to:

$$[\ \Delta P/|V|\ ] = [\ B'\ ]\ [\ \Delta\ \delta\ ] \qquad (4.20)$$

$$[\ \Delta Q/|V|\ ] = [\ B''\ ]\ [\ \Delta\ V\ ] \qquad (4.21)$$

where [B'] and [B"] are parts of the imaginary components of Y_{bus}. Both [B'] and [B"] are unchanged throughout the iterations. Therefore, they need to be calculated once at the beginning.

However, if the student enters unrealistic data, such as a highly resistive transmission line and different voltages at different buses, then the above assumptions are no longer valid. The methods may not converge and results cannot be found. Therefore, they are not used here.

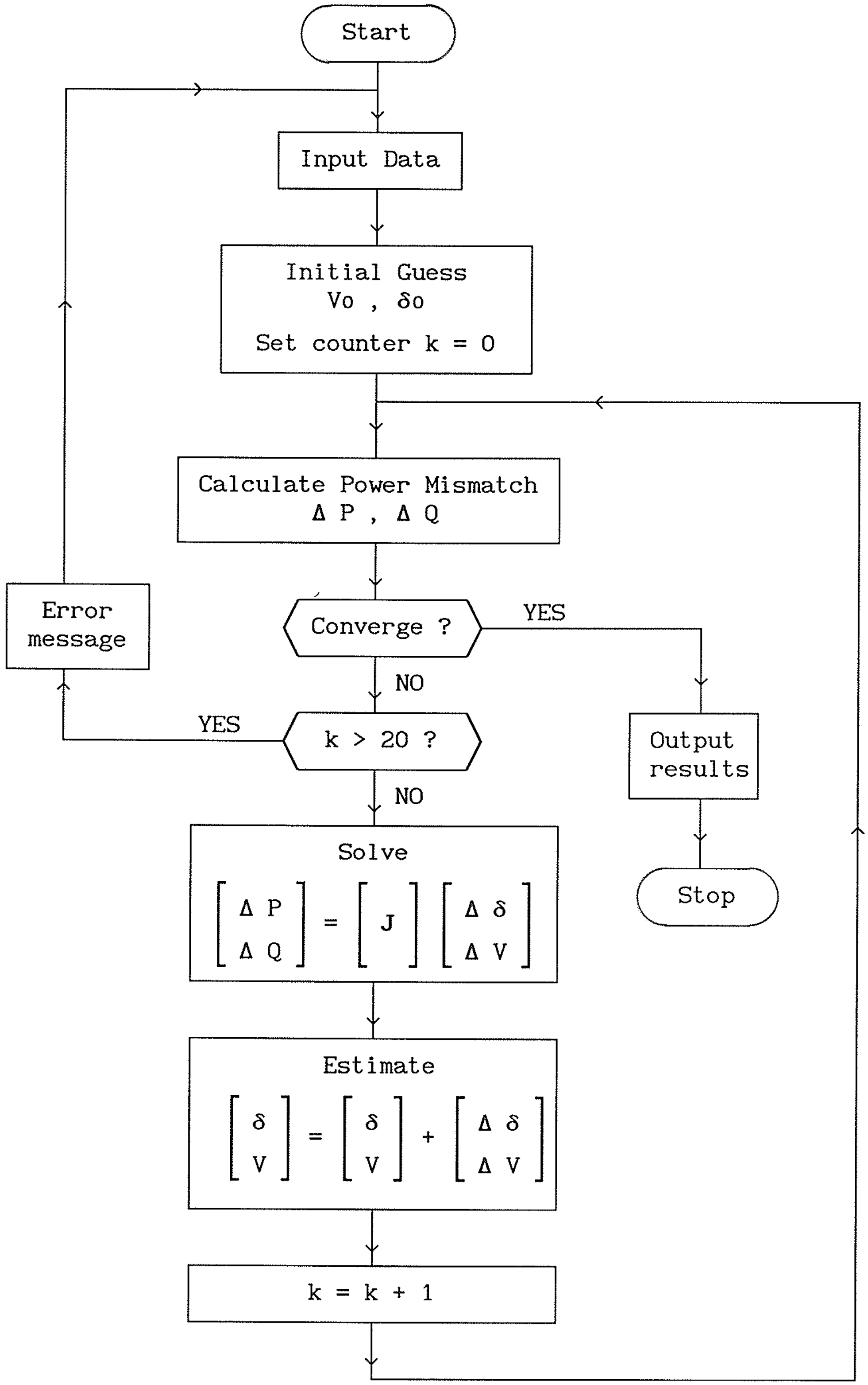


Figure 4.5 : Flow Chart of the Newton-Raphson Method

Chapter V

SYMMETRICAL THREE-PHASE FAULT PROGRAM

A symmetrical three-phase fault study provides fault current, current distribution and voltages on a power system during fault conditions. This information is important in designing an adequate protective scheme.

The symmetrical three-phase fault program accepts an unloaded bolted three-phase fault at any bus or anywhere along a line of a predefined two bus system. The system has two generators connected together through two parallel three-phase transmission lines. The one-line diagram of the power system is shown in Figure 5.1.

5.1 Input Requirements

The most severe but the least common fault is the symmetrical three-phase fault. The system remains balanced after the fault. Therefore, single-phase representation of the power system is sufficient in the study. This program adopts per unit quantities in the input and output.

The following assumptions are made to reduce the input data and simplify the calculation:

1. Each machine is represented by a constant voltage source in series with a constant subtransient reactance.

2. All the static loads are neglected and all prefault bus voltages are set at $1\angle 0^{\circ}$.

3. Transmission lines are represented by equivalent Π circuits.

Because of the assumptions, the input data is reduced to:

1. The location of the fault.
2. Positive sequence impedance and shunt capacitance of transmission line.
3. The subtransient reactance for each machine.

5.2 Output Results

The program calculates the fault current, the current distribution and the bus voltage during the fault. The results are printed at the appropriate locations on the same single-line diagram. The input data will also be shown at the upper left

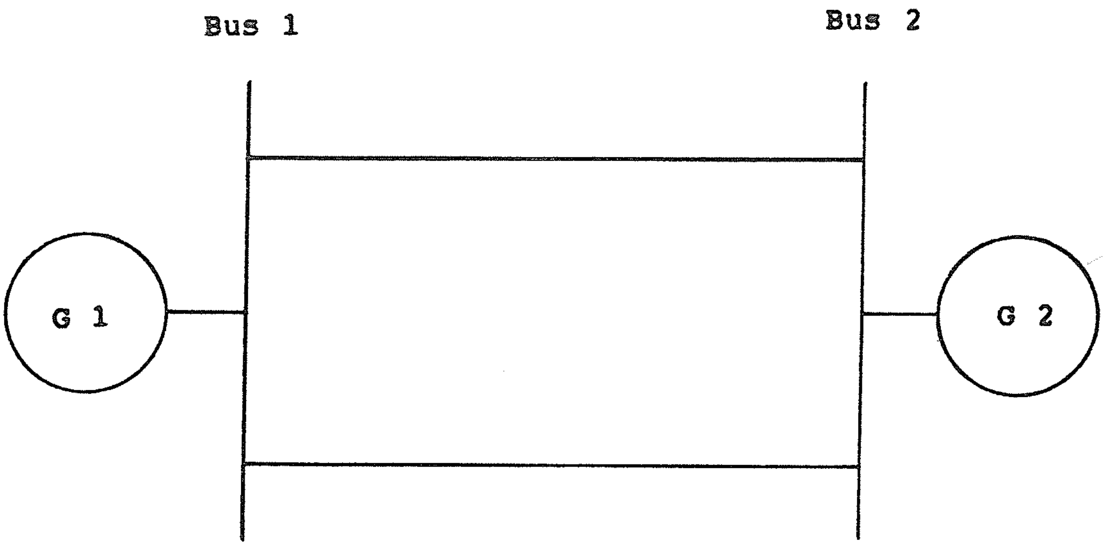


Figure 5.1 : Two Bus System of the 3-Phase Fault Program

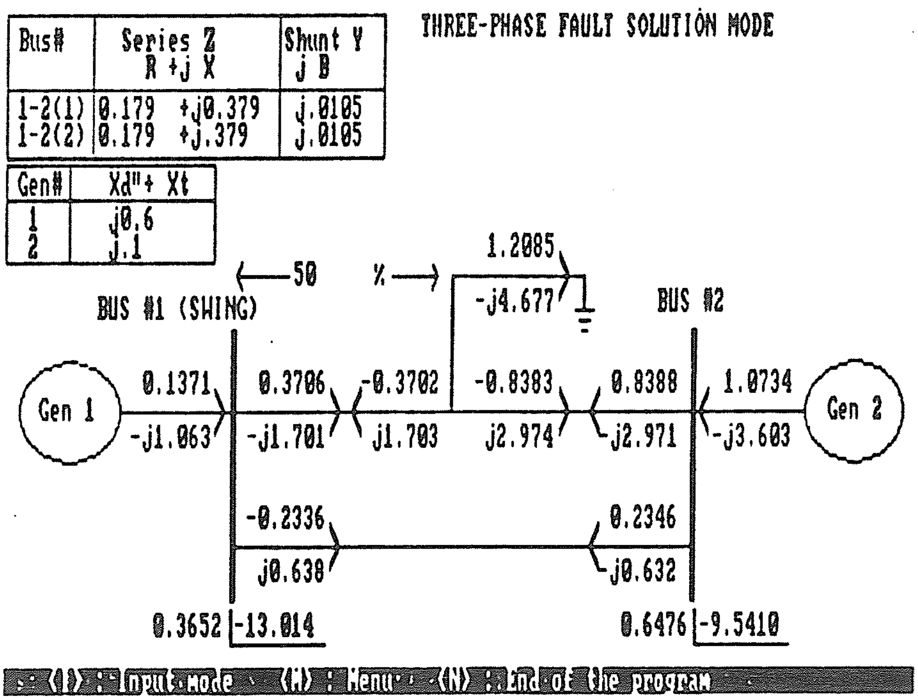


Figure 5.2 : Output Results of the 3-Phase Fault Program

corner of the screen. Figure 5.2 is a computer printout of a output results of the program.

5.3 Analytical Method

The method used in this program can be summarized in the followings. Firstly, assemble the bus admittance matrix Y_{bus} as described in the load flow program. The machine's reactance has to be included. Secondly, find the inverted matrix called the bus impedance matrix Z_{bus}. Finally, determine the fault current by the following equations (Stevenson, p. 262) assuming that the fault is at bus n:

$$I_{fault}(n) = \frac{V_p(n)}{Z_{bus}(n,n)} \qquad (5.1)$$

where $I_{fault}(n)$ = the total fault current at bus n

$V_p(n)$ = the prefault voltage of bus n

$Z_{bus}(n,n)$ = (n,n) element of Z_{bus}

Then, calculate the voltage for each bus by the equation:

$$V(k) = V_p(k) - \frac{Z_{bus}(k,n)}{Z_{bus}(n,n)} V_p(n) \qquad (5.2)$$

where $V(k)$ = the postfault voltage on bus k

$V_p(k)$ = the prefault voltage on bus k

$V_p(n)$ = the prefault voltage on bus n

Finally, find the current distribution on each transmission line by the equation:

$$I_{j-k} = \frac{V(j) - V(k)}{Z_{(j-k)}} \qquad (5.3)$$

where I_{j-k} = the postfault current between bus j and k

$V(j)$, $V(k)$ = the postfault voltage of bus j and k

$Z_{(j-k)}$ = the impedance between bus j and k

However, there is a restriction in using this program. The student should not enter zero or very small number for series impedance, say 0+j0 or 0+j0.00001, for a transmission line because the bus admittance matrix formed by the computer will be a singular or very close to a singular matrix. Therefore, either the inverted matrix cannot be found or the inverted matrix formed by the computer is not correct because the round-off error caused by the computer during the inversion process will be much bigger than the true results.

Chapter VI

UNSYMMETRICAL FAULTS PROGRAM

The unsymmetrical faults program includes studies of single line-to-ground, line-to-line and double line-to-ground faults of a predetermined power system. The program can analyse the above faults with or without fault impedance Z_F in the fault path. The single-line diagram of the power system is shown in Figure 6.1. The system has a Y-connected generator supplying a Y-connected machine through a three-phase transmission line with a three-phase bank transformer connected to each end of the transmission line.

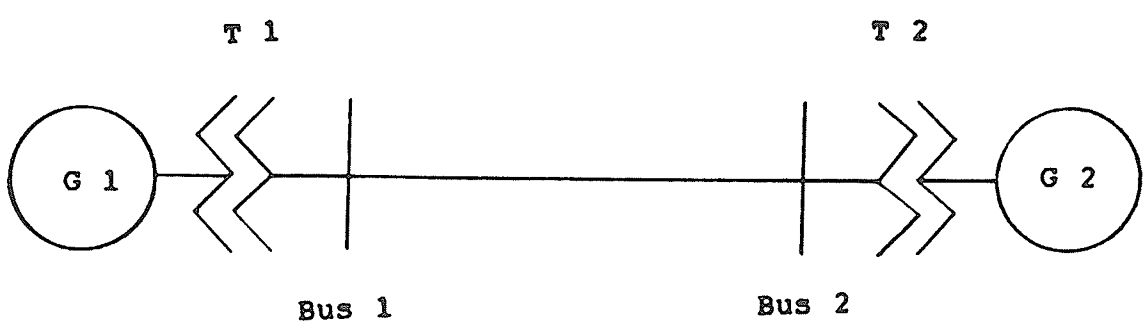


Figure 6.1: The Single-Line Diagram of the Power System in the Unsymmetrical Faults Program

6.1 Input Requirements

Input data in this program is entered in two separated diagrams. In the first diagram, the input data is:

1. The connection of each transformer and
2. The type of fault located at bus 2.

Different types of transformer connection and zero-sequence equivalent circuits are shown in Figure 6.2. The three kinds of unsymmetrical faults on a three-phase transmission line is shown in Figure 6.3. Although each fault is connected with a fault impedance, the student can analyse the bolted fault easily by entering zero as the fault impedance. Similarly, the student can change the generator and motor grounding to solidly grounded or ungrounded by entering a zero or a large number for the grounding impedance of the generator and motor.

To avoid crowding on the screen, the program produces a second picture containing the sequence networks with interconnection at the fault location and the transformer connections of the system according to the information obtained in the first picture. The student is required to enter generator, motor, transmission line and transformer data at the appropriate locations. The program represents the generator and motor as a constant voltage source in series with an impedance, and the prefault voltage for each internal voltage source is assumed to be

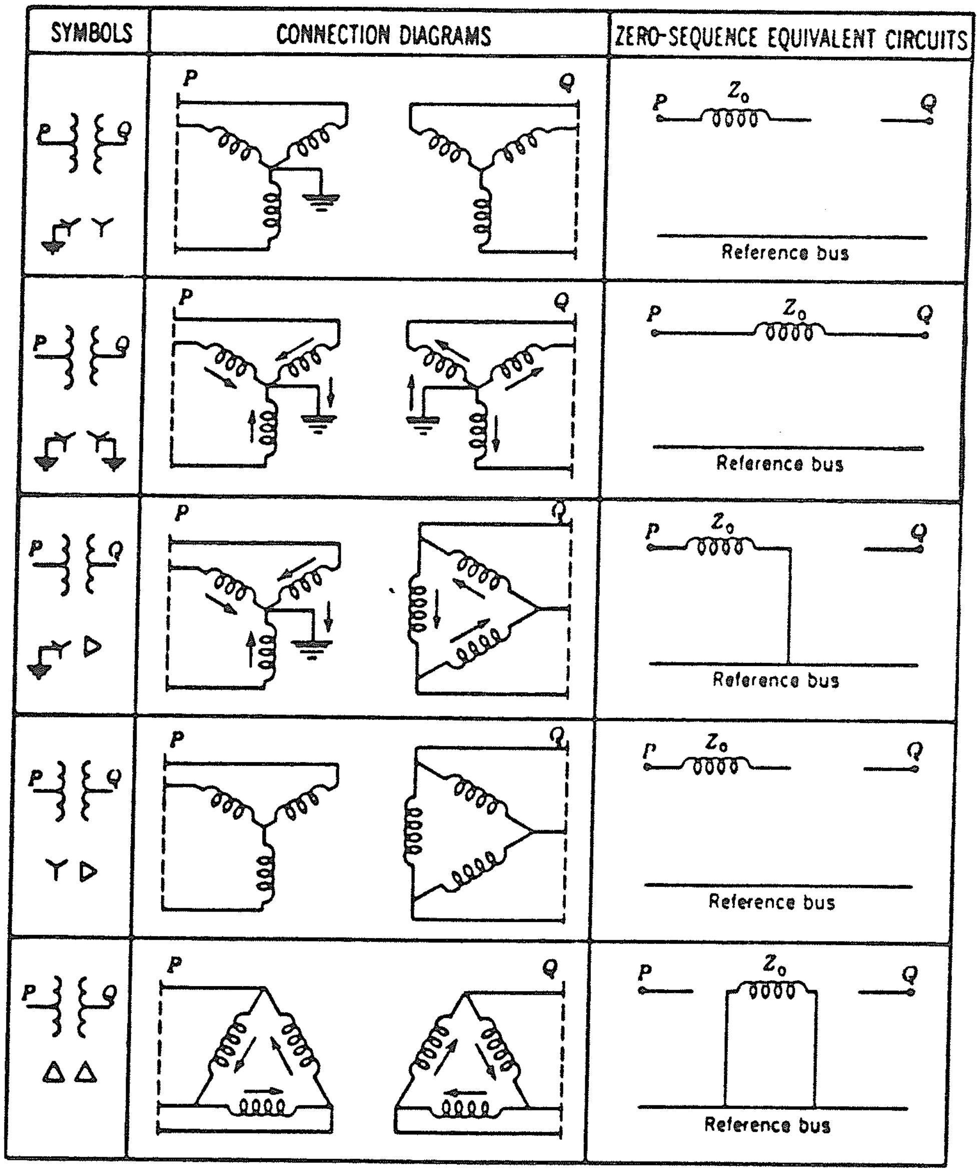
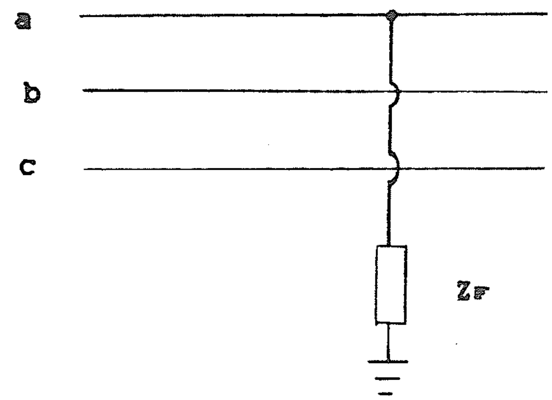
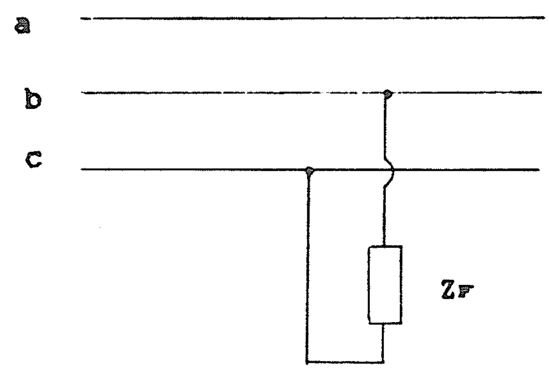


Figure 6.2: Connections & Zero-Sequence Equivalent Circuits of Three-Phase Transformer Banks

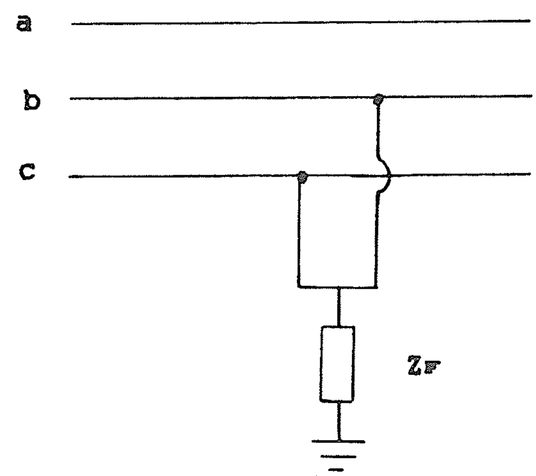
Source : Elements of power system analysis, fourth edition William D. Stevenson, Jr., page 299



(a) Single line-to-ground fault



(b) Line-to-line fault



(c) Double line-to-ground fault

Figure 6.3 : Connection of the Hypothetical Stubs for Various Faults through Impedance

$1 \angle 0^0$. The program uses a short line model to represent the transmission line. The transformer data includes series "leakage" winding impedance. The input mode of a line-to-line fault is shown in Figure 6.4.

6.2 Output Results

The fault current and current distribution, and the postfault voltages at the fault location in each sequence network will be shown in a sequence network diagram. Figure 6.5 is an output results of a line-to-line fault.

6.3 Analytical Method

The analytical procedure is simplified by using symmetrical components. In a three-phase system, the symmetrical components are called zero, positive and negative sequence. Sequence networks are interconnected only at the fault location during an unsymmetrical fault. Only the positive sequence network contains voltage sources. The neutral of the system is the reference for positive and negative sequence networks. However, the ground is the reference for the zero-sequence network. Therefore, the zero-sequence current can flow only if a circuit from the neutral to the ground is complete.

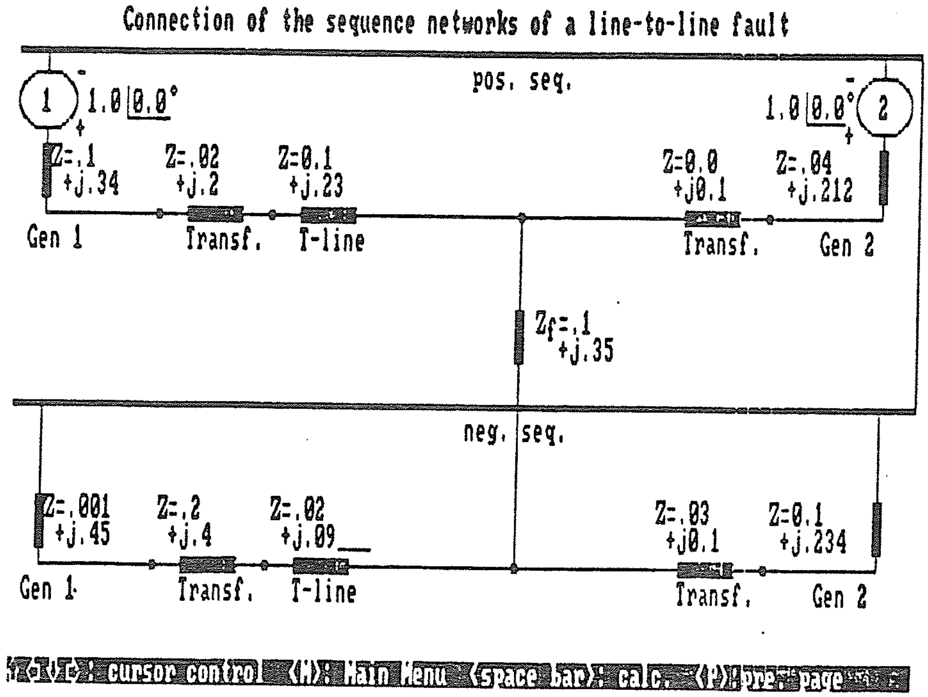


Figure 6.4 : Input Mode of the Line-to-line Fault

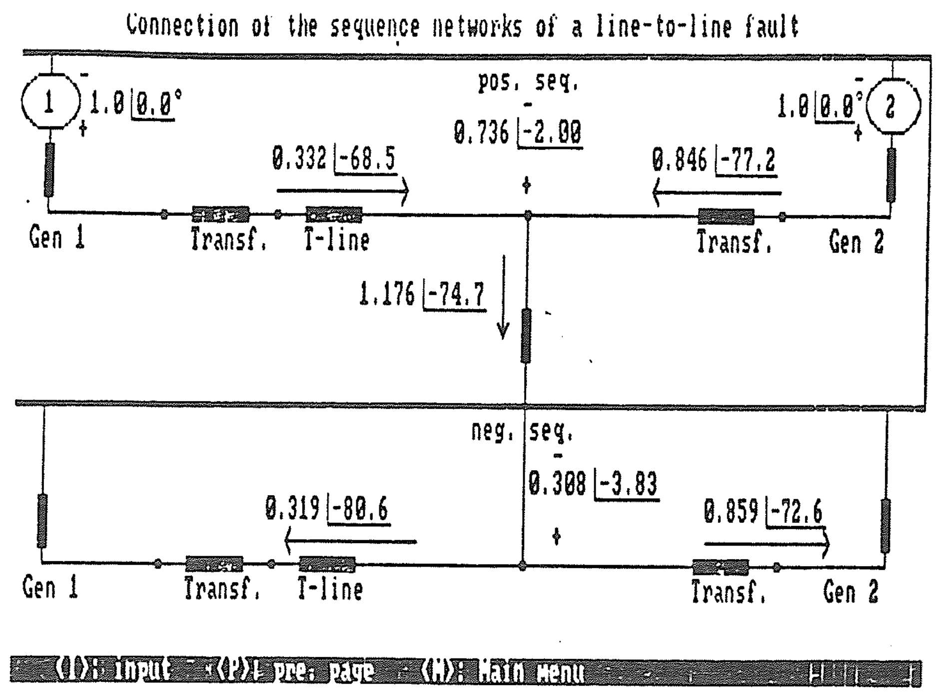


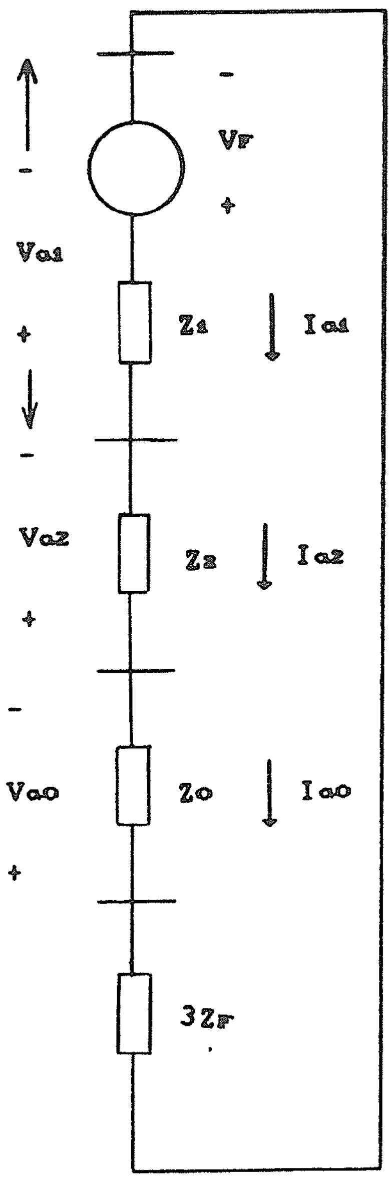
Figure 6.5 : Output Results of a Line-to-line Fault

The symmetrical components of current and voltage at the fault location can be obtained easily provided that the interconnection between the sequence networks of the power system are properly connected. The sequence networks connections of various types of unsymmetrical faults are shown in Figure 6.6. Fault impedance Z_F in the single line-to-ground and double line-to-ground faults are tripled to account for the fact that the zero sequence current is one-third of the neutral current. The program first finds out the Thevenin equivalent between the fault location and the reference bus for each sequence network. For positive sequence network, the Thevenin equivalent is the prefault voltage V_F (i.e. $1\angle 0^0$) connected in series with positive-sequence impedance Z_1. For negative and zero sequence networks , the Thevenin equivalents are the negative-sequence impedance Z_2 and zero-sequence impedance Z_0, respectively. Having the connections of various faults and the Thevenin equivalents for each sequence network, the equations for fault currents and fault voltages are derived as follows:

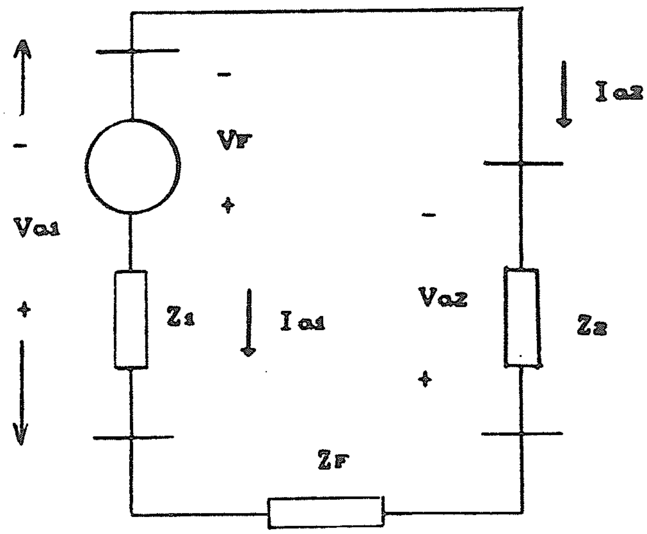
Single line-to-ground fault (phase a to ground):

$$I_{a1} = I_{a2} = I_{a0} \qquad (6.1)$$

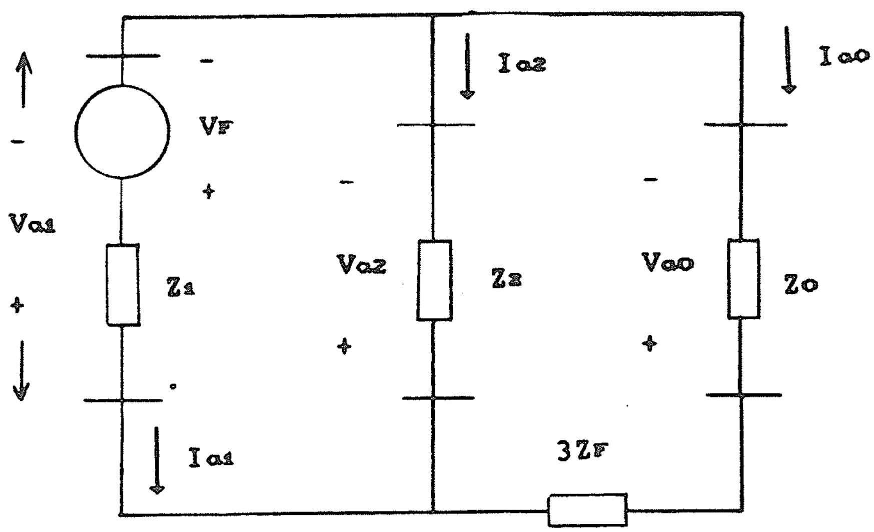
$$I_{a1} = \frac{V_F}{Z_1 + Z_2 + Z_0 + 3Z_F} \qquad (6.2)$$



(a) Single line-to-ground fault



(b) Line-to-line fault



(c) Double line-to-ground fault

Figure 6.6: Sequence Connections of Various Types of Faults Through Fault Impedance

$$V_{a1} = V_F - I_{a1} Z_1 \quad (6.3)$$

$$V_{a2} = -I_{a1} Z_2 \quad (6.4)$$

$$V_{a0} = -I_{a1} Z_0 \quad (6.5)$$

Line-to-line fault (phase b to c)

$$I_{a0} = 0, \quad I_{a1} = -I_{a2} \quad (6.6)$$

$$I_{a1} = \frac{V_F}{Z_1 + Z_2 + Z_F} \quad (6.7)$$

$$V_{a1} = V_{a2} \quad (6.8)$$

$$V_{a1} = V_F - I_{a1} Z_1 \quad (6.9)$$

Double line-to-ground fault (phase b to c to ground):

$$I_{a1} = \frac{V_F}{Z_1 + \left[\dfrac{Z_2 (Z_0 + 3Z_F)}{Z_2 + Z_0 + 3Z_F}\right]} \quad (6.10)$$

$$I_{a2} = (-I_{a1}) \left[\frac{Z_0 + 3Z_F}{Z_2 + Z_0 + 3Z_F}\right] \quad (6.11)$$

$$I_{a0} = (-I_{a1}) \left[\frac{Z_2}{Z_2 + Z_0 + 3Z_F}\right] \quad (6.12)$$

$$V_{a1} = V_{a2} \quad (6.13)$$

$$V_{a2} = -I_{a2} Z_2 \quad (6.14)$$

$$V_{a0} = -I_{a0} Z_0 \quad (6.15)$$

where I_{a1} = phase a positive-sequence current flows out of the positive-sequence network through the fault location

I_{a2} = phase a negative-sequence current flows out of the negative-sequence network through the fault location

I_{a0} = phase a zero-sequence current flows out of the zero-sequence network through the fault location

V_{a1} = phase a positive-sequence fault voltage

V_{a2} = phase a negative-sequence fault voltage

V_{a0} = phase a zero-sequence fault voltage

Chapter VII

TRANSIENT STABILITY PROGRAM

A transient stability study provides information on the capability of a power system to remain in synchronism during major disturbances such as loss of generating or transmission facilities, sudden load changes, or momentary faults. The study is best performed with a digital computer since the study involves a load flow analysis and solving differential equations representing synchronous machines. A step-by-step numerical method is required to solve the differential equations because the closed form solutions cannot be found.

This program analyzes only the first swing transient stability; in other word, damping is neglected, and voltage and speed controls are neglected. The differential equations are solved by the Runge-Kutta method. In addition, the following assumptions are applied in this program:

1. The magnitude of the internal voltage of each machine remains constant throughout the transient period.

2. The mechanical power delivered to each machine remains

unchanged during the entire transient period.

3. Only synchronous frequency currents and voltages are considered: dc offset currents and harmonics components are neglected.

The program will simulate a faulted and a postfaulted situation of a predetermined multimachine system. A symmetrical 3-phase fault will occur on any line in the system specified by the student. Then the fault will be cleared by simultaneously opening the circuit breakers at the ends of the faulted line after a specified clearing time has been reached. The multimachine system is shown in Figure 7.1. The system has two generators, a infinite bus and two loads. The program will monitor the rotor

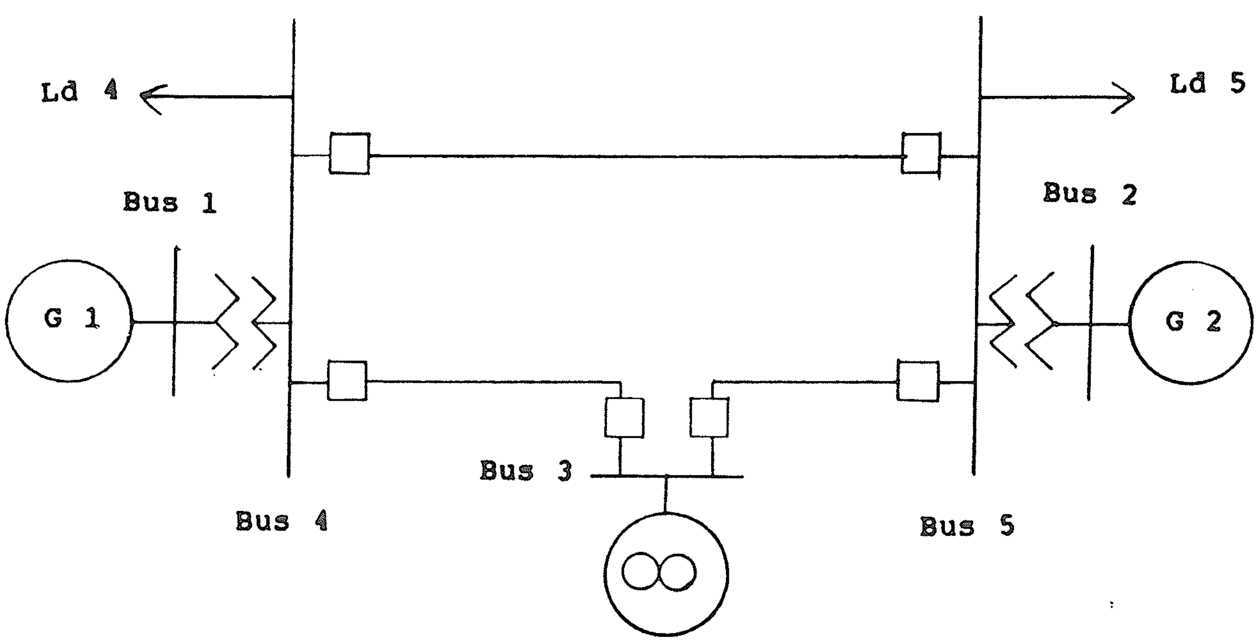


Figure 7.1 : The One-Line Diagram of the Multimachine System

angles of the two machines operating in the system throughout the entire period.

7.1 Input Requirements

Input data for this program includes:

1. Transmission line data
2. Load bus, generator bus and swing bus data.
3. Machine data
4. Fault clearing time and the transient period which the program has to monitor.
5. Plotting information.

The transmission line data and the bus data are specified as mentioned in the load flow program because a load flow analysis will be done first to find out the initial voltages and the electrical power outputs from each machine.

The machine data includes the transient reactance X_d' and the machine inertia constant H since each machine is represented by a constant internal voltage source in series with a transient reactance.

The fault clearing time and the transient period are both specified in seconds. The transient period is normally set to about one second since in the real situation the governor will be

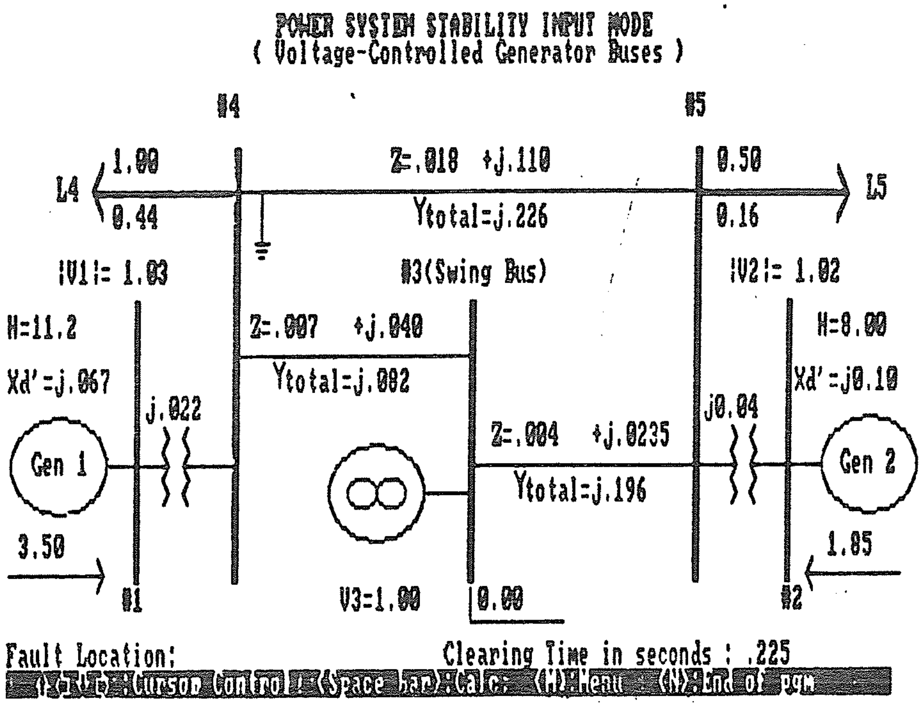
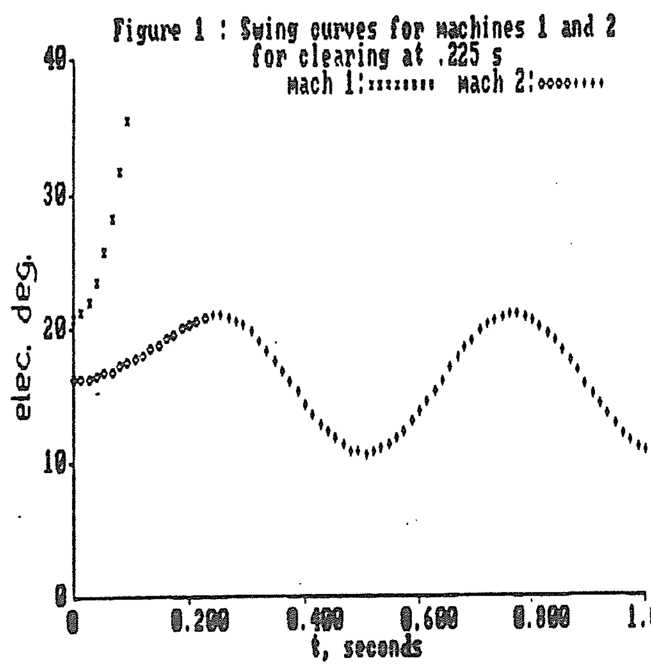


Figure 7.2 : Input Mode of the Multimachine System



Time	Mach 1 Angle	Mach 2 Angle
0.000	20.84	16.20
0.067	28.34	16.86
0.133	50.84	18.49
0.200	88.34	20.24
0.267	133.98	20.78
0.333	168.72	18.37
0.400	208.27	14.35
0.467	276.39	11.20
0.533	385.81	10.90
0.600	497.32	13.65
0.667	602.13	17.71
0.733	741.12	20.56
0.800	885.90	20.45
0.867	1037.39	17.45
0.933	1212.88	13.40
1.000	1388.71	10.82

(I) : input mode.
(C) : continue..!
(R) : previous page

Figure 7.3 : Output Plot of the Rotor Angles Vs Time Graph for the Two Machines

responding and the mechanical power input to each machine will start changing. Therefore, the machine model will no longer be valid. The diagram shown in Figure 7.2 is the input mode of this program.

7.2 Output Results

The program will produce a table to summarize the results obtained from the load flow analysis. The results will contain the voltage and power at each bus. After obtaining the necessary plotting information, the program will plot the machine angles versus time graph for both machines as shown in Figure 7.3.

7.3 Analytical Method

After the data has been obtained, the program starts the analysis in the following order:

1. A load flow analysis, the modified Newton-Raphson Method, is executed to compute the initial bus voltages, the initial machine currents and the machine electrical power outputs. The mechanical power is set to the initial machine electrical power. The transient internal voltage of each machine is then calculated using the equation.

$$E_i = E_i \underline{|\delta_i} = V_{Gi} + (j X_d') I_i \qquad (7.1)$$

Where V_{Gi} and I_i are the initial bus voltage and current corresponding to the machine i. The angle δ_i is the initial power angle of the machine i.

2. Each load is converted to a constant admittance to ground at its bus using the equation

$$Y_L = \frac{P_L - jQ_L}{|V_L|^2} \qquad (7.2)$$

Where $P_L + jQ_L$ is the power delivered to the load and $|V_L|$ is the magnitude of the corresponding bus voltage. The transient reactance of each machine and the shunt load admittance are then included in forming the prefault bus admittance matrix.

3. The prefault bus admittance matrix is modified to reflect the 3-phase fault condition. The row and column elements corresponding to the fault location will disappear because the fault bus is merged with the reference bus. All other buses except the generator buses are eleminated to reduce the size of the matrix by repeatedly using the equation

$$Y_{kj\ (new)} = Y_{kj} - \frac{Y_{kn}\ Y_{kj}}{Y_{nn}} \qquad (7.3)$$

Where $Y_{kj\ (new)}$ is the (k,j) element of the new (n-1) x (n-1) matrix.

Y_{kj}, Y_{kn}, Y_{nj}, Y_{nn} are the elements of the old n x n matrix.

4. Set the power-angle equation for the faulted condition. Solve the rotor angle δ_i from the equation.

$$\frac{2\,H_i}{\omega_s}\,\frac{d^2\delta_i}{dt^2} = P_{mi} - P_{ei} \qquad (7.4)$$

Where P_{ei} is the electrical power output from the machine i and is calculated using equation (4.2).

P_{mi} is the mechanical power input to the machine i

H_i is the machine inertia constant of the machine i

ω_s is the synchronous frequency of the machine i

δ_i is the machine angle of the machine i.

The power-angle equation is solved by the fourth order Runge-Kutta method. The transient period is equally divided into 75 intervals. Each interval has time Δt. The machine angle at time $t + \Delta t$ is computed by substituting the estimated value obtained at time t into the power angle equation and then used the fourth order Runge-Kutta method to solve the $t+\Delta t$'s value. The process goes on until the fault is cleared or the transient period is over.

5. If the fault is cleared, then the program will modify the prefault bus admittance matrix. When the fault is cleared by removing the faulted line, the program can accomplish this change by setting the elements corresponding to the fault line in the bus admittance matrix to zero and subtracting one-half of the line capacitance from the elements corresponding to buses which the faulted line is removed. The new matrix is again reduced to include only the generator buses using equation (7.3).

6. Set and solve the new power-angle equation for the rest of the transient period. The last estimated value obtained from the faulted condition will be used as the initial condition for the new power-angle equation.

In order to save memory space, the program plots the machine angles immediately after the angles are solved at each interval. The procedure is summarized in Figure 7.1.

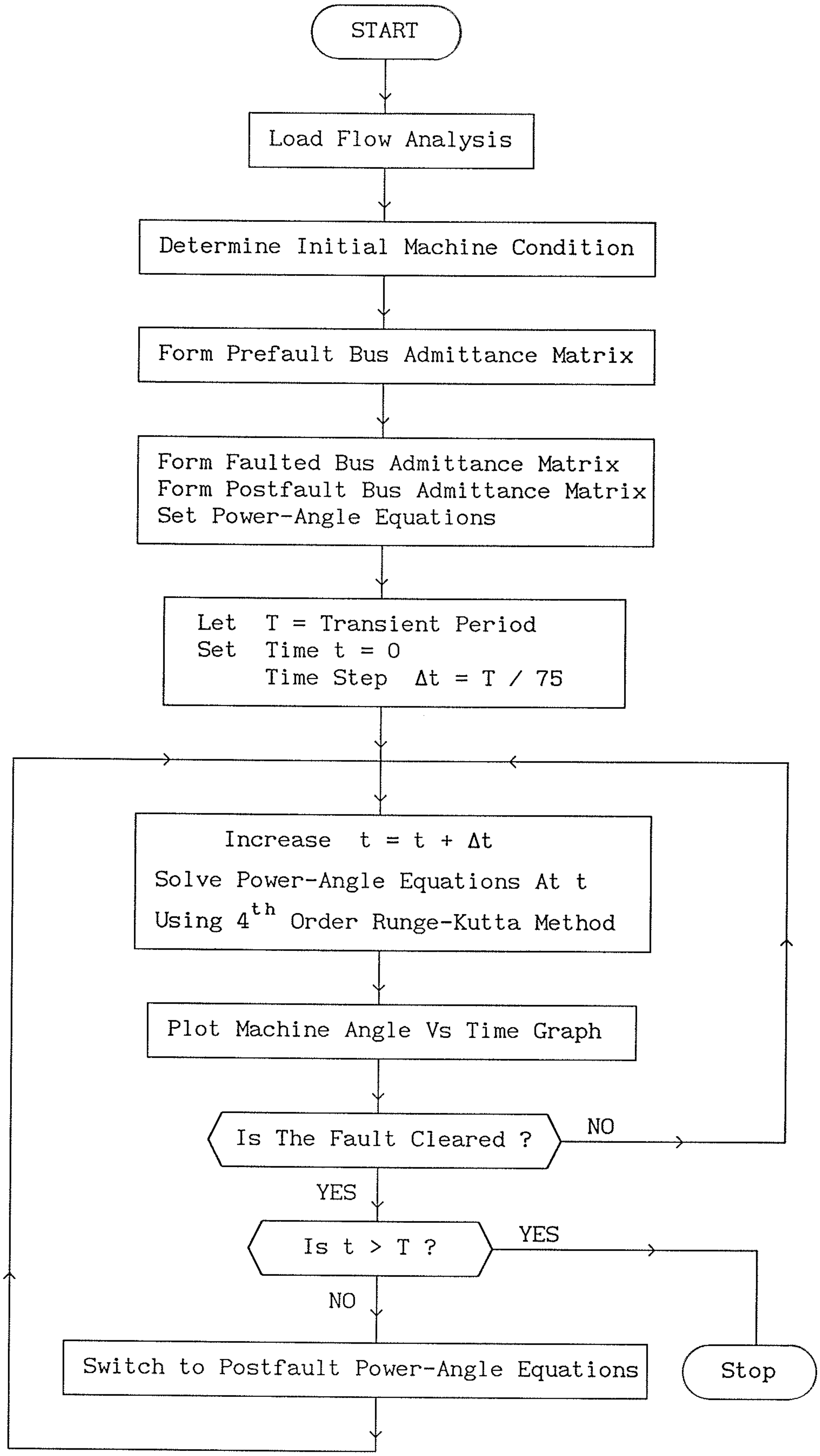


Figure 7.4 : Flow Chart6Øf the Transient Stability

Chapter VIII

SUMMARY AND CONCLUSIONS

A set of personal computer programs has been developed for introductory electric power system analysis. The programs are capable of solving the following type of problems in predefined power systems:

1. Transmission Line representation
2. Load Flow Studies
3. Symmetrical Three-phase Faults
4. Unsymmetrical Faults
5. Transient Stability.

The following user friendly features are implemented to make the programs suit a teaching environment:

1. The programs can be run on an inexpensive system: IBM-PC with a CGA card or equivalent.
2. The programs adopt graphic aided interactive techniques to improve the man-machine interface. Every program uses graphical input and graphical output.
3. Execution time is fast. Results can be seen in seconds. Syntax error is checked in the data acquisition process.

During an analysis, if an execution error causes abnormal termination, the user will be asked to enter another input.

From a teaching point of view, the programs provide a very user-friendly teaching tool in power system engineering.

There are some conclusions:

1. Quick-Basic 3.0 is a very good language for writing a graphics based program. It has many graphics functions and has fast execution speed. Moreover, the developed software is not copyright protected.
2. The Newton-Raphson method is very good a load-flow study since it is fast and reliable. With some simplifications the Newton-Raphson method can be modified to the decoupled Newton method, or "fast decoupled" method. However, there is a trade off between reliability and speed which is not worthwhile for an educational software.
3. It is important to check for unreasonable data (e.g. zero impedance for a transmission line) to ensure it does not cause computational overflow or underflow.

BIBLIOGRAPHY

Bridger, Mark. "Marshal Pascal and Pascal-2." *Byte the Small System Journal,* Vol. 12, No. 14, December 1987. p. 188.

"Distribution Primary Analysis System Program Version 4.0 Demonstration Package, User's Manual, and User's Tutorial." Electrocon International, Inc. 1986.

Glover, J. Duncan, and Mulukutla Sarma. *Power System Analysis and Design with Personal Computer Applications.* Boston: PWS, 1987

---. "A Personal Computer Software Package for Power Engineering Education." *IEEE/PES 1988 Winter Meeting,* New York, New York, January 31 - February 5, 1988.

Hatziargyriou, N. D., G. C. Contaxis, and B. C. Papadias. "Microcomputer Based Interactive Graphics for Power system Analysis Education." *IEEE Transactions on Power Apparatus and System,* Vol. PAS-104, No. 9, September 1985. pp. 2296-2301.

Haakonstad, Norma J. Letter to Prof. G. W. Swift. September 23, 1986

Lo, C. H. "An Interactive Power System Analysis with Graphics Display for Educational Use." *M. S. Thesis.* University of Missouri-Rolla, Rolla, Missouri, 1984.

Microsoft GW-Basic User's Guide and User's Reference. Microsoft Corporation, 1986.

Microsoft QuickBasic 3.0 Manual. Microsoft Corporation, 1987.

Semlyen, A., and H. Hamadanizadeh. "Computational Experiments in Power Systems." *IEEE Transactions on Power Apparatus and System,* Vol. PAS-104, No. 9, September 1985. pp. 2290-2295.

Singh, L. P. *Advanced Power System Analysis and Dynamics.* New York: Wiley, 1983.

Stagg, Glenn W., and Ahmed H. El-Abiad. *McGRAW-HILL SERIES IN ELECTRONIC SYSTEMS: Computer Methods in Power System Analysis.* New York: McGraw, 1968.

Stevenson, William D., Jr. *Elements of Power System Analysis.* 4th ed. International Student Edition. Singapore: McGraw, 1982.

Wachal, R. W., M. Z. Tranawecky, and G. W. Swift. "A Power System Analysis Package for Students, Using Computer Graphics." *IEEE Transactions on Power Apparatus and System,* Vol. PAS-103, No. 2, Febraury 1984. pp. 445-448.

Yu, David C., Shin-Tzo Chen, and Robert F. Bischke. "A PC Oriented Interactive and Graphical Simulation Package for Power System Study." *IEEE/PES 1988 Winter Meeting,* New York, New York, January 31 - February 5, 1988.

Appendix A

Listing of Source Programs

Copies of the programs are too long (150 pages) to be included here. See Prof. G. W. Swift or K. F. Wong for copies.