THE FINAL RESEARCH RESULTS SUPPORTED BY THE KFAS INTERNATIONAL SCHOLAR EXCHANGE FELLOWSHIP PROGRAM, 2004-2005

STOCHASTIC PREDICTION OF VOLTAGE SAGS IN DISTRIBUTION SYSTEMS

Dự báo ngẫu nhiên sag điện áp trong lưới phân phối điện

ISEF Laureate : Bach Khanh Quoc

Electric Power Systems Department, Electricity Faculty
Hanoi University of Technology

Host Scholar : Prof. Moon Il Seoung

Power System Laboratory, Electrical Engineering and
Computer Science Dept., School of Engineering
Seoul National University

August 20, 2005

ACKNOWLEDGEMENT

This work would not have been possible without supports from organization and individuals

This work is supported by the Korea Foundation for Advanced Studies' International Scholar Exchange Fellowship for the academic year of 2004-2005. Thanks are due to KFAS for their financial support as well as the encouragement during the author's stay in Korea.

Professor Moon Seung Il (the host scholar) gave me the opportunity to conduct my research at Power System laboratory, Electrical Engineering and Computer Science department, Seoul National University as well as provided the supervision of this work during it was developping. Doctor Won Dong Jun introduced the research field and gave useful discussions for the research. Thanks to them for their supports.

Doctor Chung Ir Yop and all members of Power System laboratory regularly participated in research discussions and created the best possible research conditions for the author to carry out this work in the laboratory. Thanks to all them for their amicably collaboration.

ABSTRACT

This research presents a new case study of fault distribution modeling in stochastic prediction study of voltage sags in the distribution system. Using two-dimension stochastic models for fault modeling allows to obtain the fault performance for the whole system of interest that helps obtaining not only sag performance at individual locations but also system sag performance through system indices of voltage sag. By using bivariate normal distribution for fault distribution modeling, the research also illustrates the influence of the used model parameters on system voltage sag performance. The research also develops $SARFI_X$ for distribution system regarding phase loads that creates better estimation for voltage sag performance.

TÓM TẮT

Nghiên cứu này trình bày một phương pháp mô hinh mới về phân bố sự cố trong các nghiên cứu dự báo ngẫu nhiên sag điện áp trong lưới phân phối điện. Việc sử dụng mô hình dự báo ngẫu nhiên nhị biến để mô phỏng phân bố sự có cho phép tính được diễn biến sự cố đối với toàn bộ lưới điện cần xem xét, giúp cho không chỉ tính toán được diễn biến sag điện áp tại từng điểm riêng biệt mà cả diễn biết sag điện áp trên toàn lưới điện thông qua các chỉ tiêu sag hệ thống. Nhờ việc sử dụng phân bố chuẩn nhị biến để mô hình hoá phân phối sự cố, nghiên cứu cũng phân tích ảnh hưởng của các tham số của mô hình đối với kết quả diễn biến sag điện áp. Nghiên cứu cũng phát triển chỉ tiêu $SARFI_X$ *cho lưới phân phối co xem xét đến phụ tảI pha, nhờ đó có được sự đánh giá tốt hơn về tình hình sag điện áp.*

INDEX TERMS

Distribution system, power quality, voltage sag frequency, stochastic prediction, bi-variate normal distribution (BND), phase loads.

TỪ KHOÁ

Lưới phân phối điện, Chất lượng điện năng, Tần xuất sag điện áp, Dự báo ngẫu nhiên, Phân bố chuẩn nhị biến, Phụ tải pha

CONTENTS

ACKNOWLEDGEMENT……………………………………………………………I

ABSTRACT………………………………………………………………………II

INDEX OF FIGURES AND TABLES……………………………………………….. V

I. INTRODUCTION………………………………………………………..…….. 1

1.1. Voltage Sag Definitions…………………………………………………………...1

1.2. Stochastic Compatibility Assessment and Concerned Problems………………..... 1

1.3. Stochastic Prediction Methods………………..………………...………………...2

II. FAULT DISTRIBUTION SIMULATION……………………………………….. 4

III. FAULT MODELING IN DISTRIBUTION SYSTEM BASING ON FAULT CAUSES..5

3.1. Equipment Failure……………………………………………………………...5

3.2. External Causes………………………………………………………………...5

IV. PROBLEM DEFINITION AND SOLUTION…………………………………… 6

4.1. Case study definition…………………………………………………………... 6

4.2. Fault rate modeling……………………………………………………………8

4.3. Procedures of stochastic prediction……………………………………………… 11

4.4. Development of voltage sag indices……………………………………………… 13

4.5. Evaluation of influence of fault distribution
modeling on voltage sag performance……………………………………………... 14

V. RESULT DEMONSTRATION AND ANALYSIS……………………………….....15

5.1. SARFI for phase loads………………...………………...……………….................15

5.2. The influence of fault distribution modeling on voltage sag performance………. 16

VI. CONCLUSIONS……………………………………………………………. 19

BIBLIOGRAPHY………………………………………………………………. 20

APPENDIX A PROGRAM OF SAGS CALCULATION IN DISTRIBUTION SYSTEM..... 21

A.1. Introduction………………...………………...……………………………….. 21

A.2. Distribution system short-circuit modeling ……………………………………... 21

A.3. Block diagram…………………………………………………………………24

A.4. Main module in Matlab Code…………………………………………………. 24

A.5. Data files………………………………………………………………………44

APPENDIX B IEEE 123 BUS RADIAL DISTRIBUTION TESTFEEDER ..…………..46

B.1. Introduction……………………………………………………………………46

B.2. Numbering principle……………………………………………………….... 46

APPENDIX C DATA FILES……………………………………………………….. 48

INDEX OF FIGURES AND TABLES

FIGURES

Figure 1. *Definitions of voltage magnitude events as used in IEEE Std.1159*………… 1

Figure 2 *Voltage divider model*……………………….…………………………….……... 3

Figure 3. *A simplified bathtub curve*………………………………………………...5

Figure 4. *Major causes of interruption for three US utilities* ………..……………….. 6

Figure 5. *Example of Bivariate Normal Distribution*…………………..……………... 6

Figure 6. *Considered fault locations*………………………………………………...7

Figure 7. *The joint probability density function f(x,y) for* $\mu_x=\mu_y=0$…………………... 10

Figure 8. *Mapping of IEEE - 123 Bus Radial Distribution Test Feeder*……………… 12

Figure 9. *Sag frequency spectrum and* $SARFI_X$ *of different phase loads for the case the mean value is at node 13 and deviation* $\sigma_x=\sigma_y=0.5\Delta_{MAX}$ …………………...15

Figure 10. *Sag frequency spectrum and* $SARFI_X$ *of the whole system for different mean positions for the case the deviation is* $\sigma_x=\sigma_y=0.5\Delta_{MAX}$……………………... 16

Figure 11. *Voltage sag frequency spectrum of the load bus 63 on phase A for different deviations.Mean value is at node 67 (upper) and node 13 (lower)* ………… 16

Figure 12. *Voltage sag frequency spectrum for loads on phase A for different deviations. Mean value is at node 67 (upper) and node 13 (lower)*…………………….. 16

Figure 13. *Voltage sag frequency spectrum and* $SARFI_X$ *for the whole system for different deviations for mean value at node 67 (two upper) and node 13 (two lower)* …...………………………………………………………….. 17

Figure 14. *Voltage sag frequency distribution for different ranges of voltage sag magnitude*………………………………………………………………… 18

Figure A.1. *The block-diagram of the program of sag calculation in distribution systems*…………………………………………………………………….22

Figure A.2. *Numbering of IEEE 123 bus radial testfeeder*……………………………… 44

TABLES

Table 1. *Permanent fault causes* …………………………………………………….. 5

Table 2. *System fault rate breakdown* ……………………………………………….. 12

Table A.1. *The format of data file testfeeder123sys.txt* …………………………………42

Table A.2. *Fault position (co-ordinate)and applicable fault type for distribution transformers*…...…………………….…..…………………………………. 49

Table A.3. *Fault position (co-ordinate)and applicable fault type for lines*……………..51

Table A.4. *Scheme renumbering and line parameters*……………….…………………53

I. INTRODUCTION

1.1. Voltage Sag Definitions

Among power quality phenomenon, voltage sag (dip) is defined by IEEE1159, 1995 as a decrease in rms voltage to between 0.1 to 0.9 of nominal voltage at power frequency for duration of 0.5 cycle to 1 minute. Figure 1 gives a diagram of classifying different events according to the magnitude and the duration.

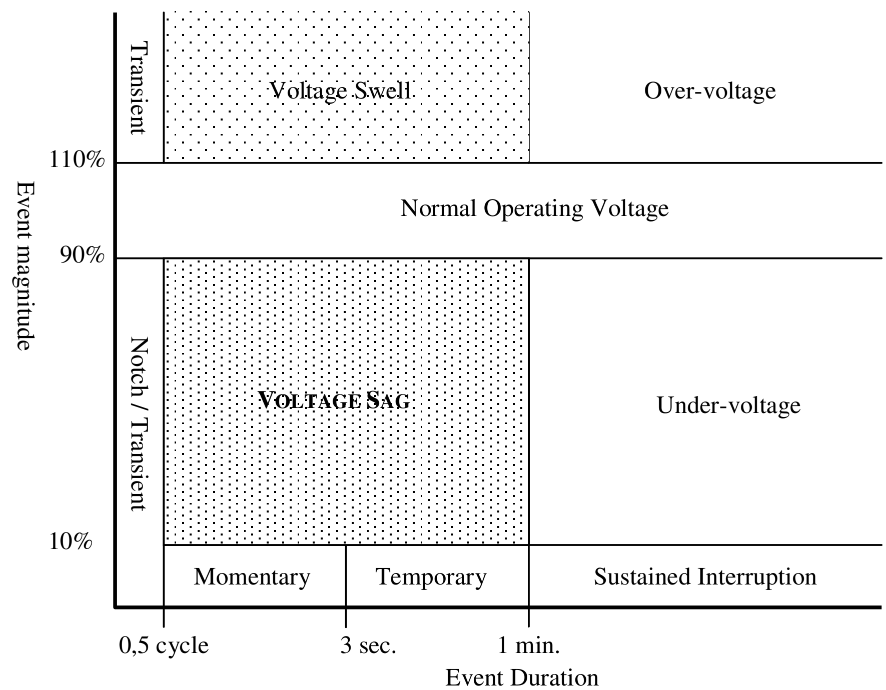

Figure 1. Definitions of voltage magnitude events as used in IEEE Std. 1159

In supplying power to an electric equipment, if above said characteristics of voltage sag exceed the equipment voltage tolerance, it may trip or malfunction that possibly results in even large economical losses. Voltage sags are mostly caused by short-circuit faults in power system, starting of large motors and transformer energizing. Although economical losses can be much easier recognized through an interruption than a voltage sag event, in contrary, there are far more voltage sags than interruptions and thus total damage due to sag is still considerable. Interests in voltage sag have been getting much greater recently due to its problem causing on the performance of such sensitive electronic equipments as adjustable-speed drives, process-control equipment and computers that are widely used nowadays.

1.2. Stochastic Compatibility Assessment and Concerned Problems

One of main purposes of voltage sag researches is to obtain the estimate at required accuracy about the influence of voltage sag on the operation of electric equipment in the power system in order to opt for a better supply. Therefore, researches about voltage sag, similar to other power quality phenomena, usually relate with the basic process known as a "compatibility assessment" [1] which undergoes following steps

- Obtain system performance that in fact is to obtain the number of voltage sags with different characteristics at all load locations throughout the system.
- Obtain equipment voltage tolerance (sensitivity to voltage variation) that is to gain the whole range of equipment behaviors against various voltage sags.
- Compare equipment sensitivity with the performance of the supply system and determine expected impact of voltage sag on equipment basing on information obtained from two earlier steps.

Researches to date have already evidenced that obtaining voltage sag performance (or voltage sag quantification) is not just the most important step, but also the most difficult step that still needs much further efforts for improvement.

The information about voltage sag performance can be obtained through several ways. Monitoring is generally seen as the best manner because it can obtain exact data about disturbance events. However, it mainly provides information about common events. Beside that, it is also challenged by several difficulties such as the trade-off between cost and accuracy, long time period of monitoring, the selection of monitoring locations as well as result prediction for locations without monitoring. An alternative is the stochastic prediction of voltage sag that simulates the whole problem on the computer using power system simulation techniques and methods of modeling causes resulting in voltage sags. The great advantage of stochastic prediction over monitoring is that it can obtain the results at required accuracy for even uncommon events for not networks in operation alone, but also the networks in design with many different network topologies and operational conditions. With the recently fast development of computer aided simulation tools, stochastic prediction of voltage sag is increasingly the preferable approach for recent researches on voltage sag quantification.

1.3. Stochastic Prediction Methods

The expected result of a stochastic prediction study is firstly the number of voltage sags (voltage sag frequency) with specific characteristics for a period of time (e.g. in a year) at all load locations in the system of interest. "The method of fault positions" and "the method of critical distances" are known as the most widely used methods to bring about the above said solutions [1].

The method of fault positions calculates magnitude and duration of the voltage sag for many possible faults in the system. The method proceeds, schematically, as follows

- Determine the area of the system in which short-circuits will be considered
- Split this area into small parts. Short-circuits within one part should lead to voltage sags with similar characteristics. Each small part is represented by one fault position in an electric circuit model for the power system.
- For each fault position, the short-circuit frequency is determined. The short-circuit frequency is the number of short-circuit faults per year in the small part of the system represented by a fault position.

- By using the electric circuit model of the power system the sag characteristics are calculated for each fault position.
- The results from two previous steps (sag characteristics and frequency of occurrence) are combined to obtain stochastical information about the number of sags with characteristics within certain ranges.

This method is normally applied for sag calculation for a large system for which using power system modeling techniques is preferable.

The method of critical distances is considered as the extension for a method of voltage sag analysis in distribution system which determines the fault position extent for different voltage sag levels. It is based on the voltage divider model (Figure 2) to find out where in the network a fault would lead to a voltage sag down to a given magnitude value. To estimate the number of sags below a certain magnitude, it is sufficient to add all lengths of lines and cables within critical distance from the point of common coupling (pcc). The total length of lines and cables within the exposed area is called "exposed length". The resulting exposed length has to be multiplied by the failure rate per unit length to obtain the number of sags per year.

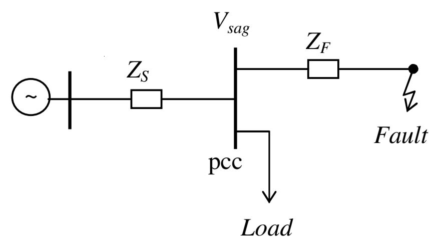

Figure 2. Voltage divider model

The method of fault positions have been applied widely in recent researches, even for distribution system due to the ease of problem simulation in computer by various comfortable software of power system modeling [10,11]. It's notable that no matter what method is used, a stochastic prediction study always needs to solve two critical problems. They are the quantitative simulation of the causes of voltage sags and the simulation of power system for computing voltage sag characteristics. The latter can be easily performed with high accuracy by simulating techniques on the computer because almost data of power system is deterministic. However, building up an "accurate" simulation of the causes of voltage sags is really uneasy because the accuracy of its data is normally outside our control. Data of causes of voltage sags can only obtained by observing the operation of the components in the system that is stochastically identical to the data recorded from monitoring and thus, it has the same uncertainties as what the monitoring can generates. To deal with these uncertainties, researches in the past often focus on specific contexts influenced by individual causes of voltage sag. Among main causes of voltage sags, faults in the power system account for a large part and the assessment of voltage sag performance basing on the fault distribution simulation is a good approximation to the overall performance of voltage sag. However, faults them selves are also random events and its simulation still deals with the uncertainties of observed fault data.

This research develops a new case study for modeling fault distribution for stochastic prediction of voltage sag in distribution system using the method of fault position. The simulation of distribution system and fault distribution modeling are made on MatLab for calculating not only site voltage sag indices, but also system indices.

II. Fault Distribution Simulation

In the stochastic prediction study, *simulating fault distribution in a power system is to determine the short-circuit fault frequency* (i.e. fault rate or the number of short-circuit faults per year) for all different fault types at any possible fault positions throughout the system of interest. Obviously, it implies significant concerns : *fault location, fault type and fault rate*.

Choosing fault locations is the first task in fault distribution simulation. The fault closely relates with component reliability because in principle, any component in the system is probable to fall into faults during operation. The inclusion of all component locations in the set of considered fault locations may improve the accuracy of the simulation, but also lead to massive computational efforts. For stochastic prediction, fault locations are generally chosen basing on the criterion saying that a fault position should represent short-circuit faults leading to sags with the similar characteristics [1]. Several researches in the past reported different patterns of choosing fault locations [8,10,11]. Distribution system features typically radial network topology with small fractions of line and distribution transformers along the trunk feeders. Therefore, it's possible to apply only *one fault location for a distribution transformer* and also *one fault location for a fraction of line*. In this manner, the distribution of fault rate for each fault locations can be made much easier.

Different fault types should be decided at each fault location mainly depending on network phasing topology. Fault rate of each fault type is determined following the fault rate distribution and the contributory percentage of different fault types which can be normally referred from the observed historical data.

Fault rate at each location heavily depends on the number of fault locations, fault type and fault causes. While two first concerns have been discussed quite a lot in researches in the past, the distribution of fault rate for selected fault locations still have got less interest. The most common assumption argued that because fault can happen anywhere in the system, stochastically it's possible to model fault rate as uniform distribution [7,8]. In this sense, fault rate at each location is identical to component failure rate which is calculated basing on component reliability. However, in reality, many factors can lead to faults, not component reliability alone and fault rates at different locations in the system are rarely the same. If the stochastic prediction study is applied for a large power system, the above said assumption of fault distribution may neglect some important factors causing faults. Recently, some report in [11] proposed some interesting one-dimension fault distributions along individual line segments for evaluating its influence on sag prediction. But in this case, it's difficult to obtain a system index about voltage sag performance. In order to have a better simulation of fault distribution which not just takes almost possible fault causes into consideration, but also allows to calculate system indices of voltage sag, it needs to analyse fault causes and build up a suitable simulation of the fault distribution for entire system of interest.

III. FAULT MODELING IN DISTRIBUTION SYSTEM BASING ON FAULT CAUSES

Although there are various factors that cause faults in the power system, it's possible to group them into two parts namely equipment failure and external factors [3]. Table 1, as an illustration for said grouping, shows the study at Pacific Gas & Electric Co. for fault performance [5].

Table 1. Permanent fault causes

Source	Equipment failure	Loss of supply	External factor
Rural	14.1 %	7.8 %	78.1 %
Urban	18.4 %	9.6 %	72.0 %

3.1. Equipment Failure

Equipment failure is basically due to defects inside itself which are probably created during the manufacture, transportation and installation. It depends on the time of putting in operation, aging period and maintenance conditions. According to the reliability theory, equipment failure is often modelled as the component failure rate. There are several models of failure rate but about 95% cases use exponential distribution which assumes that the component failure rate is constant. This value is normally understood as the average failure rate during useful life of the "bathtub" curve (Figure 3). Therefore, in the case the same type of equipment is used throughout the system (e.g. the same type of dis-tribution transformer used in dis-tribution systems), it's possible to assume *the failure rate of equipment follows the uniform dis-tribution* depending on equipment type although it still may cause some errors (e.g. not all equipments put in oper-ation in the same time or have the same maintenance conditions).

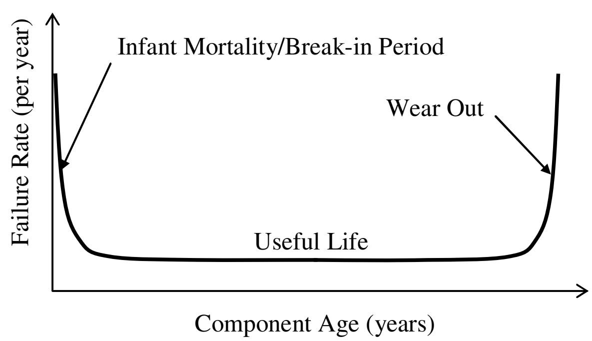

Figure 3. A simplified bathtub curve

3.2. External Causes

Beside equipment failure, there are in fact many other factors derived from ambient environment that may cause faults. Some factors can influence on the fault performance of the supply system for a large area such as severe weathers (wind storms, lightning, ice storm) meanwhile others mainly have local impacts on relative small size networks like trees, animals (birds, squirrels, mice, snakes, fire ants…). Human factors (scheduled interruption, human errors, vehicle ac-cidents, dig-ins, mischief and vandalism) can create either fault for small section of the system and severe faults for a large system. Figure 4 presents an example of the breakdown of interruption causes for three major US utilities.

External causes occur randomly and they can be simulated using different stochastic

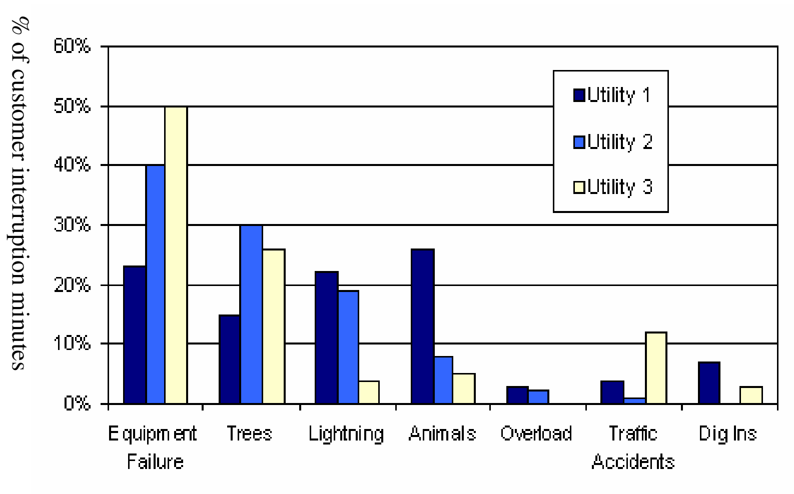

Figure 4. Major causes of interruption for three US utilities

models. Using one-dimension stochastic models seem not to be suitable because it can only apply for modelling fault distribution along individual fractions of line as above explained. An idea of two-dimensions stochastic models using bivariate distribution models will be more convenient instead. Figure 5 illustrates the BND model that may be used for simulating fault distribution in the power systems.

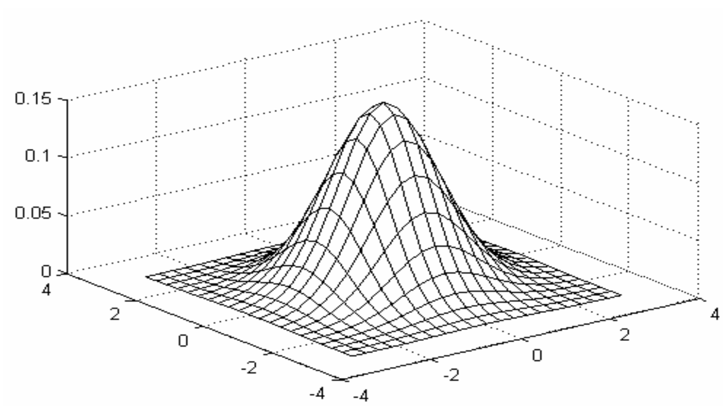

Figure 5. Example of Bivariate Normal Distribution

For a large system, it's hard to obtain a converged two-dimension fault distribution model for various causes. However, for such small to medium size networks as segments of distribution system of which historical monitoring data of fault performance shown that faults due to external factors were occurred concentrative on one location (e.g. some lines pass through an small area which is at high risk of faults due to industrial pollution or trunk fall), it's the favourite condition to obtain a fault distribution model which can converge.

This research proposes a new case study of fault distribution modelling using bivariate stochastic models for the segment of distribution system fed from a bulk point substation that allows the stochastic prediction study to calculate system sag indices and evaluate the influence of applying different fault distribution models on voltage sag performance for the system of interest.

IV. PROBLEM DEFINITION AND SOLUTION

4.1. Case study definition

To illustrate stochastic prediction study of voltage sag in the distribution network, the paper uses IEEE 123 Bus Radial Distribution Feeder [12]. It can be seen as a distribution system fed from a bulk point. It generally does not narrow the scope of application of the study with following assumptions

- Voltage sags are only caused by faults in the system.
- Only consider fault locations in the system of inte-rest. The faults in other segments of distribution network fed from other intermediate substations can be skipped as the short-circuit impedance between the nearest fault location and the point of common coupling is the impedance of the intermediate transformer which in reality is rather high. Similarly, the faults in low voltage networks are also ignored because of the large short-circuit impedance of distribution trans-former. This assumption only neglects the voltage sag caused by faults in the transmission system. It will be con-sidered if the stochastic prediction for large transmission system [8] is included.

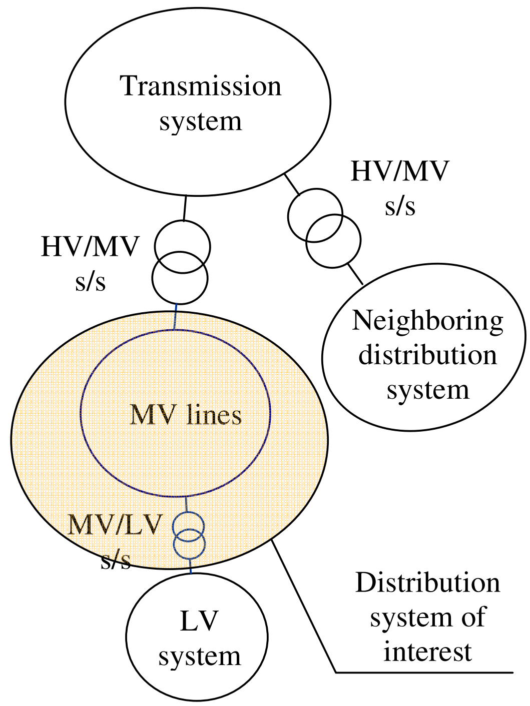

Figure 6. Considered fault locations

- In term of reliability, the system of interest is assumed to have two main components : lines and distribution transformers. The reliability of any other distribution equipment is supposed to include in these two component reliabilities.
- Fault positions are chosen as follows : one position for each node connected with distribution transformer for transformer faults and one position for each fraction of line (the line connecting between two nodes) for faults on lines. Because the lines are short, it's possible to choose the fault position anywhere along it. In this research, the fault position for a fraction of line is choosen at the end of this fraction of line which is identical to fault location of the transformer if this node is connected to a distribution transformer. For the IEEE 123 bus distribution feeders, there are 122 fraction of lines and 87 nodes connected with distribution trans-formers. Therefore, there are 209 chosen fault positions.
- Each fault position is applied all possible fault types (single phase to ground, phase to phase, two phases to ground and three phases to ground) depending on the number of available phases there.

- At each chosen fault position, fault rate due to external factors is a random variable depending on fault location. Fault rate of each fraction of line depends on its length and above chosen fault position which is a random variable depending on that position.

Basing on chosen fault positions, the computation of voltage sags at all load buses throughout the system of interest is programmed on Matlab on the basic of [4] that proves to have reliable results. Appendix A, B. Provides the Program in Latlab code developped by the research. Voltage sag frequency at each load bus is obtained when applying fault rate to each fault position. Fault rate at chosen fault position is calculated basing on fault distribution modeling as the following part. Finally, related voltage sag indices are calculated.

4.2. Fault rate modeling

Faults are random events and as above indicated, it's possible to simulate it using stochastic distribution models. On the basic of above definition and assumption, the fault rate of each fault type at each fault position equals sum of equipment failure rate and fault rate due to external factors.

$$\begin{aligned} N &= N_{trans} + N_{line} \\ &= \alpha_{eq}.N_{trans} + \alpha_{ex}.N_{trans} + \alpha_{eq}.N_{line} + \alpha_{ex}.N_{line} \\ &= N_{eq\text{-}trans} + N_{ex\text{-}trans} + N_{eq\text{-}line} + N_{ex\text{-}line} \end{aligned} \quad (1)$$

where

N_{trans} : Number of transformer fault of the system of interest
N_{line} : Number of line fault of the system of interest
$N_{eq\text{-}trans}$: Number of transformer fault of the system of interest due to equipment failure
$N_{eq\text{-}line}$: Number of line fault of the system of interest due to equipment failure
$N_{ex\text{-}trans}$: Number of transformer fault of the system of interest due to external causes
$N_{ex\text{-}line}$: Number of line fault of the system of interest due to external causes
α_{eq} : Contributory percentage of equipment failure
α_{ex} : Contributory percentage of faults due to external factors
$\alpha_{eq} + \alpha_{ex} = 100\%$

The equipment failure rate depending on the equipment type (transformer or line) is supposed to follow uniform distribution. Therefore, for the fault position of transformer i, transformer failure rate is calculated as follows

$$N_{eq-trans(i)} = \frac{\alpha_{eq}.N_{trans}}{m_t} \quad (2)$$

where

m_t : Total distribution transformer

The line failure rate is normally expressed in number of faults per year per metre (or kilometre) length. However, as above said, because of short lengths, failure rate is calculated for the whole fractions of line i as follows

$$N_{eq-line(i)} = \alpha_{eq}.N_{line}.\frac{l_i}{\sum_{k=1}^{m_b} l_k} \qquad (3)$$

where

m_b : Total fractions of line

l_i : The length of fractions of line i (in foot or metre).

The distribution of fault rate due to external factors depending on fault positions is supposedly in compliance with BND model. So, the fault rate at each fault position is as follows

For transformer i :

$$N_{ex-trans(i)} = \alpha_{ex}.N_{trans}.W_{trans(i)} \qquad (4)$$

For the fraction of line i :

$$N_{ex-line(i)} = \alpha_{ex}.N_{line}.\frac{l_i.W_{line(i)}}{\sum_{k=1}^{m_b} l_k.W_{line(k)}} \qquad (5)$$

where

$W_{trans(i)}$, $W_{line(i)}$: Weighted factors of fault rates of transformer i and fractions of line i that follow the BND model depending on fault positions.

BND is also known as bivariate Gaussian distribution which has the joint probability density function as follows

$$f(x,y) = \frac{1}{2\pi.\sigma_x.\sigma_y.\sqrt{1-\rho^2}}.\exp\left[-\frac{z}{2.(1-\rho^2)}\right] = \Phi(\mu_x,\mu_y,\sigma_x,\sigma_y,\rho) \qquad (6)$$

where

$$z = \left(\frac{x-\mu_x}{\sigma_x}\right)^2 - \frac{2.\rho.(x-\mu_x).(y-\mu_y)}{\sigma_x.\sigma_y} + \left(\frac{y-\mu_y}{\sigma_y}\right)^2 \qquad (7)$$

μ_x, μ_y, σ_x, σ_y, : Means and standard deviations of two variables x, y. ρ : Correlation coefficient.

Among well know bivariate distribution models, BND is the one that's applicable for continuous variables [13,14]. The joint probability density *f(x,y)* is a function of 5 parameters. By changing the parameters it's possible to estimate the influence of fault distribution modeling on voltage sag performance. Figure 7 plots some options of *f(x,y)* for different parameters.

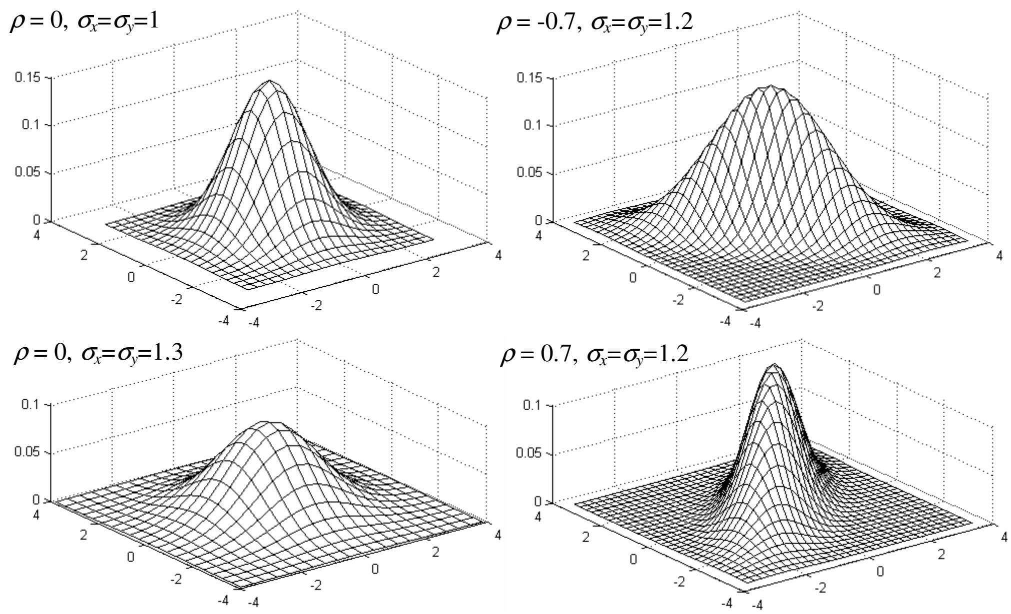

Figure 7. The joint probability density function f(x,y) for $\mu_x=\mu_y=0$

In applying BND for fault distribution modeling, if the co-ordinates of fault positions are supposedly independent variables, $\rho = 0$.

Probability for a fault to occur at the fault position (x_i,y_i) within an area $\Delta s_i = \Delta x_i.\Delta y_i$

$$F(\Delta x_i, \Delta y_i) = \iint_{\Delta s_i} f(x, y)dxdy = \frac{f(x_i, y_i).\Delta s_i}{\sum_{k=1}^{\infty} f(x_k, y_k).\Delta s_k} \tag{8}$$

If $\Delta s_i = \Delta s_0$ = const (∀i=1,m) and m is large enough then the distribution is normalized as follows

$$F(\Delta x_i, \Delta y_i) \approx \frac{f(x_i, y_i).\Delta s_0}{\sum_{k=1}^{m} f(x_k, y_k)\Delta s_0} = \frac{f(x_i, y_i)}{\sum_{k=1}^{m} f(x_k, y_k)} \tag{9}$$

For the distribution system, geographically, network nodes are disposed relatively uniform, the following approximation is applicable

- Faults rate at transformer i

$$W_{trans(i)} = \frac{f(x_i, y_i)}{\sum_{k=1}^{m_t} f(x_k, y_k)} \tag{10}$$

- Fault rate for the fraction of line i

$$W_{line(i)} = \frac{f(x_i, y_i)}{\sum_{k=1}^{m_b} f(x_k, y_k)} \tag{11}$$

where x_i, y_i is the co-ordinates of fault position i.

4.3. Procedures of stochastic prediction

The process of stochastic prediction study consists of the following steps

Firstly, the system fault rate (the total number of faults occurring in the system of interest over a certain period) is assumed to be an arbitrary number, say, 500 faults. The system fault rate of each fault type will be determined basing on the assumptions of its contributory percentage as follows

- Single phase to ground (N1) : 80%
- Two phase to ground (N11) : 10%
- Two phase together (N2) : 8%
- Three phase to ground (N3) : 2%

Also, the fault rate as per components are supposed to be

- Transformer : 50%
- Line : 50%

Above assumptions are in fact based on actual survey data [5]. Summarily, the system fault rate for each fault type is given in Table 1. A parameter (α_{eq} or α_{ex}) is also included allowing to consider the influence of fault causes due to external factors.

Secondly, the fault rate of each fault type is calculated for each fault position using fault distribution models as stated in Table 1. The whole network basing on actual dimensions in foot is mapped out as shown in Figure 8. The fault positions are assigned with co-ordinates.

Thirdly, voltage sag magnitude and phase shift at all load buses are computed for all fault positions chosen. With the application of fault rate at each fault positions, voltage sag frequency corresponding to different characteristics is obtained. The voltage sag frequency is calculated for

- Individual load bus

- All possible phase loads including phase-to-neutral, phase-to-phase, three phase loads.
- The whole system

Table 2. System fault rate breakdown

Case	Fault rate		Fault distribution model
	Transformer	Line	
N1-Equip.	$200\alpha_{eq}$	$200\alpha_{eq}$	Uniform
N1-Ext.	$200\alpha_{ex}$	$200\alpha_{ex}$	Bivariate Normal
N11-Equip.	$25\alpha_{eq}$	$25\alpha_{eq}$	Uniform
N11-Ext.	$25\alpha_{ex}$	$25\alpha_{ex}$	Bivariate Normal
N2-Equip.	$20\alpha_{eq}$	$20\alpha_{eq}$	Uniform
N2-Ext.	$20\alpha_{ex}$	$20\alpha_{ex}$	Bivariate Normal
N3-Equip.	$5\alpha_{eq}$	$5\alpha_{eq}$	Uniform
N3-Ext.	$5\alpha_{ex}$	$5\alpha_{ex}$	Bivariate Normal

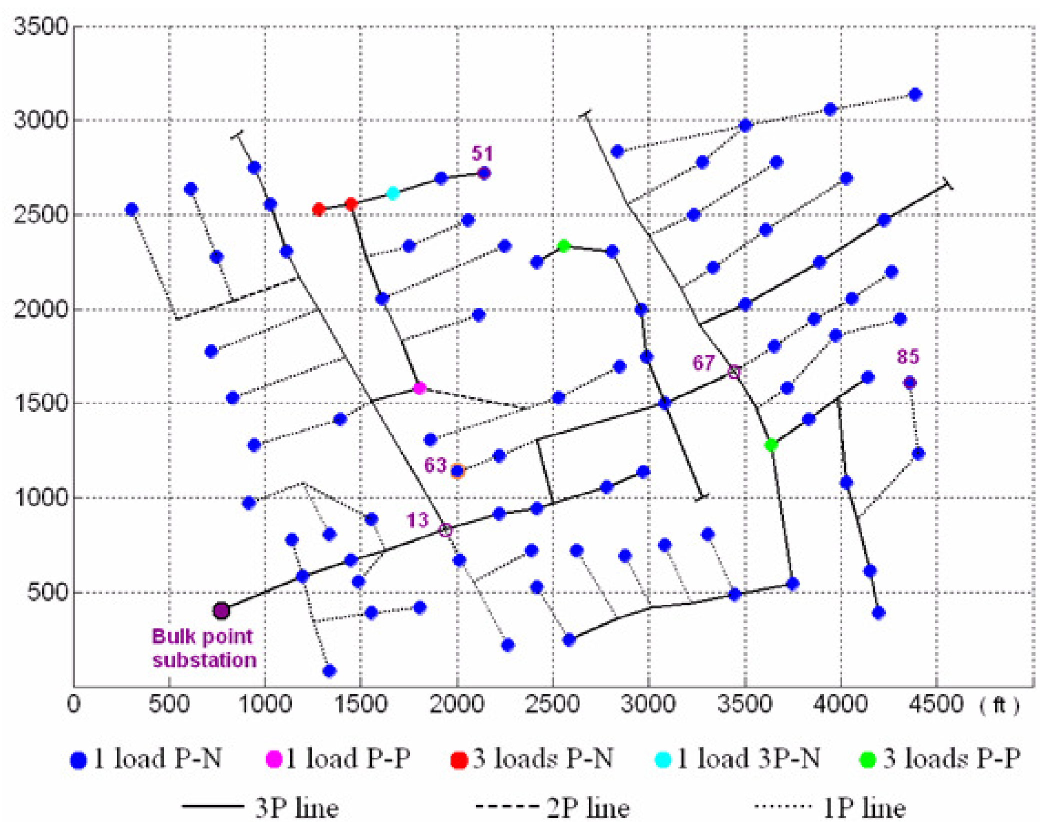

Figure 8. Mapping of IEEE - 123 Bus Radial Distribution Test Feeder

4.4. Development of voltage sag indices

Power quality indices are used to estimate the quality of supplied electric energy for the power system. To date, many power quality indices have been proposed for various power quality events. A well-known index of voltage sag is the System Average Rms Voltage Variation Frequency Index for voltage sag down to under X% of nominal voltage value ($SARFI_X$).

$$SARFI_{X-PN} = \frac{\sum_i M_{X-PN(i)}}{M_{PN}} \quad (12)$$

where

$M_{X(i)}$: Number of sags down to under X% of nominal voltage value that the customer i experiences.

M : Number of customers served from the system of interest.

It's often used for evaluating power quality of a three phase power system basing on monitored limited segmentation [7]. The assessed system is segmented so that every point in the system is contained within a section monitored by an actual power quality measuring instrument.

In distribution system, various phase loads (phase-to-neutral, phase-to-phase, three phases loads) are available and the asymmetrical fault which accounts for almost faults, never results in voltage sag to all single phase loads (e.g. phase A to ground fault may not cause voltage sag to the loads connected between phase B and neutral or phase C and neutral or loads connected between phase B and phase C). Therefore, using $SARFI_X$ for segments of system regardless of number of phases involved may not reflect exactly voltage sag performance of the distribution system. From the customer's point of view, they are more interested in the indices that can estimate voltage sag performance for phase loads. In order to take the availability of various phase loads in the distribution system into account, this paper detailed $SARFI_X$ with regard to phase loads as follows

$$SARFI_{X-PN} = \frac{\sum_i M_{X-PN(i)}}{M_{PN}} \quad (13)$$

$$SARFI_{X-PP} = \frac{\sum_i M_{X-PP(i)}}{M_{PP}} \quad (14)$$

$$SARFI_{X-3P} = \frac{\sum_i M_{X-3P(i)}}{M_{3P}} \quad (15)$$

where

$M_{X\text{-}PN(i)}$, $M_{X\text{-}PP(i)}$, $M_{X\text{-}3P(i)}$: Number of sags down to under X% that phase-to-neutral (A,B,C), phase-to-phase (A-B, B-C, C-A) or three phase load i experiences.

M_{PN} , M_{PP}, M_{3P} : Number of phase-to-neutral (A,B,C), phase-to-phase (A-B, B-C, C-A) or three phase customers served from the system of interest.

4.5. Evaluation of influence of fault distribution modeling on voltage sag performance

The fault distribution modeling uses several parameters. In practice, it's possible to adjust these parameters so that the resulting fault distribution model is suitable to the fault performance of a distribution system of interest. However, the variation of these parameters also makes the voltage sag performance to change accordingly. In modeling fault distribution, the research also considers following options of fault distribution for estimating the influence of fault distribution on voltage sag performance

- Variation in contributory percentage of fault cause due to external factors in total faults which is in fact the variation of α_{eq} (or α_{ex}). In the research, three options : α_{eq} = 0%, 50% and 100% is considered.
- Switching the position of the mean value (μ_x, μ_y) of the bivariate normal distribution. The research considers four options of the mean value at nodes 13, 51, 67 and 85 as indicated in Figure 8.
- Change in deviations σ_x, σ_y of the BND. The research also considers three options of the deviation that equals 0.2, 0.5 and 0.8 of maximum of individual deviations : (Δ_{MAX}=Max{Δx_i,Δy_i,i=1÷m_b}).

V. Result Demonstration And Analysis

Basing on above procedures of stochastic prediction, the followings are remarkable results

5.1. SARFI for phase loads

Indices of voltage sag for different phase loads are shown in Figure 9. They are voltage sag frequency spectrums and corresponding $SARFI_{X\text{-}PN}$, $SARFI_{X\text{-}PP}$ and $SARFI_{X\text{-}3Y}$ in which X ranges from 10% to 90%. In this case study, $(\mu_x,\mu_y)\equiv(x_{13},y_{13})$, $\sigma_x=\sigma_y=0.5\Delta_{MAX}$. Beside that, $SARFI_X$ for the whole system of interest with different mean values (at nodes 13, 51, 67 and 85) of the fault distribution models regardless of the number of involved phases are also depicted in Figure 10. Obviously, there are big differences between $SARFI_X$ of different phase loads and between $SARFI_X$ of phase loads and $SARFI_X$ of the whole system. $SARFI_{X\text{-}PN}$ of phase A, B and C are different because the number of single phase loads connected to individual phases is different. $SARFI_{X\text{-}PN}$ are rather low as single phase loads just expe-rience voltage sag for single phase-to-ground faults on the same phase. In the case of phase-to-phase loads, there's little deep sag frequency meanwhile shallow sag frequency are quite high that makes $SARFI_{X\text{-}PP}$ to suddenly grows from $X \geq 50\%$ and ultimately higher than $SARFI_{X\text{-}PN}$ in total. Main reasons are that phase-to-phase loads are impacted by more faults than phase-to-neutral and almost phase-to-ground faults just cause shallow sags for phase-to-phase loads. $SARFI_{X\text{-}3Y}$ for three phases is the greatest and $SARFI_{X\text{-}3Y}$ for $X \geq 80\%$ equals 500 sags per load because three phase loads will experience voltage sag for any fault types. The above mentioned remarks also explain why $SARFI_X$ defined for phase loads are more useful indices for estimating voltage sag performance in distribution system where many single phase loads exist.

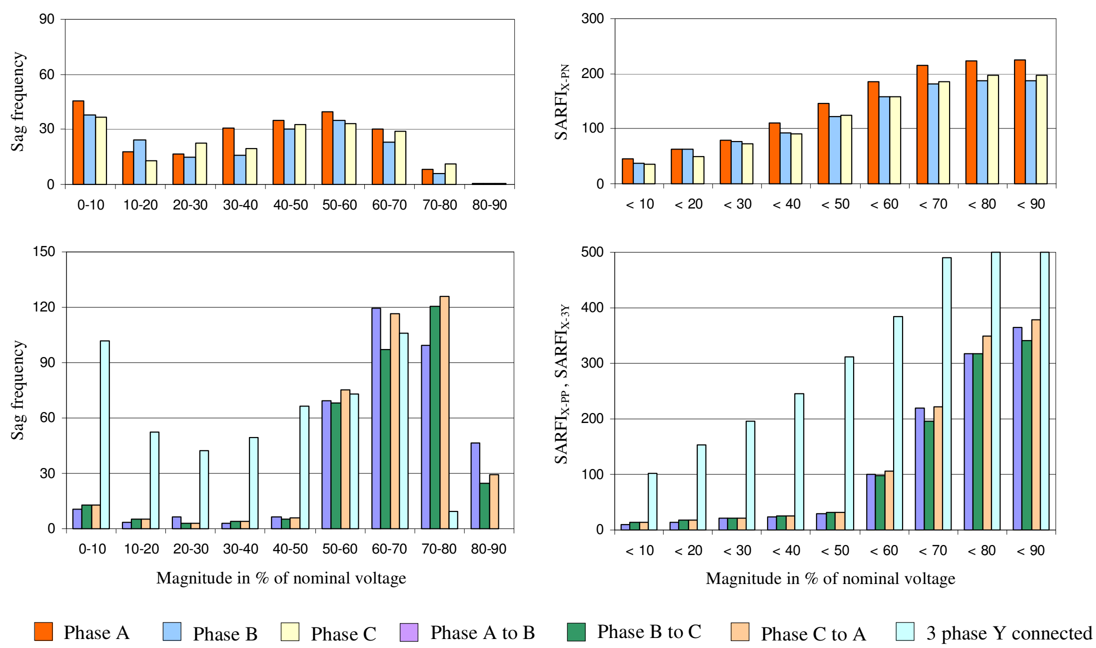

Figure 9. Sag frequency spectrum and $SARFI_X$ of different phase loads for the case the mean value is at node 13 and deviation $\sigma_x=\sigma_y=0.5\Delta_{MAX}$

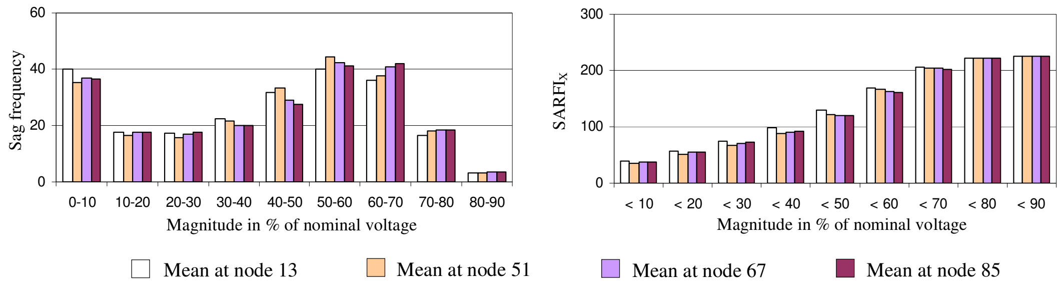

Figure 10. Sag frequency spectrum and SARFI$_X$ of the whole system for different mean positions for the case the deviation is $\sigma_x = \sigma_y = 0.5\Delta_{MAX}$

5.2. The influence of fault distribution modeling on voltage sag performance

Figure 10 also shows that different positions of the mean value of fault distribution models result in different spec-trums of voltage sag frequency. It's notable that if the mean position gets nearer the bulk point, the deep sag frequency will increase. That's mainly because of radial network topo-logy of the distribution system.

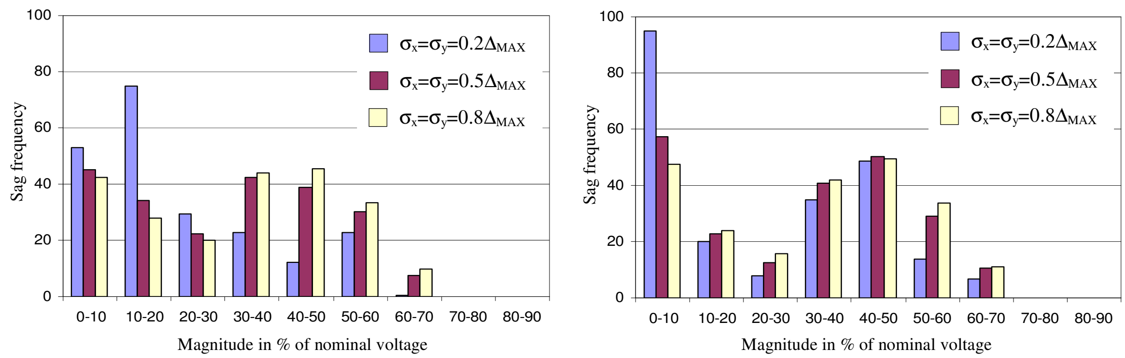

Figure 11. Voltage sag frequency spectrum of the load bus 63 on phase A for different deviations. Mean value is at node 67 (upper) and node 13 (lower)

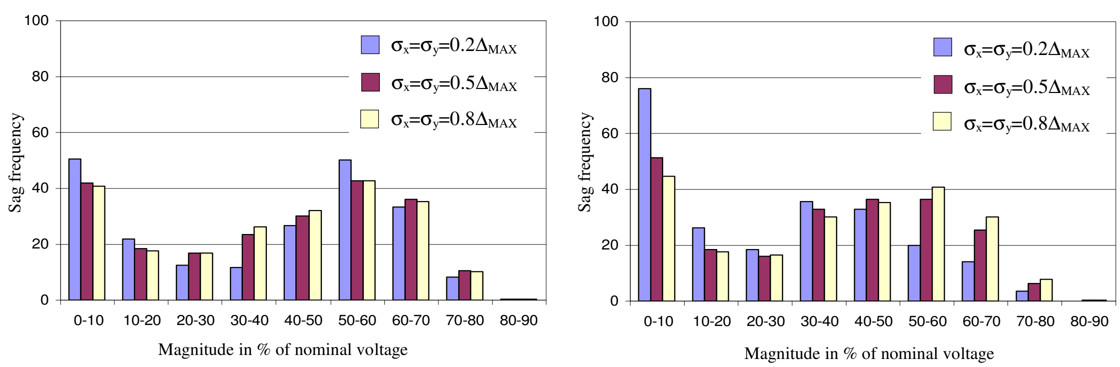

Figure 12. Voltage sag frequency spectrum for loads on phase A for different deviations. Mean value is at node 67 (upper) and node 13 (lower)

Figure 11 and Figure 12 plot the voltage sag frequency for load bus 63 on phase A (see Figure 2) and for all loads on phase A for different deviation values of fault distribution (σ_x=σ_y=0.2,0.5,0.8.Δ_{MAX}) in the case the mean values are identical to the co-ordinates of node 13 and node 67. Similarly, Figure 13 demonstrates voltage sag frequency spectrum and $SARFI_X$ for the whole system also for different deviation values (σ_x=σ_y=0.2,0.5,0.8Δ_{MAX}) and in the case the mean values are at node 13 and node 67. Increasing the deviation values σ_x and σ_y will turn the normal distribution into the uniform distribution. This change causes shape variations to the voltage sag frequency spectrum. The clear increase of deep sag frequency is shown for all cases of sag performance demonstration. If the mean position of the distribution model locates at the node 13 which is very near the bulk point, the frequency of sag below 10% even raises about 50% for the small deviation (σ_x=σ_y=0.2Δ_{MAX}). That's also explained as the result of radial network topology of the distribution system as above said.

The spectrum of voltage sag frequency for different case studies (from Fig. 11 to Fig. 13) are quite similar in which deep sag account for a large number mainly due to short feeders in distribution system. 0.4 to 0.6 p.u. sag frequency is also high as the network topology consists of one trunk line with many lateral taps in the middle. That means the point of common coupling of a large number of loads is on the middle of the trunk line. Few loads connected to trunk line near the bulk point (intermediate substation location) explain why the shallow sag frequency is very low. Figure 14 gives us a closer look on voltage sag frequency distribution for different sag magnitude. It's undoubted that the deep sag frequency appears at nodes on branches connected nearly to the far end of the trunk line. Sag down to 0.3 to 0.5 p.u. distributed rather uniformly excepting nodes near the bulk point. The shallow sag frequency mainly occurs at several nodes near the bulk point of supply.

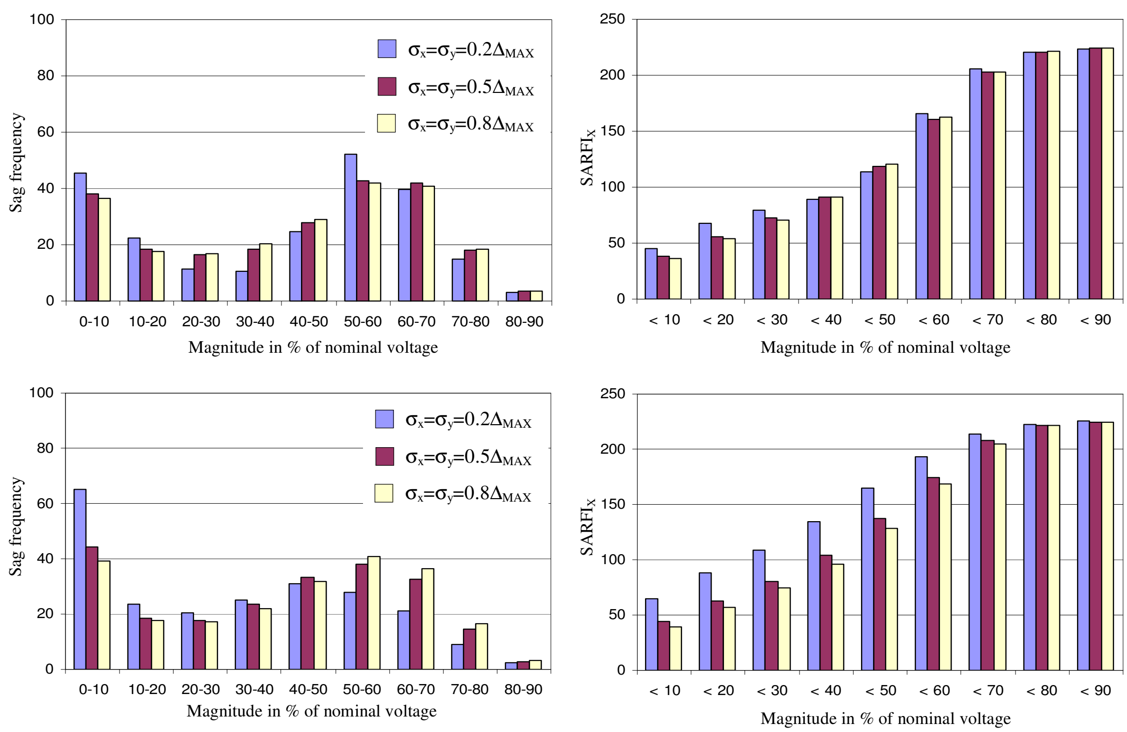

Figure 13. Voltage sag frequency spectrum and $SARFI_X$ for the whole system for different deviations for mean value at node 67 (two upper) and node 13 (two lower)

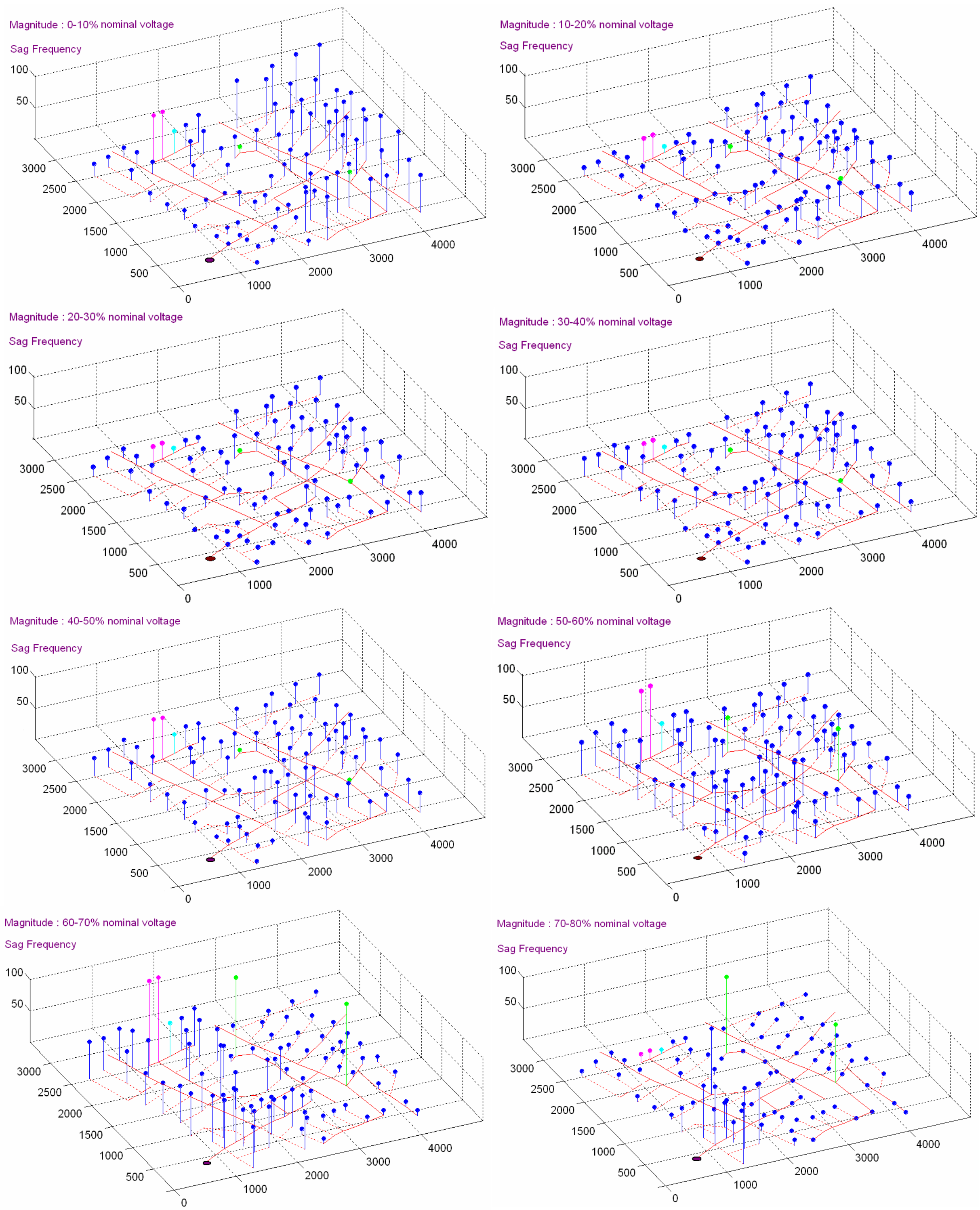

Figure 14. Voltage sag frequency distribution for different ranges of voltage sag magnitude

VI. CONCLUSIONS

This research presented a new case study of fault distribution modeling in stochastic prediction of voltage sag for the distribution system.

Using two-dimension stochastic models such as BND for modeling fault distribution allows to obtain a good overview of fault performance of the whole system of interest. Thus, it's possible not only to analyse the relation between faults and voltage sags at individual locations of the system like a specific load bus or a segment of line, but also to compute system indices of voltage sags like SARFI. The application of two-dimension stochastic models has some limits to the size of the system. For the segments of distribution system of which the size is not large such as the distribution system supplied from a bulk substation, it's practical to use this fault model.

When applying BND for modeling fault distribution, parameters of the distribution model should be selected properly to match with actual fault performance of the system of interest. The research also illustrates of the influence of the change in parameters on voltage sag performance. Altering the deviation value of the distribution model has stronger influence on sag performance especially for deep sag fre-quency pattern than switching the position of mean value. Concentrative occurrence of faults on one location in dis-tribution system will increase the number of deep sags. The results of analysis also evidences that typical radial network topology of distribution system is also another main reason for high frequency of deep sag.

The presence of different phase loads in distribution system indicated that using $SARFI_X$ for the whole system without considering the number of phase of the loads is pro-bably not to reflect exactly sag performance. To have a better assessment of voltage sag regarding phase loads, the research also develops the known rms voltage variation index - $SARFI_X$ into that regarding phase loads. The results proved that there's big different between $SARFI_{X\text{-}PN}$, $SARFI_{X\text{-}PP}$ and $SARFI_{X\text{-}3P}$ for different phase loads and $SARFI_X$ for the whole system regardless of phase loads. This development of $SARFI_X$ is more practical from the customer's point of view when power supply contract is set up.

BIBLIOGRAPHY

[1] M.H.J. Bollen, *Understanding power quality problems - voltage sags and interruptions*, IEEE Press, 2000.

[2] R.C.Dugan, M.F.McGranaghan, and H.W.Beaty, *Electric Power System Quality*, New York : McGraw-Hill, 1996.

[3] R.E.Brown, *Electric Power Distribution Reliability*, Marcel-Dekker, Inc., 2002.

[4] W.H.Kersting, *Distribution System Modeling and Analysis*, CRC Press LLC, 2002.

[5] T.A.Short, *Electric Power Distribution Handbook*, CRC Press, 2004.

[6] G.Olguin, "Voltage Dip (Sag) Estimation in Power System Based on Stochastic Assessment and Optimal Monitoring", Ph.D. dissertation, Dept. Eneg. Envir., Div. Elect. Power Eng., Chamber Univ. Techno., 2005.

[7] D. L. Brooks, R. C. Dugan, Marek Waclawiak, Ashok Sundaram, "Indices for Assessing Utility Distribution System RMS Variation Performance", *IEEE Transactions on Power Delivery*, Vol.13, No.1, p.254-259, January 1998.

[8] M.R.Qader, M.H.J.Bollen, and R.N.Allan, "Stochastic Prediction of Voltage Sags in a Large Transmission System", *IEEE Transactions on Industry Applications*, Vol.35, No.1, p.152-162, January/February 1999,

[9] M.R.Qader, M.H.J. Bollen and R.N.Allan, "Stochastic Prediction of Voltage Sags in Reliability Test System", PQA-97 Europe, Elforsk, Stockholm, Sweden, June 1997,

[10] J.A.Martinez-Velasco, J. Martin-Arnedo, "Stochastic Prediction of Voltage Dips Using an Electromagnetic Transient Program", Session 24, Paper 4, 14th PSCC, Sevilla, Jun. 2002.

[11] J.V.Milanovic, M.T.Aung and C.P.Gupta, "The Influence of Fault Distribution on Stochastic Prediction of Voltage Sags", *IEEE Transactions on Power Delivery*, Vol.20, No.1, p.278-285, January 2005.

[12] IEEE Distribution Planning Working Group Report, "Radial Dis-tribution Test Feeder" *IEEE Transaction on Power Systems*, Vol. 6, No. 3, p.975-985, August 1991.

[13] Y. Viniotis, *Probability and Random Processes for Electrical Engineers*, McGraw-Hill, 1998

[14] R. D. Yates, D. J. Goodman, Probability and Stochastic Processes – A Friendly Introduction for Electrical and Computer Engineers, John Wiley & Sons, Inc. 2005.

APPENDIX A

PROGRAM OF SAGS CALCULATION IN DISTRIBUTION SYSTEM

A.1. Introduction

The program of sags calculation in distribution system is edited by the author on Matlab Code basing on modeling technique of distribution system for short-circuit calculation in distribution system [4]. In this model, all circuit components are modeled in the form of three phase matrices. It can calculate all possible type of short-circuit in distribution system for any fault location. It can be applicable just for distribution system with radical or open-ring network topology. No application is given for ring or loopy network topology. The whole package of program consists of a main module for calculating fault current at fault location, nodal voltages for all buses and several support data files which must be compiled in advance in a suitable form.

A.2. Distribution system short-circuit modeling

A.2.1. Components modeling

<u>a. For a fraction of line</u>, the generalized matrices are as follows

$$\begin{cases} [VLG_{abc}]_n = [a].[VLG_{abc}]_m + [b].[I_{abc}]_m \\ [I_{abc}]_n = [c].[VLG_{abc}]_m + [d].[I_{abc}]_m \end{cases} \quad \text{(a1)}$$

where

$[VLG_{abc}]_n$; $[VLG_{abc}]_m$: The matrices of line phase-to-ground voltage at node n and m.

$[I_{abc}]_n$, $[I_{abc}]_m$: The matrices of phase current coming in node n and outgoing from node m.

$$[a] = [U] + ½.[Z_{abc}].[Y_{abc}] \qquad [c] = [Y_{abc}] + ¼.[Y_{abc}].[Z_{abc}].[Y_{abc}]$$

$$[b] = [Z_{abc}] \qquad [d] = [U] + ½.[Z_{abc}].[Y_{abc}]$$

$$[U] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \qquad [Z_{abc}] = \begin{bmatrix} Z_{aa} & Z_{ab} & Z_{ac} \\ Z_{ba} & Z_{bb} & Z_{bc} \\ Z_{ca} & Z_{cb} & Z_{cc} \end{bmatrix}; \qquad [Y_{abc}] = \begin{bmatrix} Y_{aa} & Y_{ab} & Y_{ac} \\ Y_{ba} & Y_{bb} & Y_{bc} \\ Y_{ca} & Y_{cb} & Y_{cc} \end{bmatrix}$$

$$[VLG_{abc}] = \begin{bmatrix} VLG_a \\ VLG_b \\ VLG_c \end{bmatrix}; \qquad [I_{abc}] = \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}$$

In the case of distribution system, $[Y_{abc}]$=[0], $[a]$=$[d]$=$[U]$, $[b]$=$[Z_{abc}]$, $[c]$=[0], it can be simplified as follows

$$\begin{cases} [VLG_{abc}]_n = [VLG_{abc}]_m + [Z_{abc}].[I_{abc}]_m \\ [I_{abc}]_n = [I_{abc}]_m \end{cases} \quad \text{(a2)}$$

<u>b. For a distribution transformer</u>, the generalized matrices are as follows

$$\begin{cases} [VLN_{ABC}] = [a_t].[VLN_{abc}] + [b_t].[I_{abc}] \\ [I_{ABC}] = [c_t].[VLN_{abc}] + [d_t].[I_{abc}] \end{cases} \quad \text{(a3)}$$

where

[VLN_{ABC}] ; [VLN_{abc}] : The matrices of line phase-to-neutral primary and secondary voltages.

[I_{ABC}], [I_{abc}] : The matrices of primary and secondary currents.

[a_t], [b_t], [c_t] and [d_t] are the matrices of factors which are defined depending on the three phase connection of the transformer. For the delta-grounded wye step-down connection transformer, these matrices are as follows

[a_t] = [W].[AV]

where

$$[W] = [A_s].[T].[A_s]^{-1}; \qquad [A_s] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & a_s^2 & a_s \\ 1 & a_s & a_s^2 \end{bmatrix}; \ a_s = 1\angle 120^0;$$

$$[T] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & t_s^* & 0 \\ 0 & 0 & t_s \end{bmatrix}; \ t_s = \frac{1}{\sqrt{3}}\angle 30^0; \quad [VA] = \begin{bmatrix} 0 & -n_t & 0 \\ 0 & 0 & -n_t \\ -n_t & 0 & 0 \end{bmatrix}; \ n_t = \frac{VLL_{Rated-Primary}}{VLN_{Rated-Secondary}}$$

$$[b_t] = [a_t].[Zt_{abc}]; \ [Zt_{abc}] = \begin{bmatrix} Zt_a & 0 & 0 \\ 0 & Zt_b & 0 \\ 0 & 0 & Zt_c \end{bmatrix}$$

A.2.2. Matrix equation for short-circuit calculation

Figure 10 illustrates the Thevenin equivalent circuit at the fault node.

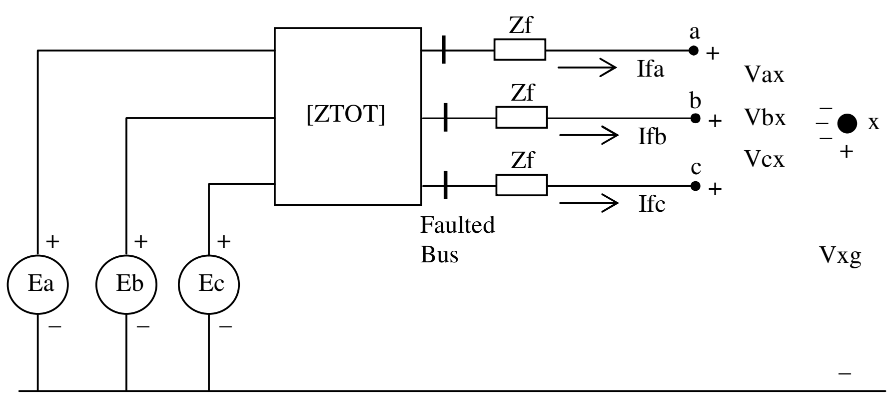

Figure A.1. Thevenin equivalent circuit

The generalized matrices equation for short-circuit calculation is as follows

$$[IP_{abc}] = [If_{abc}] + [Y].[V_{abcx}] + [Y].[V_{xg}] \qquad \text{(a4)}$$

where

[Y] = [ZEQ]$^{-1}$, [ZEQ] = [ZTOT] + [ZF]

$$[ZF]=\begin{bmatrix} Z_f & 0 & 0 \\ 0 & Z_f & 0 \\ 0 & 0 & Z_f \end{bmatrix}$$; Matrix of fault impedance. $$[ZTOT]=\begin{bmatrix} Z_{aa} & Z_{ab} & Z_{ac} \\ Z_{ba} & Z_{bb} & Z_{bc} \\ Z_{ca} & Z_{cb} & Z_{cc} \end{bmatrix}$$

[IP_{abc}] = [Y] . [E_{abc}] ; $[E_{abc}]=\begin{bmatrix} E_a \\ E_b \\ E_c \end{bmatrix}$

$$\begin{cases} IP_a = If_a + (Y_{aa}\cdot V_{ax} + Y_{ab}\cdot V_{bx} + Y_{ac}\cdot V_{cx}) + Ys_a \cdot V_{xg} \\ IP_b = If_b + (Y_{ba}\cdot V_{ax} + Y_{bb}\cdot V_{bx} + Y_{bc}\cdot V_{cx}) + Ys_b \cdot V_{xg} \\ IP_c = If_c + (Y_{ca}\cdot V_{ax} + Y_{cb}\cdot V_{bx} + Y_{cc}\cdot V_{cx}) + Ys_c \cdot V_{xg} \end{cases} \quad \text{where} \quad \begin{cases} Ys_a = Y_{aa} + Y_{ab} + Y_{ac} \\ Ys_b = Y_{ba} + Y_{bb} + Y_{bc} \\ Ys_c = Y_{ca} + Y_{cb} + Y_{cc} \end{cases} \quad (a.5)$$

Fault type definitions

For different fault types, following additional equations are set :

- Three-phase faults : $V_{ax} = V_{bx} = V_{cx} = 0$; $I_a + I_b + I_c = 0$ (a.6)
- Three phase – to – ground faults : $V_{ax} = V_{bx} = V_{cx} = V_{xg} = 0$ (a.7)
- Line – to – line faults
  (assume i-j fault with phase k unfaulted) : $V_{ix} = V_{jx} = 0$; $If_k=0$; $If_i + If_j = 0$ (a.8)
- Line – to – ground faults
  (assume phase k fault with phase I and j unfaulted) : $V_{kx} = V_{xg} = 0$; $If_i = If_j = 0$ (a.9)

Combine (a.6) to (a.9) with (a.5) we can obtain the generalized matrix equation for fault calculation in distribution system as follows

$$\begin{bmatrix} IP_a \\ IP_b \\ IP_c \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & Y_{aa} & Y_{ab} & Y_{ac} & Ys_a \\ 0 & 1 & 0 & Y_{ba} & Y_{bb} & Y_{bc} & Ys_b \\ 0 & 0 & 1 & Y_{ca} & Y_{cb} & Y_{cc} & Ys_c \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \\ - & - & - & - & - & - & - \end{bmatrix} \cdot \begin{bmatrix} If_a \\ If_b \\ If_c \\ V_{ax} \\ V_{bx} \\ V_{cx} \\ V_{xg} \end{bmatrix} \quad \text{or} \quad [IPs] = [C] . [X] \quad (a.10)$$

"-" will be filled with 1 or 0 depending on the fault type defined as (a.6) to (a.9).

Solve (a.10), fault current and phase voltage at the fault position will be determined.

A.3. Block diagram

Block-diagram of the program is plotted as Figure A.2.

Due to the radial network topology, the fault occurs at a location will create a big fault current flowing from the bulk point substation to the fault location on a set of segments of line which is called fault-line. The currents on any other segments of line are assumed to equal zero. For example, on Figure A.3 (the numbering of IEEE 123 bus radial testfeeder will be explained in annex 2), if the short-circuit occurs at node 123, the fault line is the set of the segments of line : 122, 121, 119, 117, 115, 113, 112, 103, 98, 72, 71, 63, 60, 57, 56, 55, 54, 14, 8, 7, 1. Therefore, the first block of the diagram is to find all nodes and branches on fault current carrying line.

The second step is to calculate fault current and phase voltages at the fault location using (a4). Then phase voltage for all other nodes on the fault line is also computed using (a2) and (a3).

The third step is to calculate voltage for nodes not on the fault line. To do so, it needs to trace out the shortest circuit leading from the node of interest to the node on the fault line. For example, again look at the Figure a2, if we want to calculate the voltage at node 35, firstly, we find the closest way from the node 35 to a node on the fault line. That way consists of a set of segments of line : 35, 34, 33, 31, 28, 26, 24, 23, 22, 19 and the node on the fault line is 15. After finding out the said circuit from the node of interest to the node on the fault line, voltage at the node of interest will be determined using (a2) with the assumption that $[I_{abc}]=[0]$.

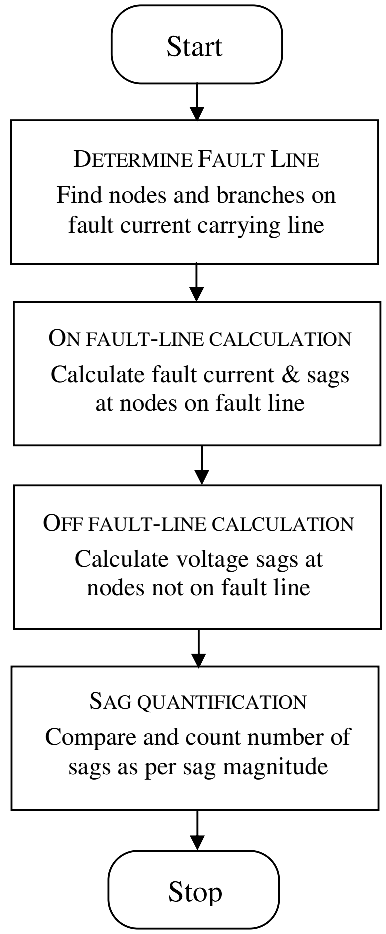

Figure A.2. The block-diagram of the program of sag calculation in distribution systems

Finally, voltage sag at all nodes will be compared with different characteriestics of voltage sag magnitudes. When applying the fault frequency for a fault position, resulting sag frequency for all other nodes will be calculated. Sag frequency at a node of load will be obtained when fault frequency of all fault locations is applied.

A.4. Main module in Matlab Code

```
function SAG_FEEDER123IEEE
% TINH TOAN THONG KE DU BAO SO SAG DIEN AP TRONG LUOI PHAN PHOI 123 NUT CUA IEEE
% Bach Quoc Khanh / BM He Thong Dien - DHBKHN 5.2005

clear all;
clc;
% 1. Nhap so lieu cau truc luoi dien :

% Nhap so lieu danh dau co dien ap pha tai cac nut tai
load phase123abc.txt;
Phavol = phase123abc;

```

```
% Nhap so lieu nhan dang dien ap day va dien ap pha cua nut tai
load load123abc.txt;
loadpha = load123abc;
[Sag_Row Sag_Col]=size(loadpha);

% Nhap so lieu tong tro (TT) cac doan duong day theo don vi tuong doi
% Base : 5000kVA - 4.16kV
load linereal123.txt;  % Phan thuc
load lineimag123.txt;  % Phan ao
branch_no = max(size(linereal123));

% Tinh TT duong day
Zline = linereal123 +i*lineimag123;

% Nhap so lieu ve so do luoi (ket noi giua cac nut)
load testfeeder123sys.txt;

System_mat = testfeeder123sys;
[Sys_Row Sys_Col]= size(System_mat);
% -----------------------------------------------------
% 2. SO LIEU LIEN QUAN TINH NGAN MACH
% Cong suat ngan mach phia cao ap (115kV) MVA :
SC_MVA = 1100;
SC_angle = 82;
MVA3pha = SC_MVA*(cos(SC_angle/180*pi)+j*sin(SC_angle/180*pi))/5; % CSNM tuong doi

% Thong so MBA-TBATG :
% MVA_transfo = 5000(KVA)
% VLL_high = 115(kV)
% VLL_low  = 4.16(kV)
zt = (1+j*8)/100; % Tong tro MBA
% -----------------------------------------------------
% 3. Tinh toan tong tro thay the (Thevenin)
alpha =-0.5+j*0.866; %Toan tu xoay 120o
alpha2=-0.5-j*0.866; %Toan tu xoay 240o
As = [1 1 1;1 alpha2 alpha;1 alpha alpha2];
U  = [1 0 0;0 1 0; 0 0 1];
rt3 = sqrt(3);

AV = rt3*[0 -1 0; 0 0 -1; -1 0 0];
% MT ty so bien dien ap dang tuong doi (VLL_cao=AV*VLN_ha)
AI = 1/rt3*U;
% MT ty so bien dong dien dang tuong doi (ID_cao=AI*VLN_ha)
D  = [1 -1 0; 0 1 -1; -1 0 1];
% MT quan he Uday/Upha (VLL_cao = At.VLN_cao)
ts = (cos(pi/6)+j*sin(pi/6))/sqrt(3);
T  = [1 0 0; 0 conj(ts) 0; 0 0 ts];
W  = As*T*inv(As);

% Tinh ma tran tong tro tuong duong cua he thong
Zsys_posi = 1/conj(MVA3pha);
Zsys_zero = 3*Zsys_posi;
Zseq = [Zsys_zero 0 0; 0 Zsys_posi 0; 0 0 Zsys_posi];
Zabc_sys = As*Zseq*inv(As);

% Tinh ma tran tong tro MBA
Ztabc = [zt 0 0; 0 zt 0; 0 0 zt];

% Ma tran tong quat cua MBA Y/d-11
at = W*AV;
bt = at*Ztabc;
ct = zeros(3,3);
dt = D*AI;
A_t= inv(AV)*D;

% Qui doi ma tran tong tro he thong ve thu cap TBATG (D/yn-11)
```

```
Zsource = A_t*Zabc_sys*dt + Ztabc;

% 4. Dien ap Thevenin qui ve diem ngan mach (dang tuong doi so voi U = 4.16kV)
ELL_source = [1; alpha2; alpha];
ELN_source = W*ELL_source;
Eth = A_t*ELN_source;

load load_coordinate.txt;
Xnode = load_coordinate(:,1);
Ynode = load_coordinate(:,2);

disp('CHUONG TRINH TINH TOAN SAG DIEN AP TRONG LUOI PHAN PHOI');
disp('        B.Q.Khanh - HTD/DHBKHN (5.2005)');
disp(' ');
disp('Xin hay doi trong 25 giay !');
% -------------------------------------------------------------------
% TINH TOAN SAG DO SU CO MBAPP
% Nhap so lieu cac dang ngan mach

load faulttype_line.txt;
[FauLoNo1 FauTyNo1]= size(faulttype_line);
% Fault Location Number & Fault Type Number at each location
Faultcode1 = faulttype_line;

% Nhap so lieu ty le su co cac nhanh duong day
load faultrate_line73_50.txt;
rate1=faultrate_line73_50;

% Kiem tra muc do sag
sag0_10DD =zeros(Sag_Row,Sag_Col);
sag10_20DD=zeros(Sag_Row,Sag_Col);
sag20_30DD=zeros(Sag_Row,Sag_Col);
sag30_40DD=zeros(Sag_Row,Sag_Col);
sag40_50DD=zeros(Sag_Row,Sag_Col);
sag50_60DD=zeros(Sag_Row,Sag_Col);
sag60_70DD=zeros(Sag_Row,Sag_Col);
sag70_80DD=zeros(Sag_Row,Sag_Col);
sag80_90DD=zeros(Sag_Row,Sag_Col);

for branch123=1:FauLoNo1
   % Tinh tong tro duong day tu thanh cai ha ap TBATG den diem NM
   SC_bus = branch123+1;
   % Gia thiet su co tren duong day xay ra o cuoi cac nhanh.
   % VD : Xet su co tren nhanh 1 xay ra tai nut 2 o cuoi nhanh 1
   nut(1) = SC_bus;
   v=1;
   while SC_bus ~=1
      v=v+1;
      for ih=1:Sys_Row;
         for il=1:Sys_Col;
            if System_mat(ih,il)==SC_bus;
               nut(v)=ih;
               SC_bus=ih;
            end
         end
      end
   end
   Zm = zeros(3,3);
   for ik=1:v-1
      Zm(1,1) = Zm(1,1) + Zline(nut(ik)-1,1);
      Zm(1,2) = Zm(1,2) + Zline(nut(ik)-1,2);
      Zm(1,3) = Zm(1,3) + Zline(nut(ik)-1,3);
      Zm(2,1) = Zm(1,2);
      Zm(2,2) = Zm(2,2) + Zline(nut(ik)-1,4);
      Zm(2,3) = Zm(2,3) + Zline(nut(ik)-1,5);
      Zm(3,1) = Zm(1,3);
```

```
    Zm(3,2) = Zm(2,3);
    Zm(3,3) = Zm(3,3) + Zline(nut(ik)-1,6);
end
% Tong tro Thevenin tu nguon den diem ngan mach
ZTOT = Zsource + Zm;

% --------------------------------------------------
% 5. Tinh dong dien ngan mach
for Ftype=1:FauTyNo1
    SC_code=Faultcode1(branch123,Ftype);
    if SC_code ~= 0
        Zf = 0;
        Zfault = [Zf 0 0;0 Zf 0;0 0 Zf];
        ZEQ = ZTOT+Zfault;
        YTOT = inv(ZEQ);
        Ysabc = YTOT(:,1)+YTOT(:,2)+YTOT(:,3);

        IPabc = YTOT*Eth;

        C1 = [U,YTOT,Ysabc];
        C2 = zeros(4,7);

        switch SC_code
            case 1 % Ib = Ic = 0, Ua = Ug = 0
                C2(1,2)=1; C2(2,3)=1; C2(3,4)=1; C2(4,7)=1;
            case 2 % Ia = Ic = 0, Ub = Ug = 0
                C2(1,1)=1; C2(2,3)=1; C2(3,5)=1; C2(4,7)=1;
            case 3 % Ia = Ib = 0, Uc = Ug = 0
                C2(1,1)=1; C2(2,2)=1; C2(3,6)=1; C2(4,7)=1;
            case 4 % Ia+Ib = 0, Ic = 0, Ua = Ub = 0
                C2(1,1)=1; C2(1,2)=1; C2(2,3)=1; C2(3,4)=1; C2(4,5)=1;
            case 5 % Ib+Ic = 0, Ia = 0, Ub = Uc = 0
                C2(1,2)=1; C2(1,3)=1; C2(2,1)=1; C2(3,5)=1; C2(4,6)=1;
            case 6 % Ia+Ic = 0, Ib = 0, Ua = Uc = 0
                C2(1,1)=1; C2(1,3)=1; C2(2,2)=1; C2(3,4)=1; C2(4,6)=1;
            case 7 % Ic = 0, Ua = Ub = Ug = 0
                C2(1,3)=1; C2(2,4)=1; C2(3,5)=1; C2(4,7)=1;
            case 8 % Ia = 0, Ub = Uc = Ug = 0
                C2(1,1)=1; C2(2,5)=1; C2(3,6)=1; C2(4,7)=1;
            case 9 % Ib = 0, Ua = Uc = Ug = 0
                C2(1,2)=1; C2(2,4)=1; C2(3,6)=1; C2(4,7)=1;
            case 10 % Ua = Uc = Uc = Ug = 0
                C2(1,4)=1; C2(2,5)=1; C2(3,6)=1; C2(4,7)=1;
            case 11 % Ia+Ib+Ic=0 Ua = Uc = Uc = 0
                C2(1,1)=1; C2(1,2)=1; C2(1,3)=1; C2(2,4)=1; C2(3,5)=1; C2(4,6)=1;
            otherwise
        end
        C=[C1;C2];

        IP = [IPabc;0;0;0;0];
        X = inv(C)*IP;

        % 6. Tinh dien ap cac nut tren nhanh ngan mach tro ve nguon

        Ifabc(1,1) = X(1,1);
        Ifabc(2,1) = X(2,1);
        Ifabc(3,1) = X(3,1);
        Vfabc(1,1)  = X(4,1) + X(7,1);
        Vfabc(2,1)  = X(5,1) + X(7,1);
        Vfabc(3,1)  = X(6,1) + X(7,1);

        for ij=1:v-1
            Zn(1,1)=Zline(nut(ij)-1,1);
            Zn(1,2)=Zline(nut(ij)-1,2);
            Zn(1,3)=Zline(nut(ij)-1,3);
            Zn(2,1)=Zn(1,2);
```

```
            Zn(2,2)=Zline(nut(ij)-1,4);
            Zn(2,3)=Zline(nut(ij)-1,5);
            Zn(3,1)=Zn(1,3);
            Zn(3,2)=Zn(2,3);
            Zn(3,3)=Zline(nut(ij)-1,6);

            An = U;
            Bn = Zn;
            Cn = zeros(3,3);
            Dn = U;
            Vfabc(:,ij+1) = An*Vfabc(:,ij)+Bn*Ifabc(:,ij);
            Ifabc(:,ij+1) = Cn*Vfabc(:,ij)+Dn*Ifabc(:,ij);

            for pha=1:3
                if Phavol(nut(ij),pha) ~= 1;
                    Vfabc(pha,ij) = 0;
                end
            end
        end
        for ig=1:v
            Vabc(:,nut(ig))=Vfabc(:,ig);
        end


        % Tinh dien ap cac nut khong nam tren tuyen ngan mach
        for ii=1:Sys_Row
            kt=0;
            node = ii;
            for in=1:v
                if node == nut(in)
                    kt=1;
                end
            end
            if kt ~= 1
                kt1 = 0;
                % Tim duong noi tu nut ii toi nut gan nhat tren tuyen NM
                while kt1 ~=1
                    for ir=1:Sys_Row;
                        for ic=1:Sys_Col;
                            if System_mat(ir,ic)==node
                                node=ir;
                            end
                        end
                    end
                    for iv=1:v
                        if node == nut(iv)
                            kt1=1;
                            v_node = iv;
                        end
                    end
                end
                Vabc(:,ii)=Vfabc(:,v_node);
                for pha_i=1:3
                    if Phavol(ii,pha_i) ==0
                        Vabc(pha_i,ii)=0;
                    end
                end
            else
            end
        end

        % Kiem tra muc do sag

        VLLabc=zeros(3,Sys_Row);
        for iy=1:Sys_Row
            if Phavol(iy,4)==1
```

```
            VLLabc(1,iy)=Vabc(1,iy)-Vabc(2,iy);
        end
        if Phavol(iy,5)==1
            VLLabc(2,iy)=Vabc(2,iy)-Vabc(3,iy);
        end
        if Phavol(iy,6)==1
            VLLabc(3,iy)=Vabc(3,iy)-Vabc(1,iy);
        end
    end

    for is=1:Sys_Row
        % Kiem tra so pha cua nut
        if loadpha(is,1)==1;
            modulsag = abs(Vabc(1,is)*sqrt(3));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10DD(is,1)=sag0_10DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20DD(is,1)=sag10_20DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30DD(is,1)=sag20_30DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40DD(is,1)=sag30_40DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50DD(is,1)=sag40_50DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60DD(is,1)=sag50_60DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70DD(is,1)=sag60_70DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80DD(is,1)=sag70_80DD(is,1)+rate1(branch123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
                sag80_90DD(is,1)=sag80_90DD(is,1)+rate1(branch123,Ftype);
            end
        end
        if loadpha(is,2)==1;
            modulsag = abs(Vabc(2,is)*sqrt(3));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10DD(is,2)=sag0_10DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20DD(is,2)=sag10_20DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30DD(is,2)=sag20_30DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40DD(is,2)=sag30_40DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50DD(is,2)=sag40_50DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60DD(is,2)=sag50_60DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70DD(is,2)=sag60_70DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80DD(is,2)=sag70_80DD(is,2)+rate1(branch123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
                sag80_90DD(is,2)=sag80_90DD(is,2)+rate1(branch123,Ftype);
            end
        end
        if loadpha(is,3)==1;
            modulsag = abs(Vabc(3,is)*sqrt(3));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10DD(is,3)=sag0_10DD(is,3)+rate1(branch123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20DD(is,3)=sag10_20DD(is,3)+rate1(branch123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
```

```
        sag20_30DD(is,3)=sag20_30DD(is,3)+rate1(branch123,Ftype);
    elseif modulsag <=0.4 & modulsag >0.3
        sag30_40DD(is,3)=sag30_40DD(is,3)+rate1(branch123,Ftype);
    elseif modulsag <=0.5 & modulsag >0.4
        sag40_50DD(is,3)=sag40_50DD(is,3)+rate1(branch123,Ftype);
    elseif modulsag <=0.6 & modulsag >0.5
        sag50_60DD(is,3)=sag50_60DD(is,3)+rate1(branch123,Ftype);
    elseif modulsag <=0.7 & modulsag >0.6
        sag60_70DD(is,3)=sag60_70DD(is,3)+rate1(branch123,Ftype);
    elseif modulsag <=0.8 & modulsag >0.7
        sag70_80DD(is,3)=sag70_80DD(is,3)+rate1(branch123,Ftype);
    elseif modulsag <=0.9 & modulsag >0.8
        sag80_90DD(is,3)=sag80_90DD(is,3)+rate1(branch123,Ftype);
    end
end
if loadpha(is,4)==1;
    modulsag = abs(VLLabc(1,is));
    % Sag cumulation
    if modulsag <= 0.1
        sag0_10DD(is,4)=sag0_10DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.2 & modulsag >0.1
        sag10_20DD(is,4)=sag10_20DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.3 & modulsag >0.2
        sag20_30DD(is,4)=sag20_30DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.4 & modulsag >0.3
        sag30_40DD(is,4)=sag30_40DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.5 & modulsag >0.4
        sag40_50DD(is,4)=sag40_50DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.6 & modulsag >0.5
        sag50_60DD(is,4)=sag50_60DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.7 & modulsag >0.6
        sag60_70DD(is,4)=sag60_70DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.8 & modulsag >0.7
        sag70_80DD(is,4)=sag70_80DD(is,4)+rate1(branch123,Ftype);
    elseif modulsag <=0.9 & modulsag >0.8
        sag80_90DD(is,4)=sag80_90DD(is,4)+rate1(branch123,Ftype);
    end
end
if loadpha(is,5)==1;
    modulsag = abs(VLLabc(2,is));
    % Sag cumulation
    if modulsag <= 0.1
        sag0_10DD(is,5)=sag0_10DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.2 & modulsag >0.1
        sag10_20DD(is,5)=sag10_20DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.3 & modulsag >0.2
        sag20_30DD(is,5)=sag20_30DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.4 & modulsag >0.3
        sag30_40DD(is,5)=sag30_40DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.5 & modulsag >0.4
        sag40_50DD(is,5)=sag40_50DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.6 & modulsag >0.5
        sag50_60DD(is,5)=sag50_60DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.7 & modulsag >0.6
        sag60_70DD(is,5)=sag60_70DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.8 & modulsag >0.7
        sag70_80DD(is,5)=sag70_80DD(is,5)+rate1(branch123,Ftype);
    elseif modulsag <=0.9 & modulsag >0.8
        sag80_90DD(is,5)=sag80_90DD(is,5)+rate1(branch123,Ftype);
    end
end
if loadpha(is,6)==1;
    modulsag = abs(VLLabc(3,is));
    % Sag cumulation
    if modulsag <= 0.1
        sag0_10DD(is,6)=sag0_10DD(is,6)+rate1(branch123,Ftype);
```

```
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20DD(is,6)=sag10_20DD(is,6)+rate1(branch123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30DD(is,6)=sag20_30DD(is,6)+rate1(branch123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40DD(is,6)=sag30_40DD(is,6)+rate1(branch123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50DD(is,6)=sag40_50DD(is,6)+rate1(branch123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60DD(is,6)=sag50_60DD(is,6)+rate1(branch123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70DD(is,6)=sag60_70DD(is,6)+rate1(branch123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80DD(is,6)=sag70_80DD(is,6)+rate1(branch123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
                sag80_90DD(is,6)=sag80_90DD(is,6)+rate1(branch123,Ftype);
            end
        end
        if loadpha(is,7)==1;
            modulsag = min(abs(Vabc(:,is)*sqrt(3)));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10DD(is,7)=sag0_10DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20DD(is,7)=sag10_20DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30DD(is,7)=sag20_30DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40DD(is,7)=sag30_40DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50DD(is,7)=sag40_50DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60DD(is,7)=sag50_60DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70DD(is,7)=sag60_70DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80DD(is,7)=sag70_80DD(is,7)+rate1(branch123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
                sag80_90DD(is,7)=sag80_90DD(is,7)+rate1(branch123,Ftype);
            end
        end
        if loadpha(is,8)==1;
            modulsag = min(abs(VLLabc(:,is)*sqrt(3)));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10DD(is,8)=sag0_10DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20DD(is,8)=sag10_20DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30DD(is,8)=sag20_30DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40DD(is,8)=sag30_40DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50DD(is,8)=sag40_50DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60DD(is,8)=sag50_60DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70DD(is,8)=sag60_70DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80DD(is,8)=sag70_80DD(is,8)+rate1(branch123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
                sag80_90DD(is,8)=sag80_90DD(is,8)+rate1(branch123,Ftype);
            end
        end
    end
else
```

```
      end

   end % Ket thuc tinh cac loai ngan mach tai 1 nut tai

end % Ket thuc tinh ngan mach tai toan bo cac nut tai tren luoi

% -------------------------------------------------------------------
% TINH TOAN SAG DO SU CO MBAPP

% Nhap so lieu cac dang ngan mach
load faulttype_transfo.txt;
[FauLoNo2 FauTyNo2]= size(faulttype_transfo);
% Fault Location Number & Fault Type Number at each location
Faultcode2 = faulttype_transfo;

% Nhap so lieu ty le su co tai cac TBAPP
load faultrate_transfo73_50.txt;
rate2=faultrate_transfo73_50;

% Kiem tra muc do sag
sag0_10BA =zeros(Sag_Row,Sag_Col);
sag10_20BA=zeros(Sag_Row,Sag_Col);
sag20_30BA=zeros(Sag_Row,Sag_Col);
sag30_40BA=zeros(Sag_Row,Sag_Col);
sag40_50BA=zeros(Sag_Row,Sag_Col);
sag50_60BA=zeros(Sag_Row,Sag_Col);
sag60_70BA=zeros(Sag_Row,Sag_Col);
sag70_80BA=zeros(Sag_Row,Sag_Col);
sag80_90BA=zeros(Sag_Row,Sag_Col);

for node123=1:FauLoNo2
   % Tinh tong tro duong day tu thanh cai ha ap TBATG den diem NM
   SC_bus = node123;
   nut(1) = SC_bus;
   v=1;
   while SC_bus ~=1
      v=v+1;
      for ih=1:Sys_Row;
         for il=1:Sys_Col;
            if System_mat(ih,il)==SC_bus;
               nut(v)=ih;
               SC_bus=ih;
            end
         end
      end
   end
   Zm = zeros(3,3);
   for ik=1:v-1
      Zm(1,1) = Zm(1,1) + Zline(nut(ik)-1,1);
      Zm(1,2) = Zm(1,2) + Zline(nut(ik)-1,2);
      Zm(1,3) = Zm(1,3) + Zline(nut(ik)-1,3);
      Zm(2,1) = Zm(1,2);
      Zm(2,2) = Zm(2,2) + Zline(nut(ik)-1,4);
      Zm(2,3) = Zm(2,3) + Zline(nut(ik)-1,5);
      Zm(3,1) = Zm(1,3);
      Zm(3,2) = Zm(2,3);
      Zm(3,3) = Zm(3,3) + Zline(nut(ik)-1,6);
   end
   % Tong tro Thevenin tu nguon den diem ngan mach
   ZTOT = Zsource + Zm;

   % ---------------------------------------------------
   % 5. Tinh dong dien ngan mach
   for Ftype=1:FauTyNo2
      SC_code=Faultcode2(node123,Ftype);
      if SC_code ~= 0
```

```
    Zf = 0;
    Zfault = [Zf 0 0;0 Zf 0;0 0 Zf];
    ZEQ = ZTOT+Zfault;
    YTOT = inv(ZEQ);
    Ysabc = YTOT(:,1)+YTOT(:,2)+YTOT(:,3);

    IPabc = YTOT*Eth;

    C1 = [U,YTOT,Ysabc];
    C2 = zeros(4,7);

    switch SC_code
        case 1 % Ib = Ic = 0, Ua = Ug = 0
            C2(1,2)=1; C2(2,3)=1; C2(3,4)=1; C2(4,7)=1;
        case 2 % Ia = Ic = 0, Ub = Ug = 0
            C2(1,1)=1; C2(2,3)=1; C2(3,5)=1; C2(4,7)=1;
        case 3 % Ia = Ib = 0, Uc = Ug = 0
            C2(1,1)=1; C2(2,2)=1; C2(3,6)=1; C2(4,7)=1;
        case 4 % Ia+Ib = 0, Ic = 0, Ua = Ub = 0
            C2(1,1)=1; C2(1,2)=1; C2(2,3)=1; C2(3,4)=1; C2(4,5)=1;
        case 5 % Ib+Ic = 0, Ia = 0, Ub = Uc = 0
            C2(1,2)=1; C2(1,3)=1; C2(2,1)=1; C2(3,5)=1; C2(4,6)=1;
        case 6 % Ia+Ic = 0, Ib = 0, Ua = Uc = 0
            C2(1,1)=1; C2(1,3)=1; C2(2,2)=1; C2(3,4)=1; C2(4,6)=1;
        case 7 % Ic = 0, Ua = Ub = Ug = 0
            C2(1,3)=1; C2(2,4)=1; C2(3,5)=1; C2(4,7)=1;
        case 8 % Ia = 0, Ub = Uc = Ug = 0
            C2(1,1)=1; C2(2,5)=1; C2(3,6)=1; C2(4,7)=1;
        case 9 % Ib = 0, Ua = Uc = Ug = 0
            C2(1,2)=1; C2(2,4)=1; C2(3,6)=1; C2(4,7)=1;
        case 10 % Ua = Uc = Uc = Ug = 0
            C2(1,4)=1; C2(2,5)=1; C2(3,6)=1; C2(4,7)=1;
        case 11 % Ia+Ib+Ic=0 Ua = Uc = Uc = 0
            C2(1,1)=1; C2(1,2)=1; C2(1,3)=1; C2(2,4)=1; C2(3,5)=1; C2(4,6)=1;
        otherwise
    end
    C=[C1;C2];

    IP = [IPabc;0;0;0;0];
    X = inv(C)*IP;

    % 6. Tinh dien ap cac nut tren nhanh ngan mach tro ve nguon

    Ifabc(1,1) = X(1,1);
    Ifabc(2,1) = X(2,1);
    Ifabc(3,1) = X(3,1);
    Vfabc(1,1)  = X(4,1) + X(7,1);
    Vfabc(2,1)  = X(5,1) + X(7,1);
    Vfabc(3,1)  = X(6,1) + X(7,1);

    for ij=1:v-1
        Zn(1,1)=Zline(nut(ij)-1,1);
        Zn(1,2)=Zline(nut(ij)-1,2);
        Zn(1,3)=Zline(nut(ij)-1,3);
        Zn(2,1)=Zn(1,2);
        Zn(2,2)=Zline(nut(ij)-1,4);
        Zn(2,3)=Zline(nut(ij)-1,5);
        Zn(3,1)=Zn(1,3);
        Zn(3,2)=Zn(2,3);
        Zn(3,3)=Zline(nut(ij)-1,6);

        An = U;
        Bn = Zn;
        Cn = zeros(3,3);
        Dn = U;
        Vfabc(:,ij+1) = An*Vfabc(:,ij)+Bn*Ifabc(:,ij);
```

```
    Ifabc(:,ij+1) = Cn*Vfabc(:,ij)+Dn*Ifabc(:,ij);

    for pha=1:3
        if Phavol(nut(ij),pha) ~= 1;
            Vfabc(pha,ij) = 0;
        end
    end
end
for ig=1:v
    Vabc(:,nut(ig))=Vfabc(:,ig);
end

% Tinh dien ap cac nut khong nam tren tuyen ngan mach
for ii=1:Sys_Row
    kt=0;
    node = ii;
    for in=1:v
        if node == nut(in)
            kt=1;
        end
    end
    if kt ~= 1
        kt1 = 0;
        % Tim duong noi tu nut ii toi nut gan nhat tren tuyen NM
        while kt1 ~=1
            for ir=1:Sys_Row;
                for ic=1:Sys_Col;
                    if System_mat(ir,ic)==node
                        node=ir;
                    end
                end
            end
            for iv=1:v
                if node == nut(iv)
                    kt1=1;
                    v_node = iv;
                end
            end
        end
        Vabc(:,ii)=Vfabc(:,v_node);
        for pha_i=1:3
            if Phavol(ii,pha_i) ==0
                Vabc(pha_i,ii)=0;
            end
        end
    else
    end
end

% Kiem tra muc do sag

VLLabc=zeros(3,Sys_Row);
for iy=1:Sys_Row
    if Phavol(iy,4)==1
        VLLabc(1,iy)=Vabc(1,iy)-Vabc(2,iy);
    end
    if Phavol(iy,5)==1
        VLLabc(2,iy)=Vabc(2,iy)-Vabc(3,iy);
    end
    if Phavol(iy,6)==1
        VLLabc(3,iy)=Vabc(3,iy)-Vabc(1,iy);
    end
end

for is=1:Sys_Row
    % Kiem tra so pha cua nut
```

```
        if loadpha(is,1)==1;
            modulsag = abs(Vabc(1,is)*sqrt(3));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10BA(is,1)=sag0_10BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20BA(is,1)=sag10_20BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30BA(is,1)=sag20_30BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40BA(is,1)=sag30_40BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50BA(is,1)=sag40_50BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60BA(is,1)=sag50_60BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70BA(is,1)=sag60_70BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80BA(is,1)=sag70_80BA(is,1)+rate2(node123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
                sag80_90BA(is,1)=sag80_90BA(is,1)+rate2(node123,Ftype);
            end
        end
        if loadpha(is,2)==1;
            modulsag = abs(Vabc(2,is)*sqrt(3));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10BA(is,2)=sag0_10BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20BA(is,2)=sag10_20BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30BA(is,2)=sag20_30BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40BA(is,2)=sag30_40BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50BA(is,2)=sag40_50BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60BA(is,2)=sag50_60BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70BA(is,2)=sag60_70BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80BA(is,2)=sag70_80BA(is,2)+rate2(node123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
                sag80_90BA(is,2)=sag80_90BA(is,2)+rate2(node123,Ftype);
            end
        end
        if loadpha(is,3)==1;
            modulsag = abs(Vabc(3,is)*sqrt(3));
            % Sag cumulation
            if modulsag <= 0.1
                sag0_10BA(is,3)=sag0_10BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.2 & modulsag >0.1
                sag10_20BA(is,3)=sag10_20BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.3 & modulsag >0.2
                sag20_30BA(is,3)=sag20_30BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.4 & modulsag >0.3
                sag30_40BA(is,3)=sag30_40BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.5 & modulsag >0.4
                sag40_50BA(is,3)=sag40_50BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.6 & modulsag >0.5
                sag50_60BA(is,3)=sag50_60BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.7 & modulsag >0.6
                sag60_70BA(is,3)=sag60_70BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.8 & modulsag >0.7
                sag70_80BA(is,3)=sag70_80BA(is,3)+rate2(node123,Ftype);
            elseif modulsag <=0.9 & modulsag >0.8
```

```
        sag80_90BA(is,3)=sag80_90BA(is,3)+rate2(node123,Ftype);
    end
end
if loadpha(is,4)==1;
    modulsag = abs(VLLabc(1,is));
    % Sag cumulation
    if modulsag <= 0.1
        sag0_10BA(is,4)=sag0_10BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.2 & modulsag >0.1
        sag10_20BA(is,4)=sag10_20BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.3 & modulsag >0.2
        sag20_30BA(is,4)=sag20_30BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.4 & modulsag >0.3
        sag30_40BA(is,4)=sag30_40BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.5 & modulsag >0.4
        sag40_50BA(is,4)=sag40_50BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.6 & modulsag >0.5
        sag50_60BA(is,4)=sag50_60BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.7 & modulsag >0.6
        sag60_70BA(is,4)=sag60_70BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.8 & modulsag >0.7
        sag70_80BA(is,4)=sag70_80BA(is,4)+rate2(node123,Ftype);
    elseif modulsag <=0.9 & modulsag >0.8
        sag80_90BA(is,4)=sag80_90BA(is,4)+rate2(node123,Ftype);
    end
end
if loadpha(is,5)==1;
    modulsag = abs(VLLabc(2,is));
    % Sag cumulation
    if modulsag <= 0.1
        sag0_10BA(is,5)=sag0_10BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.2 & modulsag >0.1
        sag10_20BA(is,5)=sag10_20BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.3 & modulsag >0.2
        sag20_30BA(is,5)=sag20_30BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.4 & modulsag >0.3
        sag30_40BA(is,5)=sag30_40BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.5 & modulsag >0.4
        sag40_50BA(is,5)=sag40_50BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.6 & modulsag >0.5
        sag50_60BA(is,5)=sag50_60BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.7 & modulsag >0.6
        sag60_70BA(is,5)=sag60_70BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.8 & modulsag >0.7
        sag70_80BA(is,5)=sag70_80BA(is,5)+rate2(node123,Ftype);
    elseif modulsag <=0.9 & modulsag >0.8
        sag80_90BA(is,5)=sag80_90BA(is,5)+rate2(node123,Ftype);
    end
end
if loadpha(is,6)==1;
    modulsag = abs(VLLabc(3,is));
    % Sag cumulation
    if modulsag <= 0.1
        sag0_10BA(is,6)=sag0_10BA(is,6)+rate2(node123,Ftype);
    elseif modulsag <=0.2 & modulsag >0.1
        sag10_20BA(is,6)=sag10_20BA(is,6)+rate2(node123,Ftype);
    elseif modulsag <=0.3 & modulsag >0.2
        sag20_30BA(is,6)=sag20_30BA(is,6)+rate2(node123,Ftype);
    elseif modulsag <=0.4 & modulsag >0.3
        sag30_40BA(is,6)=sag30_40BA(is,6)+rate2(node123,Ftype);
    elseif modulsag <=0.5 & modulsag >0.4
        sag40_50BA(is,6)=sag40_50BA(is,6)+rate2(node123,Ftype);
    elseif modulsag <=0.6 & modulsag >0.5
        sag50_60BA(is,6)=sag50_60BA(is,6)+rate2(node123,Ftype);
    elseif modulsag <=0.7 & modulsag >0.6
        sag60_70BA(is,6)=sag60_70BA(is,6)+rate2(node123,Ftype);
```

```
                    elseif modulsag <=0.8 & modulsag >0.7
                        sag70_80BA(is,6)=sag70_80BA(is,6)+rate2(node123,Ftype);
                    elseif modulsag <=0.9 & modulsag >0.8
                        sag80_90BA(is,6)=sag80_90BA(is,6)+rate2(node123,Ftype);
                    end
                end
                if loadpha(is,7)==1;
                    modulsag = min(abs(Vabc(:,is)*sqrt(3)));
                    % Sag cumulation
                    if modulsag <= 0.1
                        sag0_10BA(is,7)=sag0_10BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.2 & modulsag >0.1
                        sag10_20BA(is,7)=sag10_20BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.3 & modulsag >0.2
                        sag20_30BA(is,7)=sag20_30BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.4 & modulsag >0.3
                        sag30_40BA(is,7)=sag30_40BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.5 & modulsag >0.4
                        sag40_50BA(is,7)=sag40_50BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.6 & modulsag >0.5
                        sag50_60BA(is,7)=sag50_60BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.7 & modulsag >0.6
                        sag60_70BA(is,7)=sag60_70BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.8 & modulsag >0.7
                        sag70_80BA(is,7)=sag70_80BA(is,7)+rate2(node123,Ftype);
                    elseif modulsag <=0.9 & modulsag >0.8
                        sag80_90BA(is,7)=sag80_90BA(is,7)+rate2(node123,Ftype);
                    end
                end
                if loadpha(is,8)==1;
                    modulsag = min(abs(VLLabc(:,is)*sqrt(3)));
                    % Sag cumulation
                    if modulsag <= 0.1
                        sag0_10BA(is,8)=sag0_10BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.2 & modulsag >0.1
                        sag10_20BA(is,8)=sag10_20BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.3 & modulsag >0.2
                        sag20_30BA(is,8)=sag20_30BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.4 & modulsag >0.3
                        sag30_40BA(is,8)=sag30_40BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.5 & modulsag >0.4
                        sag40_50BA(is,8)=sag40_50BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.6 & modulsag >0.5
                        sag50_60BA(is,8)=sag50_60BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.7 & modulsag >0.6
                        sag60_70BA(is,8)=sag60_70BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.8 & modulsag >0.7
                        sag70_80BA(is,8)=sag70_80BA(is,8)+rate2(node123,Ftype);
                    elseif modulsag <=0.9 & modulsag >0.8
                        sag80_90BA(is,8)=sag80_90BA(is,8)+rate2(node123,Ftype);
                    end
                end
            end

        else
        end

    end % Ket thuc tinh cac loai ngan mach tai 1 nut tai

end % Ket thuc tinh ngan mach tai toan bo cac nut tai tren luoi

disp('Tinh xong. An phim bat ky de xem ket qua !');
pause
clc
sag0_10 = sag0_10DD + sag0_10BA;
sag10_20 = sag10_20DD + sag10_20BA;
```

```
sag20_30 = sag20_30DD + sag20_30BA;
sag30_40 = sag30_40DD + sag30_40BA;
sag40_50 = sag40_50DD + sag40_50BA;
sag50_60 = sag50_60DD + sag50_60BA;
sag60_70 = sag60_70DD + sag60_70BA;
sag70_80 = sag70_80DD + sag70_80BA;
sag80_90 = sag80_90DD + sag80_90BA;


fprintf(' I. IN KET QUA \n');
fprintf(' 1. TONG SO SAG THEO MUC DO SAG  \n');
fprintf('  Nut   0-10    10-20    20-30    30-40    40-50    50-60    60-70    70-80    80-90 ');
fprintf(' \n ');

sag0_10M =sum(sag0_10');
sag10_20M=sum(sag10_20');
sag20_30M=sum(sag20_30');
sag30_40M=sum(sag30_40');
sag40_50M=sum(sag40_50');
sag50_60M=sum(sag50_60');
sag60_70M=sum(sag60_70');
sag70_80M=sum(sag70_80');
sag80_90M=sum(sag80_90');

for iy=1:Sag_Row
   fprintf(' %3.0f',iy);
   fprintf('  %5.1f  ',sag0_10M(1,iy));
   fprintf('  %5.1f  ',sag10_20M(1,iy));
   fprintf('  %5.1f  ',sag20_30M(1,iy));
   fprintf('  %5.1f  ',sag30_40M(1,iy));
   fprintf('  %5.1f  ',sag40_50M(1,iy));
   fprintf('  %5.1f  ',sag50_60M(1,iy));
   fprintf('  %5.1f  ',sag60_70M(1,iy));
   fprintf('  %5.1f  ',sag70_80M(1,iy));
   fprintf('  %5.1f  ',sag80_90M(1,iy));
   fprintf(' \n ');
end
fprintf(' Sum ');
fprintf('%5.1f   ',sum(sag0_10M'));
fprintf('%5.1f   ',sum(sag10_20M'));
fprintf('%5.1f   ',sum(sag20_30M'));
fprintf('%5.1f   ',sum(sag30_40M'));
fprintf('%5.1f   ',sum(sag40_50M'));
fprintf('%5.1f   ',sum(sag50_60M'));
fprintf('%5.1f   ',sum(sag60_70M'));
fprintf('%5.1f   ',sum(sag70_80M'));
fprintf('%5.1f   \n',sum(sag80_90M'));

fprintf(' \n');
fprintf(' 2. TONG SO SAG TICH LUU THEO MUC DO SAG \n');
fprintf('  Nut    <10      <20      <30      <40      <50      <60      <70      <80      <90 ');
fprintf(' \n');

sag_cumulative10 = sag0_10M;
sag_cumulative20 = sag0_10M+sag10_20M;
sag_cumulative30 = sag0_10M+sag10_20M+sag20_30M;
sag_cumulative40 = sag0_10M+sag10_20M+sag20_30M+sag30_40M;
sag_cumulative50 = sag0_10M+sag10_20M+sag20_30M+sag30_40M+sag40_50M;
sag_cumulative60 = sag0_10M+sag10_20M+sag20_30M+sag30_40M+sag40_50M+sag50_60M;
sag_cumulative70 = sag0_10M+sag10_20M+sag20_30M+sag30_40M+sag40_50M+sag50_60M+sag60_70M;
sag_cumulative80 = sag0_10M+sag10_20M+sag20_30M+sag30_40M+sag40_50M+sag50_60M+sag60_70M+sag70_80M;
sag_cumulative90 = sag0_10M+sag10_20M+sag20_30M+sag30_40M+sag40_50M+sag50_60M+sag60_70M+sag70_80M+sag80_90M;

for iy=1:Sys_Row
```

```
    fprintf(' %3.0f ',iy);
    fprintf(' %6.1f  ',sag_cumulative10(1,iy));
    fprintf(' %6.1f  ',sag_cumulative20(1,iy));
    fprintf(' %6.1f  ',sag_cumulative30(1,iy));
    fprintf(' %6.1f  ',sag_cumulative40(1,iy));
    fprintf(' %6.1f  ',sag_cumulative50(1,iy));
    fprintf(' %6.1f  ',sag_cumulative60(1,iy));
    fprintf(' %6.1f  ',sag_cumulative70(1,iy));
    fprintf(' %6.1f  ',sag_cumulative80(1,iy));
    fprintf(' %6.1f  ',sag_cumulative90(1,iy));
    fprintf(' \n');
end
sum_sag_cumulative10 = sum(sag_cumulative10');
sum_sag_cumulative20 = sum(sag_cumulative20');
sum_sag_cumulative30 = sum(sag_cumulative30');
sum_sag_cumulative40 = sum(sag_cumulative40');
sum_sag_cumulative50 = sum(sag_cumulative50');
sum_sag_cumulative60 = sum(sag_cumulative60');
sum_sag_cumulative70 = sum(sag_cumulative70');
sum_sag_cumulative80 = sum(sag_cumulative80');
sum_sag_cumulative90 = sum(sag_cumulative90');

fprintf('------------------\n');
fprintf('Tong');
fprintf(' %7.1f ',sum_sag_cumulative10);
fprintf(' %7.1f ',sum_sag_cumulative20);
fprintf(' %7.1f ',sum_sag_cumulative30);
fprintf(' %7.1f ',sum_sag_cumulative40);
fprintf(' %7.1f ',sum_sag_cumulative50);
fprintf(' %7.1f ',sum_sag_cumulative60);
fprintf(' %7.1f ',sum_sag_cumulative70);
fprintf(' %7.1f ',sum_sag_cumulative80);
fprintf(' %7.1f ',sum_sag_cumulative90);
fprintf(' \n ');

pause
fprintf(' 3. SO LUONG SAG THEO PHU TAI \n');
fprintf(' Nut    Pha-A    Pha-B    Pha-C      A-B      B-C      C-A      3PY      3PD   ');
fprintf(' \n ');

sagA  =[sag0_10(:,1) sag10_20(:,1) sag20_30(:,1) sag30_40(:,1) sag40_50(:,1) sag50_60(:,1) sag60_70(:,1) sag70_80(:,1) sag80_90(:,1)];
sagB  =[sag0_10(:,2) sag10_20(:,2) sag20_30(:,2) sag30_40(:,2) sag40_50(:,2) sag50_60(:,2) sag60_70(:,2) sag70_80(:,2) sag80_90(:,2)];
sagC  =[sag0_10(:,3) sag10_20(:,3) sag20_30(:,3) sag30_40(:,3) sag40_50(:,3) sag50_60(:,3) sag60_70(:,3) sag70_80(:,3) sag80_90(:,3)];
sagAB =[sag0_10(:,4) sag10_20(:,4) sag20_30(:,4) sag30_40(:,4) sag40_50(:,4) sag50_60(:,4) sag60_70(:,4) sag70_80(:,4) sag80_90(:,4)];
sagBC =[sag0_10(:,5) sag10_20(:,5) sag20_30(:,5) sag30_40(:,5) sag40_50(:,5) sag50_60(:,5) sag60_70(:,5) sag70_80(:,5) sag80_90(:,5)];
sagCA =[sag0_10(:,6) sag10_20(:,6) sag20_30(:,6) sag30_40(:,6) sag40_50(:,6) sag50_60(:,6) sag60_70(:,6) sag70_80(:,6) sag80_90(:,6)];
sag3PY=[sag0_10(:,7) sag10_20(:,7) sag20_30(:,7) sag30_40(:,7) sag40_50(:,7) sag50_60(:,7) sag60_70(:,7) sag70_80(:,7) sag80_90(:,7)];
sag3PD=[sag0_10(:,8) sag10_20(:,8) sag20_30(:,8) sag30_40(:,8) sag40_50(:,8) sag50_60(:,8) sag60_70(:,8) sag70_80(:,8) sag80_90(:,8)];

sum_sagA  =sum(sagA');
sum_sagB  =sum(sagB');
sum_sagC  =sum(sagC');
sum_sagAB =sum(sagAB');
sum_sagBC =sum(sagBC');
sum_sagCA =sum(sagCA');
sum_sag3PY=sum(sag3PY');
sum_sag3PD=sum(sag3PD');

for iy=1:Sag_Row
```

```
    fprintf(' %3.0f ',iy);
    fprintf('  %5.1f  ',sum_sagA(1,iy));
    fprintf('  %5.1f  ',sum_sagB(1,iy));
    fprintf('  %5.1f  ',sum_sagC(1,iy));
    fprintf('  %5.1f  ',sum_sagAB(1,iy));
    fprintf('  %5.1f  ',sum_sagBC(1,iy));
    fprintf('  %5.1f  ',sum_sagCA(1,iy));
    fprintf('  %5.1f  ',sum_sag3PY(1,iy));
    fprintf('  %5.1f  ',sum_sag3PD(1,iy));
    fprintf('  \n ');
end
fprintf(' Sum ');
fprintf(' %6.1f  ',sum(sum_sagA'));
fprintf(' %6.1f  ',sum(sum_sagB'));
fprintf(' %6.1f  ',sum(sum_sagC'));
fprintf(' %6.1f  ',sum(sum_sagAB'));
fprintf(' %6.1f  ',sum(sum_sagBC'));
fprintf(' %6.1f  ',sum(sum_sagCA'));
fprintf(' %6.1f  ',sum(sum_sag3PY'));
fprintf(' %6.1f  \n',sum(sum_sag3PD'));

pause
clc
fprintf(' 4. SO LUONG SAG THEO MUC DO SAG TREN TUNG PHA PHU TAI \n');
fprintf('  Nut    <10      <20      <30      <40      <50      <60      <70      <80      <90 ');
fprintf(' \n');

fprintf(' VD : Phu tai pha A \n');
pause
for iy=1:Sag_Row
    fprintf(' %3.0f ',iy);
    fprintf('  %5.1f  ',sagA(iy,1));
    fprintf('  %5.1f  ',sagA(iy,2));
    fprintf('  %5.1f  ',sagA(iy,3));
    fprintf('  %5.1f  ',sagA(iy,4));
    fprintf('  %5.1f  ',sagA(iy,5));
    fprintf('  %5.1f  ',sagA(iy,6));
    fprintf('  %5.1f  ',sagA(iy,7));
    fprintf('  %5.1f  ',sagA(iy,8));
    fprintf('  %5.1f  ',sagA(iy,9));
    fprintf('  \n ');
end
pause
fprintf(' Phu tai pha B \n');
for iy=1:Sag_Row
    fprintf(' %3.0f ',iy);
    fprintf('  %5.1f  ',sagB(iy,1));
    fprintf('  %5.1f  ',sagB(iy,2));
    fprintf('  %5.1f  ',sagB(iy,3));
    fprintf('  %5.1f  ',sagB(iy,4));
    fprintf('  %5.1f  ',sagB(iy,5));
    fprintf('  %5.1f  ',sagB(iy,6));
    fprintf('  %5.1f  ',sagB(iy,7));
    fprintf('  %5.1f  ',sagB(iy,8));
    fprintf('  %5.1f  ',sagB(iy,9));
    fprintf('  \n ');
end
pause
fprintf(' Phu tai pha C \n');
for iy=1:Sag_Row
    fprintf(' %3.0f ',iy);
    fprintf('  %5.1f  ',sagC(iy,1));
    fprintf('  %5.1f  ',sagC(iy,2));
    fprintf('  %5.1f  ',sagC(iy,3));
    fprintf('  %5.1f  ',sagC(iy,4));
    fprintf('  %5.1f  ',sagC(iy,5));
```

```
    fprintf('  %5.1f  ',sagC(iy,6));
    fprintf('  %5.1f  ',sagC(iy,7));
    fprintf('  %5.1f  ',sagC(iy,8));
    fprintf('  %5.1f  ',sagC(iy,9));
    fprintf('  \n ');
end

pause
clc
% Chia 3 so sag cua cac nut phu tai co o tren ca 3 pha (34,70,103)
sag0_10M(1,34)=sag0_10M(1,34)/3
sag10_20M(1,34)=sag10_20M(1,34)/3
sag20_30M(1,34)=sag20_30M(1,34)/3
sag30_40M(1,34)=sag30_40M(1,34)/3
sag40_50M(1,34)=sag40_50M(1,34)/3
sag50_60M(1,34)=sag50_60M(1,34)/3
sag60_70M(1,34)=sag60_70M(1,34)/3
sag70_80M(1,34)=sag70_80M(1,34)/3
sag80_90M(1,34)=sag80_90M(1,34)/3

sag0_10M(1,70)=sag0_10M(1,70)/3
sag10_20M(1,70)=sag10_20M(1,70)/3
sag20_30M(1,70)=sag20_30M(1,70)/3
sag30_40M(1,70)=sag30_40M(1,70)/3
sag40_50M(1,70)=sag40_50M(1,70)/3
sag50_60M(1,70)=sag50_60M(1,70)/3
sag60_70M(1,70)=sag60_70M(1,70)/3
sag70_80M(1,70)=sag70_80M(1,70)/3
sag80_90M(1,70)=sag80_90M(1,70)/3

sag0_10M(1,103)=sag0_10M(1,103)/3
sag10_20M(1,103)=sag10_20M(1,103)/3
sag20_30M(1,103)=sag20_30M(1,103)/3
sag30_40M(1,103)=sag30_40M(1,103)/3
sag40_50M(1,103)=sag40_50M(1,103)/3
sag50_60M(1,103)=sag50_60M(1,103)/3
sag60_70M(1,103)=sag60_70M(1,103)/3
sag70_80M(1,103)=sag70_80M(1,103)/3
sag80_90M(1,103)=sag80_90M(1,103)/3

Ch=100;
while Ch ~= 0;

    % Ve do thi so luong sag tai cac nut
    for ki=Sys_Row:-1:2;
        sendbus(ki-1)=ki;
        Xnode_se(1,ki-1)=Xnode(ki);
        Ynode_se(1,ki-1)=Ynode(ki);
        for kj=1:Sys_Row;
            for kk=1:3;
                if System_mat(kj,kk)==ki;
                    receivbus(ki-1)= kj;
                    Xnode_re(1,ki-1)=Xnode(kj);
                    Ynode_re(1,ki-1)=Ynode(kj);
                end
            end
        end
    end

    Xbranch=[Xnode_se;Xnode_re];
    Ybranch=[Ynode_se;Ynode_re];
    Zbranch=zeros(2,branch_no);

    disp('II. BIEU DIEN DO THI TAN XUAT SAG')
    disp('Chon mot trong cac loai sag duoi day');
    disp('- Bo qua      : 0');
```

```
disp('- Sag 0-0.1   : 1');
disp('- Sag 0.1-0.2 : 2');
disp('- Sag 0.2-0.3 : 3');
disp('- Sag 0.3-0.4 : 4');
disp('- Sag 0.4-0.5 : 5');
disp('- Sag 0.5-0.6 : 6');
disp('- Sag 0.6-0.7 : 7');
disp('- Sag 0.7-0.8 : 8');
disp('- Sag 0.8-0.9 : 9');
disp('- Tong so sag 0-0.9 : 10');
disp(' ');
Ch=input('Cho loai sag (0-10) :');
hold off
for iv=1:branch_no
   Xv=Xbranch(:,iv);
   Yv=Ybranch(:,iv);
   Zv=Zbranch(:,iv);
   Kiemtrapha = Phavol(iv+1,[1 2 3])';
   if isequal(Kiemtrapha,[1;0;0]) ==1
      plot3(Xv,Yv,Zv,':r');
   elseif isequal(Kiemtrapha,[0;1;0]) ==1
      plot3(Xv,Yv,Zv,':r');
   elseif isequal(Kiemtrapha,[0;0;1]) ==1
      plot3(Xv,Yv,Zv,':r');
   elseif isequal(Kiemtrapha,[1;1;0]) ==1
      plot3(Xv,Yv,Zv,'--r');
   elseif isequal(Kiemtrapha,[1;0;1]) ==1
      plot3(Xv,Yv,Zv,'--r');
   elseif isequal(Kiemtrapha,[0;1;1]) ==1
      plot3(Xv,Yv,Zv,'--r');
   elseif isequal(Kiemtrapha,[1;1;1]) ==1
      plot3(Xv,Yv,Zv,'r');
   end
   hold on
end

axis([0 5000 0 3500 0 150]);

grid on

view(-25,60);

% Ve so luong sag tai cac nut

if Ch==1
   for iy=1:Sys_Row
      if sag0_10M(1,iy) ~=0
         stem3(Xnode(iy), Ynode(iy), sag0_10M(1,iy),'fill');
         hold on
      end
   end
elseif Ch==2
   for iy=1:Sys_Row
      if sag10_20M(1,iy) ~=0
         stem3(Xnode(iy), Ynode(iy), sag10_20M(1,iy),'fill');
         hold on
      end
   end
elseif Ch==3
   for iy=1:Sys_Row
      if sag20_30M(1,iy) ~=0
         stem3(Xnode(iy), Ynode(iy), sag20_30M(1,iy),'fill');
         hold on
      end
   end
elseif Ch==4
```

```
        for iy=1:Sys_Row
            if sag30_40M(1,iy) ~=0
                stem3(Xnode(iy), Ynode(iy), sag30_40M(1,iy),'fill');
                hold on
            end
        end
    elseif Ch==5
        for iy=1:Sys_Row
            if sag40_50M(1,iy) ~=0
                stem3(Xnode(iy), Ynode(iy), sag40_50M(1,iy),'fill');
                hold on
            end
        end
    elseif Ch==6
        for iy=1:Sys_Row
            if sag50_60M(1,iy) ~=0
                stem3(Xnode(iy), Ynode(iy), sag50_60M(1,iy),'fill');
                hold on
            end
        end
    elseif Ch==7
        for iy=1:Sys_Row
            if sag60_70M(1,iy) ~=0
                stem3(Xnode(iy), Ynode(iy), sag60_70M(1,iy),'fill');
                %axis([0 5000 0 3500 0 200]);
                hold on
            end
        end
    elseif Ch==8
        for iy=1:Sys_Row
            if sag70_80M(1,iy) ~=0
                stem3(Xnode(iy), Ynode(iy), sag70_80M(1,iy),'fill');
                %axis([0 5000 0 3500 0 200]);
                hold on
            end
        end
    elseif Ch==9
        for iy=1:Sys_Row
            if sag80_90M(1,iy) ~=0
                stem3(Xnode(iy), Ynode(iy), sag80_90M(1,iy),'fill');
                hold on
            end
        end
    elseif Ch==10
        for iy=1:Sys_Row
            if sag_cumulative90(1,iy) ~=0
                stem3(Xnode(iy), Ynode(iy), sag_cumulative90(1,iy),'fill');
                axis([0 5000 0 3500 0 800]);
                hold on
            end
        end
    end % Ket thuc ve so luong sag tai cac nut
    clc
    hold off

end

disp(' ')
disp(' KET THUC')
```

A.5. Data files

Details of data file is given on the Table C.1, Appendix C. In running program on Matlab, data files are made into Notepad files including

- phase123abc.txt : The number of line phase at nodes. This data file indicates how many line phase is available at all nodes of the system. The file is in form of matrix 6 column x 123 rows. 6 Column implies phas A, B, C, AB, BC, CA. '1' appears on column 2, row i (i=1,123) means phase B is available at the node i. '0' has opposite meaning. This file is to control the calculation of phase-to-neutral and phase-to-phase voltage at all nodes (no matter this node is connected to distribution transformer or not)
- load123abc.txt : The load locations. This file marks the location of different phase load. It comes in the matrix of 8x123. 8 column means the phase load : Phase A, B, C, AB, BC, CA, 3phase wye connected, 3phase delta connected.
- linereal123.txt and lineimag123.txt : line parameters. These two files provide real part and imaginary part of line impedance matrix.
- testfeeder123sys.txt : Network topology. This file provides marking of all connections of the network. Because one node will connect to maximum three other nodes to the load direction, the data file is in the matrix from of 3 column and 122 row. An illustrating example is given on the table a1 as follows

Table A.1. The format of data file testfeeder123sys.txt

Sending node (*)	Receiving node (**)		
1	2	0	0
2	3	7	8
3	4	5	0
4	0	0	0

(*) This node does not appear on the data file. (**) This node appear on the data file.

The first row implies that from node 1 (bulk supply point), there's only a branch connected to node 2.

The second row indicates that from node 2, there're two branches connected to node 7 and node 8.

The third row means that from node 3, there're two branches connected to node 4 and node 5.

The forth round shows that the node 4 is the end of a branch without further connection.

- faulttype_line.txt : This data file provides the selection of fault type for the fault location of all segments of line. This selection depends on the number of available phase line. This program considers ten different fault types with corresponding fault codes as follows
  - 1 : phase A to ground
  - 2 : phase B to ground
  - 3 : phase C to ground
  - 4 : phase A to phase B
  - 5 : phase B to phase C
  - 6 : phase C to phase A
  - 7 : phase A and B to ground
  - 8 : phase B and C to ground
  - 9 : phase C and A to ground
  - 10 : three phase to ground

- faultrate_line73_50.txt : This data file provides the fault rate of above listed fault types for each line fault position. This data can be calculated using fault distribution modeling as stated in the part IV. 73-50 implies the mean value is at node 73 and the $\alpha_{eq} = \alpha_{ex} = 50\%$. For other case study of fault distribution with different values of distribution parameters, this data file must be recalculated.
- faulttype_transfo.txt : This data file provides the selection of fault type for the fault location of all distribution transformers. This selection depends on the number of available phase lines that connects with distribution transformers. Similarly, this program also considers ten different fault types with corresponding fault codes.
- faultrate_transfo73_50.txt : Similar to faultrate_line73_50.txt, this data file gives the fault rate of above listed fault types for each transformer fault position.
- load_coordinate.txt : This file is the input of the co-ordination (in ft) of fault positions. It's identical to the node position (Figure 8).

APPENDIX B

IEEE 123 BUS RADIAL DISTRIBUTION TESTFEEDER

B.1. Introduction

IEEE 123 bus radial distribution testfeeder is proposed by IEEE Distribution Planning Working Group in 1991 as an object with full data for many computer programs for the analysis of radial distribution feeders to verify its correctness. The research also uses this network to check the program made by author that gives confident results. It's also used for sag analysis. However, the research renumbers the whole scheme for facilitating to make data file and main module programing. The renumbering is demonstrated as the following Figure A.2.

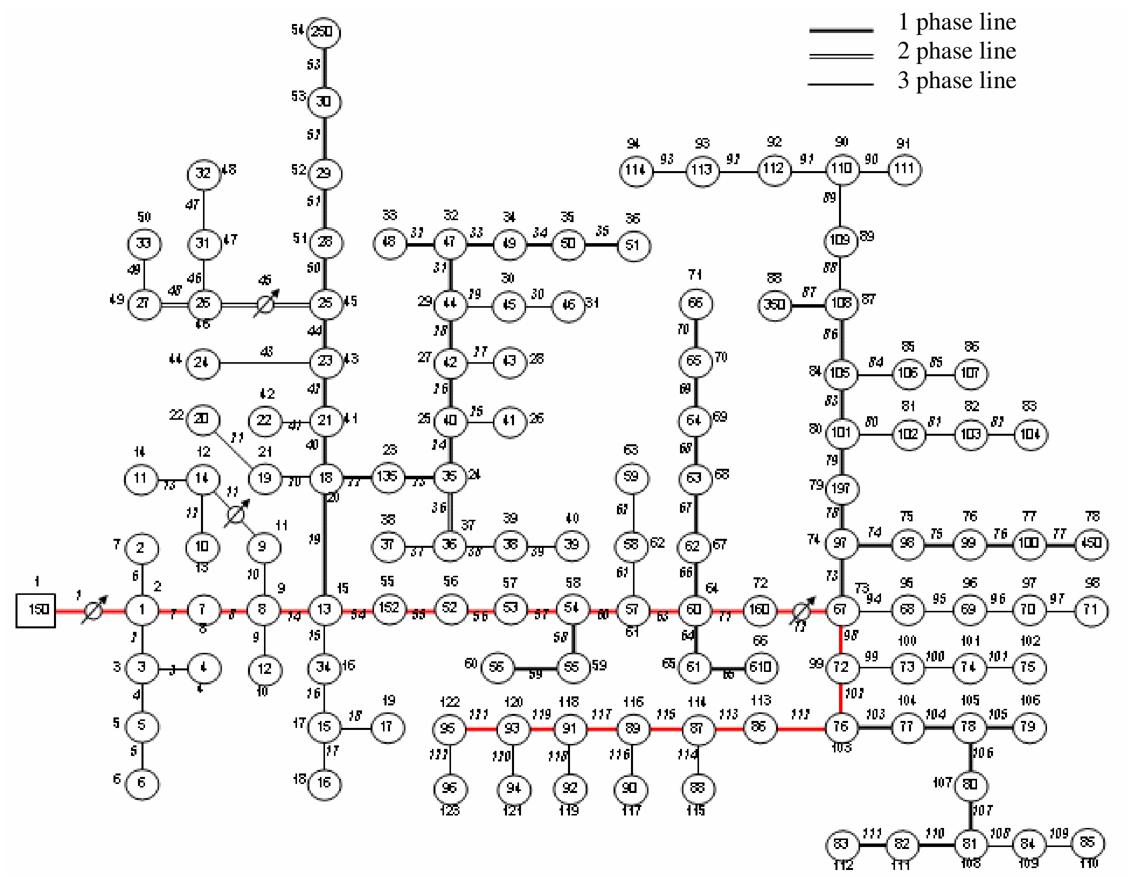

Figure A.2. Numbering of IEEE 123 bus radial testfeeder

B.2. Numbering principle

The actual numbering [12] is given as the numbers inside the nodal circle. It's not really a consecutive series of integer number. For example, some big numbers are assigned to some nodes like 150, 152, 160, 250, 450, 610. It does not facilitate the programing. Therefore, an new numbering is made with number marked just outside the nodal circle for node numbering and along the segment of line. There're 122 segments of line in total. The principle of numbering the line is that the number of the segment of line equals to number of the node (from which it goes out) minus 1. For example, for the direction to the bulk point, node 55 connects to node 15 . Therefore, the segment "55-15" is numbered as 54 (55-1).

APPENDIX C

DATA FILES

Table A.2. Fault position (co-ordinate) and applicable fault types for distribution transformers

Table A.3. Fault position (co-ordinate) and applicable fault types for lines

Table A.4. Scheme renumbering and line parameters

Table A.2. Fault position (co-ordinate) and applicable fault types for distribution transformers

Node	Co-ordinate in ft (load_coordinate.txt)		Load phase-type (load123abc.txt)								Nodal phase-type (phase123abc.txt)						Fault code (faulttype_transfo.txt)									
	x_i	y_i	A	B	C	AB	BC	CA	3PY	3PD	A	B	C	AB	BC	CA	AN	BN	CN	AB	BC	CA	ABN	BCN	CAN	3N
1	806	417	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
2	1195	583	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
3	1250	347	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1333	83	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
5	1556	389	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
6	1806	417	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
7	1139	778	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
8	1445	667	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
9	1625	722	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
10	1486	556	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
11	1556	889	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
12	1195	1083	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	1333	806	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
14	917	972	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
15	1945	833	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
16	2014	667	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
17	2084	556	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
18	2264	222	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
19	2389	722	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
20	1556	1514	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
21	1389	1417	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
22	945	1278	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
23	1556	1514	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
24	1806	1583	0	0	0	1	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
25	1708	1833	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
26	2111	1972	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
27	1611	2056	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
28	2250	2334	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
29	1528	2278	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
30	1750	2334	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
31	2056	2472	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
32	1445	2556	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	2	3	4	5	6	7	8	9	10
33	1278	2528	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	2	3	4	5	6	7	8	9	10
34	1667	2611	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	2	3	4	5	6	7	8	9	10
35	1917	2695	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	3	0	0	0	0	0	0	0
36	2139	2722	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
37	2361	1472	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
38	1861	1306	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
39	2528	1528	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
40	2847	1695	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
41	1417	1750	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
42	833	1528	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
43	1278	2000	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
44	722	1778	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
45	1181	2167	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
46	833	2042	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
47	750	2278	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
48	611	2639	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
49	542	1945	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
50	306	2528	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
51	1111	2306	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
52	1028	2556	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
53	945	2750	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	3	0	0	0	0	0	0	0
54	861	2917	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
55	1945	833	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
56	2222	917	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
57	2417	945	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
58	2500	972	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
59	2778	1056	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
60	2972	1139	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0

Node	Co-ordinate in ft (load_coordinate.txt)		Load phase-type (load123abc.txt)								Nodal phase-type (phase123abc.txt)						Fault code (faulttype_transfo.txt)									
	x_i	y_i	A	B	C	AB	BC	CA	3PY	3PD	A	B	C	AB	BC	CA	AN	BN	CN	AB	BC	CA	ABN	BCN	CAN	3N
61	2417	1306	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
62	2222	1222	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
63	2000	1139	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
64	3084	1500	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
65	3278	1000	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
66	3278	1000	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
67	2986	1750	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	3	0	0	0	0	0	0	0
68	2959	2000	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
69	2806	2306	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0
70	2556	2334	0	0	0	1	1	1	0	0	1	1	1	1	1	1	1	2	3	4	5	6	7	8	9	10
71	2417	2250	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	3	0	0	0	0	0	0	0
72	3084	1500	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
73	3445	1667	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
74	3264	1917	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
75	3500	2028	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
76	3889	2250	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0
77	4223	2472	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	3	0	0	0	0	0	0	0
78	4556	2667	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
79	3264	1917	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
80	3167	2111	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
81	3334	2222	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
82	3611	2417	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
83	4028	2695	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
84	3000	2389	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
85	3236	2500	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
86	3667	2778	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
87	2889	2556	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
88	2667	3028	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
89	3278	2778	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
90	3500	2972	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
91	2834	2834	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
92	3500	2972	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
93	3945	3056	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
94	4389	3139	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
95	3653	1806	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
96	3861	1945	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
97	4056	2056	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
98	4264	2195	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
99	3556	1472	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
100	3723	1583	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
101	3973	1861	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
102	4306	1945	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
103	3639	1278	0	0	0	1	1	1	0	0	1	1	1	1	1	1	1	2	3	4	5	6	7	8	9	10
104	3834	1417	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0
105	3986	1528	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
106	4139	1639	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
107	4028	1083	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0
108	4084	889	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
109	4403	1236	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
110	4361	1611	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
111	4153	611	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
112	4195	389	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	3	0	0	0	0	0	0	0
113	3750	542	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0
114	3445	486	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0
115	3306	806	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
116	3222	444	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
117	3084	750	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0
118	3014	417	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
119	2875	695	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0
120	2834	361	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
121	2625	722	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
122	2584	250	0	1	0	0	0	0	0	0	1	1	1	1	1	1	0	2	0	0	0	0	0	0	0	0
123	2417	528	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0

Table A.3. Fault position (co-ordinate) and applicable fault types for lines

Branch	Length in ft	Co-ordinate in ft (load_coordinate.txt)		Line Phase No			Fault code (faulttype_line.txt)									
		x_i	y_i	A	B	C	AN	BN	CN	AB	BC	CA	ABN	BCN	CAN	3N
1	400	1195	583	1	1	1	1	2	3	4	5	6	7	8	9	10
2	250	1250	347	0	0	1	0	0	3	0	0	0	0	0	0	0
3	200	1333	83	0	0	1	0	0	3	0	0	0	0	0	0	0
4	325	1556	389	0	0	1	0	0	3	0	0	0	0	0	0	0
5	250	1806	417	0	0	1	0	0	3	0	0	0	0	0	0	0
6	175	1139	778	0	1	0	0	2	0	0	0	0	0	0	0	0
7	300	1445	667	1	1	1	1	2	3	4	5	6	7	8	9	10
8	200	1625	722	1	1	1	1	2	3	4	5	6	7	8	9	10
9	225	1486	556	0	1	0	0	2	0	0	0	0	0	0	0	0
10	225	1556	889	1	0	0	1	0	0	0	0	0	0	0	0	0
11	425	1195	1083	1	0	0	1	0	0	0	0	0	0	0	0	0
12	250	1333	806	1	0	0	1	0	0	0	0	0	0	0	0	0
13	250	917	972	1	0	0	1	0	0	0	0	0	0	0	0	0
14	300	1945	833	1	1	1	1	2	3	4	5	6	7	8	9	10
15	150	2014	667	0	0	1	0	0	3	0	0	0	0	0	0	0
16	100	2084	556	0	0	1	0	0	3	0	0	0	0	0	0	0
17	350	2264	222	0	0	1	0	0	3	0	0	0	0	0	0	0
18	375	2389	722	0	0	1	0	0	3	0	0	0	0	0	0	0
19	825	1556	1514	1	1	1	1	2	3	4	5	6	7	8	9	10
20	250	1389	1417	1	0	0	1	0	0	0	0	0	0	0	0	0
21	325	945	1278	1	0	0	1	0	0	0	0	0	0	0	0	0
22	0	1556	1514	1	1	1	1	2	3	4	5	6	7	8	9	10
23	375	1806	1583	1	1	1	1	2	3	4	5	6	7	8	9	10
24	250	1708	1833	1	1	1	1	2	3	4	5	6	7	8	9	10
25	325	2111	1972	0	0	1	0	0	3	0	0	0	0	0	0	0
26	250	1611	2056	1	1	1	1	2	3	4	5	6	7	8	9	10
27	500	2250	2334	0	1	0	0	2	0	0	0	0	0	0	0	0
28	200	1528	2278	1	1	1	1	2	3	4	5	6	7	8	9	10
29	200	1750	2334	1	0	0	1	0	0	0	0	0	0	0	0	0
30	300	2056	2472	1	0	0	1	0	0	0	0	0	0	0	0	0
31	250	1445	2556	1	1	1	1	2	3	4	5	6	7	8	9	10
32	150	1278	2528	1	1	1	1	2	3	4	5	6	7	8	9	10
33	250	1667	2611	1	1	1	1	2	3	4	5	6	7	8	9	10
34	250	1917	2695	1	1	1	1	2	3	4	5	6	7	8	9	10
35	250	2139	2722	1	1	1	1	2	3	4	5	6	7	8	9	10
36	650	2361	1472	1	1	0	1	2	0	4	0	0	7	0	0	0
37	300	1861	1306	1	0	0	1	0	0	0	0	0	0	0	0	0
38	250	2528	1528	0	1	0	0	2	0	0	0	0	0	0	0	0
39	325	2847	1695	0	1	0	0	2	0	0	0	0	0	0	0	0
40	300	1417	1750	1	1	1	1	2	3	4	5	6	7	8	9	10
41	525	833	1528	0	1	0	0	2	0	0	0	0	0	0	0	0
42	250	1278	2000	1	1	1	1	2	3	4	5	6	7	8	9	10
43	550	722	1778	0	0	1	0	0	3	0	0	0	0	0	0	0
44	275	1181	2167	1	1	1	1	2	3	4	5	6	7	8	9	10
45	350	833	2042	1	0	1	1	0	3	0	0	6	0	0	9	0
46	225	750	2278	0	0	1	0	0	3	0	0	0	0	0	0	0
47	300	611	2639	0	0	1	0	0	3	0	0	0	0	0	0	0
48	275	542	1945	1	0	1	1	0	3	0	0	6	0	0	9	0
49	500	306	2528	1	0	0	1	0	0	0	0	0	0	0	0	0
50	200	1111	2306	1	1	1	1	2	3	4	5	6	7	8	9	10
51	300	1028	2556	1	1	1	1	2	3	4	5	6	7	8	9	10
52	350	945	2750	1	1	1	1	2	3	4	5	6	7	8	9	10
53	200	861	2917	1	1	1	1	2	3	4	5	6	7	8	9	10
54	0	1945	833	1	1	1	1	2	3	4	5	6	7	8	9	10
55	400	2222	917	1	1	1	1	2	3	4	5	6	7	8	9	10
56	200	2417	945	1	1	1	1	2	3	4	5	6	7	8	9	10
57	125	2500	972	1	1	1	1	2	3	4	5	6	7	8	9	10
58	275	2778	1056	1	1	1	1	2	3	4	5	6	7	8	9	10
59	250	2972	1139	1	1	1	1	2	3	4	5	6	7	8	9	10
60	350	2417	1306	1	1	1	1	2	3	4	5	6	7	8	9	10

Branch	Length in ft	Co-ordinate in ft (load_coordinate.txt)		Line Phase No			Fault code (faulttype_line.txt)									
		x_i	y_i	A	B	C	AN	BN	CN	AB	BC	CA	ABN	BCN	CAN	3N
61	250	2222	1222	0	1	0	0	2	0	0	0	0	0	0	0	0
62	250	2000	1139	0	1	0	0	2	0	0	0	0	0	0	0	0
63	750	3084	1500	1	1	1	1	2	3	4	5	6	7	8	9	10
64	550	3278	1000	1	1	1	1	2	3	4	5	6	7	8	9	10
65	0	3278	1000	1	1	1	1	2	3	4	5	6	7	8	9	10
66	250	2986	1750	1	1	1	1	2	3	4	5	6	7	8	9	10
67	175	2959	2000	1	1	1	1	2	3	4	5	6	7	8	9	10
68	350	2806	2306	1	1	1	1	2	3	4	5	6	7	8	9	10
69	425	2556	2334	1	1	1	1	2	3	4	5	6	7	8	9	10
70	325	2417	2250	1	1	1	1	2	3	4	5	6	7	8	9	10
71	0	3084	1500	1	1	1	1	2	3	4	5	6	7	8	9	10
72	350	3445	1667	1	1	1	1	2	3	4	5	6	7	8	9	10
73	250	3264	1917	1	1	1	1	2	3	4	5	6	7	8	9	10
74	275	3500	2028	1	1	1	1	2	3	4	5	6	7	8	9	10
75	550	3889	2250	1	1	1	1	2	3	4	5	6	7	8	9	10
76	300	4223	2472	1	1	1	1	2	3	4	5	6	7	8	9	10
77	800	4556	2667	1	1	1	1	2	3	4	5	6	7	8	9	10
78	0	3264	1917	1	1	1	1	2	3	4	5	6	7	8	9	10
79	250	3167	2111	1	1	1	1	2	3	4	5	6	7	8	9	10
80	225	3334	2222	0	0	1	0	0	3	0	0	0	0	0	0	0
81	325	3611	2417	0	0	1	0	0	3	0	0	0	0	0	0	0
82	700	4028	2695	0	0	1	0	0	3	0	0	0	0	0	0	0
83	275	3000	2389	1	1	1	1	2	3	4	5	6	7	8	9	10
84	225	3236	2500	0	1	0	0	2	0	0	0	0	0	0	0	0
85	575	3667	2778	0	1	0	0	2	0	0	0	0	0	0	0	0
86	325	2889	2556	1	1	1	1	2	3	4	5	6	7	8	9	10
87	1000	2667	3028	1	1	1	1	2	3	4	5	6	7	8	9	10
88	450	3278	2778	1	0	0	1	0	0	0	0	0	0	0	0	0
89	300	3500	2972	1	0	0	1	0	0	0	0	0	0	0	0	0
90	575	2834	2834	1	0	0	1	0	0	0	0	0	0	0	0	0
91	125	3500	2972	1	0	0	1	0	0	0	0	0	0	0	0	0
92	525	3945	3056	1	0	0	1	0	0	0	0	0	0	0	0	0
93	325	4389	3139	1	0	0	1	0	0	0	0	0	0	0	0	0
94	200	3653	1806	1	0	0	1	0	0	0	0	0	0	0	0	0
95	275	3861	1945	1	0	0	1	0	0	0	0	0	0	0	0	0
96	325	4056	2056	1	0	0	1	0	0	0	0	0	0	0	0	0
97	275	4264	2195	1	0	0	1	0	0	0	0	0	0	0	0	0
98	275	3556	1472	1	1	1	1	2	3	4	5	6	7	8	9	10
99	275	3723	1583	0	0	1	0	0	3	0	0	0	0	0	0	0
100	350	3973	1861	0	0	1	0	0	3	0	0	0	0	0	0	0
101	400	4306	1945	0	0	1	0	0	3	0	0	0	0	0	0	0
102	200	3639	1278	1	1	1	1	2	3	4	5	6	7	8	9	10
103	400	3834	1417	1	1	1	1	2	3	4	5	6	7	8	9	10
104	100	3986	1528	1	1	1	1	2	3	4	5	6	7	8	9	10
105	225	4139	1639	1	1	1	1	2	3	4	5	6	7	8	9	10
106	475	4028	1083	1	1	1	1	2	3	4	5	6	7	8	9	10
107	175	4084	889	1	1	1	1	2	3	4	5	6	7	8	9	10
108	675	4403	1236	0	0	1	0	0	3	0	0	0	0	0	0	0
109	475	4361	1611	0	0	1	0	0	3	0	0	0	0	0	0	0
110	250	4153	611	1	1	1	1	2	3	4	5	6	7	8	9	10
111	250	4195	389	1	1	1	1	2	3	4	5	6	7	8	9	10
112	700	3750	542	1	1	1	1	2	3	4	5	6	7	8	9	10
113	450	3445	486	1	1	1	1	2	3	4	5	6	7	8	9	10
114	175	3306	806	1	0	0	1	0	0	0	0	0	0	0	0	0
115	275	3222	444	1	1	1	1	2	3	4	5	6	7	8	9	10
116	250	3084	750	0	1	0	0	2	0	0	0	0	0	0	0	0
117	225	3014	417	1	1	1	1	2	3	4	5	6	7	8	9	10
118	300	2875	695	0	0	1	0	0	3	0	0	0	0	0	0	0
119	225	2834	361	1	1	1	1	2	3	4	5	6	7	8	9	10
120	275	2625	722	1	0	0	1	0	0	0	0	0	0	0	0	0
121	300	2584	250	1	1	1	1	2	3	4	5	6	7	8	9	10
122	200	2417	528	0	1	0	0	2	0	0	0	0	0	0	0	0

Table A.4. Scheme renumbering and line parameters

Sending node (renum-bering)	Receiving nodes (testfeder-123sys.txt)			Original of numbering	Linesegment numbering	Real part of line impedance in p.u. (Base : 5MVA & 4.16kV - linereal123.txt)						Imaginary part of line impedance in p.u. (Base : 5MVA & 4.16kV - lineimag123.txt)					
						A-self	A-B	A-C	B-self	B-C	C-self	A-self	A-B	A-C	B-self	B-C	C-self
1	2	0	0	s/s													
2	3	7	8	1	1	0.0087	0.002	0.002	0.0087	0.002	0.0087	0.031	0.0001	0.0001	0.031	0.0001	0.031
3	4	5	0	3	2	0	0	0	0	0	0.0182	0	0	0	0	0	0.0184
4	0	0	0	4	3	0	0	0	0	0	0.0145	0	0	0	0	0	0.0147
5	6	0	0	5	4	0	0	0	0	0	0.0236	0	0	0	0	0	0.024
6	0	0	0	6	5	0	0	0	0	0	0.0182	0	0	0	0	0	0.0184
7	0	0	0	2	6	0	0	0	0.0127	0	0	0	0	0	0.0129	0	0
8	9	0	0	7	7	0.0066	0.0015	0.0015	0.0066	0.0015	0.0066	0.0233	0.0001	0.0001	0.0233	0.0001	0.0233
9	10	11	15	8	8	0.0044	0.001	0.001	0.0044	0.001	0.0044	0.0155	0.0001	0.0001	0.0155	0.0001	0.0155
10	0	0	0	12	9	0	0	0	0.0164	0	0	0	0	0	0.0166	0	0
11	12	0	0	9	10	0.0164	0	0	0	0	0	0.0166	0	0	0	0	0
12	13	14	0	14	11	0.0309	0	0	0	0	0	0.0313	0	0	0	0	0
13	0	0	0	10	12	0.0182	0	0	0	0	0	0.0184	0	0	0	0	0
14	0	0	0	11	13	0.0182	0	0	0	0	0	0.0184	0	0	0	0	0
15	16	20	55	13	14	0.0066	0.0015	0.0015	0.0066	0.0015	0.0066	0.0233	0.0001	0.0001	0.0233	0.0001	0.0233
16	17	0	0	34	15	0	0	0	0	0	0.0109	0	0	0	0	0	0.0111
17	18	19	0	15	16	0	0	0	0	0	0.0073	0	0	0	0	0	0.0074
18	0	0	0	16	17	0	0	0	0	0	0.0255	0	0	0	0	0	0.0258
19	0	0	0	17	18	0	0	0	0	0	0.0273	0	0	0	0	0	0.0277
20	21	23	41	18	19	0.018	0.0042	0.0042	0.018	0.0042	0.018	0.064	0.0003	0.0003	0.064	0.0002	0.064
21	22	0	0	19	20	0.0182	0	0	0	0	0	0.0184	0	0	0	0	0
22	0	0	0	20	21	0.0236	0	0	0	0	0	0.024	0	0	0	0	0
23	24	0	0	135	22	0	0	0	0	0	0	0	0	0	0	0	0
24	25	37	0	35	23	0.0082	0.0019	0.0019	0.0082	0.0019	0.0082	0.0291	0.0001	0.0001	0.0291	0.0001	0.0291
25	26	27	0	40	24	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
26	0	0	0	41	25	0	0	0	0	0	0.0236	0	0	0	0	0	0.024
27	28	29	0	42	26	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
28	0	0	0	43	27	0	0	0	0.0364	0	0	0	0	0	0.0369	0	0
29	30	32	0	44	28	0.0044	0.001	0.001	0.0044	0.001	0.0044	0.0155	0.0001	0.0001	0.0155	0.0001	0.0155
30	31	0	0	45	29	0.0145	0	0	0	0	0	0.0147	0	0	0	0	0
31	0	0	0	46	30	0.0218	0	0	0	0	0	0.0221	0	0	0	0	0
32	33	34	0	47	31	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
33	0	0	0	48	32	0.0033	0.0008	0.0008	0.0033	0.0008	0.0033	0.0116	0	0	0.0116	0	0.0116
34	35	0	0	49	33	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
35	36	0	0	50	34	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
36	0	0	0	51	35	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
37	38	39	0	36	36	0.0163	0.0055	0	0.0164	0	0	0.0383	0.0137	0	0.0379	0	0
38	0	0	0	37	37	0.0218	0	0	0	0	0	0.0221	0	0	0	0	0
39	40	0	0	38	38	0	0	0	0.0182	0	0	0	0	0	0.0184	0	0
40	0	0	0	39	39	0	0	0	0.0236	0	0	0	0	0	0.024	0	0
41	42	43	0	21	40	0.0066	0.0015	0.0015	0.0066	0.0015	0.0066	0.0233	0.0001	0.0001	0.0233	0.0001	0.0233
42	0	0	0	22	41	0	0	0	0.0382	0	0	0	0	0	0.0387	0	0
43	44	45	0	23	42	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
44	0	0	0	24	43	0	0	0	0	0	0.04	0	0	0	0	0	0.0406
45	46	51	0	25	44	0.006	0.0014	0.0014	0.006	0.0014	0.006	0.0213	0.0001	0.0001	0.0213	0.0001	0.0213
46	47	49	0	26	45	0.0088	0	0.0029	0	0	0.0088	0.0206	0	0.0074	0	0	0.0204
47	48	0	0	27	46	0	0	0	0	0	0.0164	0	0	0	0	0	0.0166
48	0	0	0	33	47	0	0	0	0	0	0.0218	0	0	0	0	0	0.0221
49	50	0	0	31	48	0.0069	0	0.0023	0	0	0.0069	0.0162	0	0.0058	0	0	0.016
50	0	0	0	32	49	0.0364	0	0	0	0	0	0.0369	0	0	0	0	0
51	52	0	0	28	50	0.0044	0.001	0.001	0.0044	0.001	0.0044	0.0155	0.0001	0.0001	0.0155	0.0001	0.0155
52	53	0	0	29	51	0.0066	0.0015	0.0015	0.0066	0.0015	0.0066	0.0233	0.0001	0.0001	0.0233	0.0001	0.0233
53	54	0	0	30	52	0.0076	0.0018	0.0018	0.0076	0.0018	0.0076	0.0272	0.0001	0.0001	0.0272	0.0001	0.0272
54	0	0	0	250	53	0.0044	0.001	0.001	0.0044	0.001	0.0044	0.0155	0.0001	0.0001	0.0155	0.0001	0.0155
55	56	0	0	152	54	0	0	0	0	0	0	0	0	0	0	0	0
56	57	0	0	52	55	0.0087	0.002	0.002	0.0087	0.002	0.0087	0.031	0.0001	0.0001	0.031	0.0001	0.031
57	58	0	0	53	56	0.0044	0.001	0.001	0.0044	0.001	0.0044	0.0155	0.0001	0.0001	0.0155	0.0001	0.0155
58	59	61	0	54	57	0.0027	0.0006	0.0006	0.0027	0.0006	0.0027	0.0097	0	0	0.0097	0	0.0097
59	60	0	0	55	58	0.006	0.0014	0.0014	0.006	0.0014	0.006	0.0213	0.0001	0.0001	0.0213	0.0001	0.0213
60	0	0	0	56	59	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
61	62	64	0	57	60	0.0076	0.0018	0.0018	0.0076	0.0018	0.0076	0.0272	0.0001	0.0001	0.0272	0.0001	0.0272

Sending node (renum-bering)	Receiving nodes (testfeder-123sys.txt)			Original of numbering	Linesegment numbering	Real part of line impedance in p.u. (Base : 5MVA & 4.16kV - linereal123.txt)						Imaginary part of line impedance in p.u. (Base : 5MVA & 4.16kV - lineimag123.txt)					
						A-self	A-B	A-C	B-self	B-C	C-self	A-self	A-B	A-C	B-self	B-C	C-self
62	63	0	0	58	61	0	0	0	0.0182	0	0	0	0	0	0.0184	0	0
63	0	0	0	59	62	0	0	0	0.0182	0	0	0	0	0	0.0184	0	0
64	65	67	72	60	63	0.0164	0.0038	0.0038	0.0164	0.0038	0.0164	0.0582	0.0002	0.0002	0.0582	0.0002	0.0582
65	66	0	0	61	64	0.012	0.0028	0.0028	0.012	0.0028	0.012	0.0427	0.0002	0.0002	0.0427	0.0002	0.0427
66	0	0	0	610	65	0	0	0	0	0	0	0	0	0	0	0	0
67	68	0	0	62	66	0.0009	0	0	0.0009	0	0.0009	0.0001	0	0	0.0001	0	0.0001
68	69	0	0	63	67	0.0006	0	0	0.0006	0	0.0006	0.0001	0	0	0.0001	0	0.0001
69	70	0	0	64	68	0.0012	0	0	0.0012	0	0.0012	0.0001	0	0	0.0001	0	0.0001
70	71	0	0	65	69	0.0014	0	0	0.0014	0	0.0014	0.0001	0	0	0.0001	0	0.0001
71	0	0	0	66	70	0.0011	0	0	0.0011	0	0.0011	0.0001	0	0	0.0001	0	0.0001
72	73	0	0	160	71	0	0	0	0	0	0	0	0	0	0	0	0
73	74	95	99	67	72	0.0076	0.0018	0.0018	0.0076	0.0018	0.0076	0.0272	0.0001	0.0001	0.0272	0.0001	0.0272
74	75	79	0	97	73	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
75	76	0	0	98	74	0.006	0.0014	0.0014	0.006	0.0014	0.006	0.0213	0.0001	0.0001	0.0213	0.0001	0.0213
76	77	0	0	99	75	0.012	0.0028	0.0028	0.012	0.0028	0.012	0.0427	0.0002	0.0002	0.0427	0.0002	0.0427
77	78	0	0	100	76	0.0066	0.0015	0.0015	0.0066	0.0015	0.0066	0.0233	0.0001	0.0001	0.0233	0.0001	0.0233
78	0	0	0	450	77	0.0175	0.0041	0.0041	0.0175	0.0041	0.0175	0.0621	0.0002	0.0002	0.0621	0.0003	0.0621
79	80	0	0	197	78	0	0	0	0	0	0	0	0	0	0	0	0
80	81	84	0	101	79	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
81	82	0	0	102	80	0	0	0	0	0	0.0164	0	0	0	0	0	0.0166
82	83	0	0	103	81	0	0	0	0	0	0.0236	0	0	0	0	0	0.024
83	0	0	0	104	82	0	0	0	0	0	0.0509	0	0	0	0	0	0.0516
84	85	87	0	105	83	0.006	0.0014	0.0014	0.006	0.0014	0.006	0.0213	0.0001	0.0001	0.0213	0.0001	0.0213
85	86	0	0	106	84	0	0	0	0.0164	0	0	0	0	0	0.0166	0	0
86	0	0	0	107	85	0	0	0	0.0418	0	0	0	0	0	0.0424	0	0
87	88	89	0	108	86	0.0071	0.0017	0.0017	0.0071	0.0017	0.0071	0.0252	0.0001	0.0001	0.0252	0.0001	0.0252
88	0	0	0	350	87	0.0218	0.0051	0.0051	0.0218	0.0051	0.0218	0.0776	0.0003	0.0003	0.0776	0.0003	0.0776
89	90	0	0	109	88	0.0327	0	0	0	0	0	0.0332	0	0	0	0	0
90	91	92	0	110	89	0.0218	0	0	0	0	0	0.0221	0	0	0	0	0
91	0	0	0	111	90	0.0418	0	0	0	0	0	0.0424	0	0	0	0	0
92	93	0	0	112	91	0.0091	0	0	0	0	0	0.0092	0	0	0	0	0
93	94	0	0	113	92	0.0382	0	0	0	0	0	0.0387	0	0	0	0	0
94	0	0	0	114	93	0.0236	0	0	0	0	0	0.024	0	0	0	0	0
95	96	0	0	68	94	0.0145	0	0	0	0	0	0.0147	0	0	0	0	0
96	97	0	0	69	95	0.02	0	0	0	0	0	0.0203	0	0	0	0	0
97	98	0	0	70	96	0.0236	0	0	0	0	0	0.024	0	0	0	0	0
98	0	0	0	71	97	0.02	0	0	0	0	0	0.0203	0	0	0	0	0
99	100	103	0	72	98	0.006	0.0014	0.0014	0.006	0.0014	0.006	0.0213	0.0001	0.0001	0.0213	0.0001	0.0213
100	101	0	0	73	99	0	0	0	0	0	0.02	0	0	0	0	0	0.0203
101	102	0	0	74	100	0	0	0	0	0	0.0255	0	0	0	0	0	0.0258
102	0	0	0	75	101	0	0	0	0	0	0.0291	0	0	0	0	0	0.0295
103	104	113	0	76	102	0.0044	0.001	0.001	0.0044	0.001	0.0044	0.0155	0.0001	0.0001	0.0155	0.0001	0.0155
104	105	0	0	77	103	0.0087	0.002	0.002	0.0087	0.002	0.0087	0.031	0.0001	0.0001	0.031	0.0001	0.031
105	106	107	0	78	104	0.0022	0.0005	0.0005	0.0022	0.0005	0.0022	0.0078	0	0	0.0078	0	0.0078
106	0	0	0	79	105	0.0049	0.0011	0.0011	0.0049	0.0011	0.0049	0.0175	0.0001	0.0001	0.0175	0.0001	0.0175
107	108	0	0	80	106	0.0104	0.0024	0.0024	0.0104	0.0024	0.0104	0.0369	0.0001	0.0002	0.0369	0.0001	0.0369
108	109	111	0	81	107	0.0038	0.0009	0.0009	0.0038	0.0009	0.0038	0.0136	0.0001	0.0001	0.0136	0.0001	0.0136
109	110	0	0	82	108	0	0	0	0	0	0.0491	0	0	0	0	0	0.0498
110	0	0	0	83	109	0	0	0	0	0	0.0345	0	0	0	0	0	0.035
111	112	0	0	84	110	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
112	0	0	0	85	111	0.0055	0.0013	0.0013	0.0055	0.0013	0.0055	0.0194	0.0001	0.0001	0.0194	0.0001	0.0194
113	114	0	0	86	112	0.0153	0.0036	0.0036	0.0153	0.0036	0.0153	0.0543	0.0002	0.0002	0.0543	0.0002	0.0543
114	115	116	0	87	113	0.0098	0.0023	0.0023	0.0098	0.0023	0.0098	0.0349	0.0001	0.0001	0.0349	0.0001	0.0349
115	0	0	0	88	114	0.0127	0	0	0	0	0	0.0129	0	0	0	0	0
116	117	118	0	89	115	0.006	0.0014	0.0014	0.006	0.0014	0.006	0.0213	0.0001	0.0001	0.0213	0.0001	0.0213
117	0	0	0	90	116	0	0	0	0.0182	0	0	0	0	0	0.0184	0	0
118	119	120	0	91	117	0.0049	0.0011	0.0011	0.0049	0.0011	0.0049	0.0175	0.0001	0.0001	0.0175	0.0001	0.0175
119	0	0	0	92	118	0	0	0	0	0	0.0218	0	0	0	0	0	0.0221
120	121	122	0	93	119	0.0049	0.0011	0.0011	0.0049	0.0011	0.0049	0.0175	0.0001	0.0001	0.0175	0.0001	0.0175
121	0	0	0	94	120	0.02	0	0	0	0	0	0.0203	0	0	0	0	0
122	123	0	0	95	121	0.0066	0.0015	0.0015	0.0066	0.0015	0.0066	0.0233	0.0001	0.0001	0.0233	0.0001	0.0233
123	0	0	0	96	122	0	0	0	0.0145	0	0	0	0	0	0.0147	0	0