# OPTIMAL LINE SEPARATION IN POWER DISTRIBUTION SYSTEM

## A DISSERTATION

submitted in partial fulfillment of the
requirements for the award of the degree
of
MASTER OF TECHNOLOGY
in
WATER RESOURCES DEVELOPMENT
*( Hydro-Electric System Engineering & Management )*

By
**DINESH KUMAR GHIMIRE**



WATER RESOURCES DEVELOPMENT TRAINING CENTRE
INDIAN INSTITUTE OF TECHNOLOGY, ROORKEE
ROORKEE - 247 667 (INDIA)
DECEMBER, 2001

# CANDIDATE'S DECLARATION

I hereby declare that the work, which is being presented in this dissertation entitled " **OPTIMAL LINE SEPARATION IN POWER DISTRIBUTION SYSTEM"** in partial fulfillment of the requirement for the award of the Degree of Master of Technology in Water Resources Development (Hydro Electric System Engineering and Management) submitted in Water Resources Development Training Centre, Indian Institute of Technology, Roorkee is an authentic record of my own work carried out during a period from July 15, 2001 to December 16, 2001 under the supervision of **Prof. Devadutta Das,** ***Professor & Head, WRDTC,*** **IIT Roorkee** and **Dr. Hari Om Gupta,** ***Professor, Department of Electrical Engineering,*** IIT Roorkee,

I have not submitted the matter embodied in this dissertation for award of any other degree.

Dated: Dec. ,2001
Place: Roorkee

**(DINESH KUMAR GHIMIRE)**
Trainee Officer, 45th WRD(HESM)

---

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

18/12/01

**(Dr. HARI OM GUPTA)**
Professor,
Deptt. of Elect. Engg.
IIT, Roorkee-247667

**(Prof. DEVADUTTA DAS)**
Professor & Head
W.R.D.T.C.
IIT, Roorkee - 247667

# ACKNOWLEDGEMENT

I express my gratitude and sincere thanks to Prof. Devadutta Das, Professor & Head WRDTC, Indian Institute of Technology, Roorkee and Dr. Hari Om Gupta, Professor, Department of Electrical Engineering, Indian Institute of Technology, Roorkee for their constant guidance and timely suggestions without which this study would have been impossible. I am also thankful to all the faculty members and staffs of WRDTC for their kind co-operation rendered to me directly or indirectly.

I extend my thanks to Dr. Sunita Devi, Programmer, Computer Lab, WRDTC and all other Computer staffs for providing every facility available in the Lab. required for my dissertation work.

Thanks are also due to His Majesty's Government of Nepal for giving me the opportunity and sanctioning my study leave for this course.

I am equally obliged to all of my colleagues for their direct and indirect help during my stay at Roorkee.

My parents and In-laws also deserve my sincere thanks for their continuous moral support and blessings. Last but not the least, I extremely appreciate my wife Binita Ghimire, son Anish and daughter Ichchha for their full support, love and prayers without which my study would not be successful.

**(DINESH KUMAR GHIMIRE)**

## ABSTRACT

The electricity being an important element for the growth in industrial and business development, the consumer expectations for quality and uninterrupted supply have gone up. With increasing pace of liberalization , particularly the private company's participation in the power sector, the electricity will no more be a monopoly. With a goal of reliable and good quality electrical power supply, no utility can afford absence of a good automation system to monitor and control the distribution network. Thus to cope with the increasing demand for electrical energy together with its quality and reliability, adequate transmission and distribution facilities need to be built up. Apart from this, the system is to be continuously monitored to ensure reliable and uninterrupted power. The distribution automation system (DAS) is being extensively employed now-a-days in which a lot of sensors are placed in the power distribution system. These sensors communicate the system status very fast to the central computer station so that corrective measures could be taken.

One of the most important functions of DAS is to reconfigure the feeder for service restoration in case of fault in a particular feeder. The present study deals with the most economical arrangement that should be made for service restoration in a radial distribution system when particular feeder develops a fault.

Due to the low first cost and simplicity, power distribution networks are ordinarily radial in nature except in case where bulk load is to be supplied. The radial network consists of a straightforward circuit arrangement with single path from the distribution substation making switching and relaying scheme very simple.

The major drawback of the radial system is that a fault in a particular feeder results in service interruption to all the loads supplied through it. To overcome this major drawback of the radial system, a provision is made to provide alternate supply from the nearby feeder through a line interconnected between these feeders. The line is normally

switched off and comes into operation only when supply of one of the feeders is disrupted. Thus with this arrangement, each feeder serves as a normal feed to all load connected in it and an emergency feed to the load of nearby feeders. The line consisting of this interconnecting switch is termed as the **"separation line"**.

In the large distribution system with varying load profiles at various buses, placement of the "separation line" becomes crucial from the view point of losses. The provision of the "seperation line" already exists almost everywhere in the Indian electric utilities. But this is there just to overcome the social pressure and no attention has been paid toward minimizing losses and saving energy. **The present study has been made with an objective of developing an algorithm and a software for determining the optimal placement of the "separation line" from the point of view of minimizing the losses in the system.** The operation cost of the system will also be reduced without much expenses.

The problem of optimal line separation can be assessed in two ways. First is to find the optimal placement of the line for a new distribution system. Second is to verify the correctness of the already fixed separation line in an existing system. The second type of problem has been considered in the present study.

The methodology used for solving the problem starts with the load flow solution of the given distribution network. Since large sparse networks have to be dealt with, a "linknet" structure [10] has been used for storing the information regarding network topology to do away with the problem of computer memory. Initially, the "separation lines" of the existing system are reconnected and the network is made non-radial giving rise to n loops for n separation lines. This non-radial network is solved using the Gauss Seidel algorithm [11] to determine the probable pairs of lines meeting at a node where power is supplied from two or more adjacent nodes. Out of these pairs of lines, all possible combinations of separation lines are formed and the network is again solved for each of these combinations using Z-loop algorithm [3] for a given load pattern. The combination giving the lowest energy loss is then selected as the optimal separation line.

A computer software in $C^{++}$ has been developed based on the stated methodology. The validity of the developed software has been checked by running the software for two real 38 and 114 bus, 33 KV power distribution networks of Bhopal city, India. The system data are taken as obtained from Madhya Pradesh State Electricity Board (MPSEB) of India. It has been found that the system losses can be reduced and monetary saving can be accrued without any additional investment just by changing the existing switching location as per the results obtained. Moreover the voltage profile of the buses has been found to be more smooth for the suggested solution.

# CONTENTS

# LIST OF FIGURES

## LIST OF TABLES

**CHAPTER – 1**

# INTRODUCTION

## 1.1 GENERAL

With the rising standards of living, the public is becoming more demanding with respect to the reliability of essential services. At the same time due to the rapid increase in demand for electrical energy the size and complexity of power distribution networks have grown tremendously. Therefore to supply quality power to the ultimate customers adequate transmission and distribution facilities need to be built up and consequently there have also been substantial developments in the power transmission and distribution system.

Apart from building up the infrastructures of distribution system, it is also very important to continuously monitor the system to ensure safe, reliable and uninterruptible power supply. Towards this objective, supervisory control and data acquisition (SCADA) system have been employed worldwide to continuously monitor and control transmission system. Similarly to continuously monitor and control the distribution system, Distribution Automation System (DAS) have emerged a few years ago. In a distribution automation system, a lot of sensors and remote terminal units (RTUs) are placed in the power distribution system. The data controlled by these sensor and RTUs are sent to the central computer station over a dedicated communication channel. The received data are then analyzed by appropriate analysis software to assess the present health of the system. If any anomaly in the operation of the distribution system is detected proper remedial and control decision are taken at the central computer station with the help of proper software analysis packages. These control decisions are then sent to the network through the same or other dedicated communication channel for proper implementation.

Obviously even with the presence of accurate sensors, RTUs and robust reliable communication channel, the effectiveness of DAS largely depends upon the quality of the analysis software at the central computer station. As all the decision regarding the present health and remedial control action are to be taken by the software analysis packages, the algorithm of the software packages must be such that the results produced are highly accurate. Moreover as the health monitoring function and control decision function are to be carried out quickly to minimize the adverse effect of any untoward condition in a power distribution system, the algorithms of the software analysis packages need to be very fast.

The most common different control decision function which are currently being employed in any modern distribution automation system are:

- Feeder reconfiguration to minimize the loss in the system
- Volt -VAR control
- Accurate fault location in a distribution system
- Feeder reconfiguration for service restoration
- Feeder load balancing
- Demand side management
- Remote monitoring

Among the above listed functions, feeder reconfiguration for service restoration and loss minimization is very important function in power distribution system.

Feeder reconfiguration to restore the service includes a provision of an interconnecting switch between the two neighboring feeders of a radial distribution system. At the time of a fault occurring in any of the feeder, Distribution Automation System senses the fault and closes the switch automatically to transfer the load of unhealthy feeders to the healthy feeder. The provision of this normally open interconnecting switch already exists in almost of the State Electricity Boards of India.

However no logical practice have been evolved so far as the location of the switch is concerned. The interconnection provision is provided just to satisfy the social pressures. No care has been taken to minimize the losses and save energy.

The location of the interconnecting switch is crucial from the view point of system losses especially in a large system with varying load profiles. Since the provision of the switch already exists, determining its optimal location so as to have minimum energy loss would save considerable amount of money without any additional effort and investment. A methodology and a software has been developed to find the said optimum location in the present study.

## 1.2 THE DISTRIBUTION SYSTEM

An electric distribution system is all of that part of an electric power system between the bulk power sources and the consumer's service switches. The bulk power sources are located in or near the load area to be served by the distribution system and may be either generating station or power substation supplied over transmission lines. Distribution systems can, in general, be divided into six parts: namely sub transmission circuits, distribution substation, distribution or primary feeders, distribution transformer, secondary circuits or secondaries and consumers service connection and meters or consumers services.

The various components in a distribution system are shown in figure 1.1 [5]. The sub-transmission circuits extend from bulk power source or sources to the various distribution substation location in the load areas. They may be radial circuits connected to bulk power sources at only one end or loop and ring circuits connected to one or more power sources at both ends. The sub-transmission circuits consist of underground cable, aerial cable or overhead open conductors carried on poles or some combination of them . The sub-transmission voltage is usually between 33 KV and 11 KV inclusive.

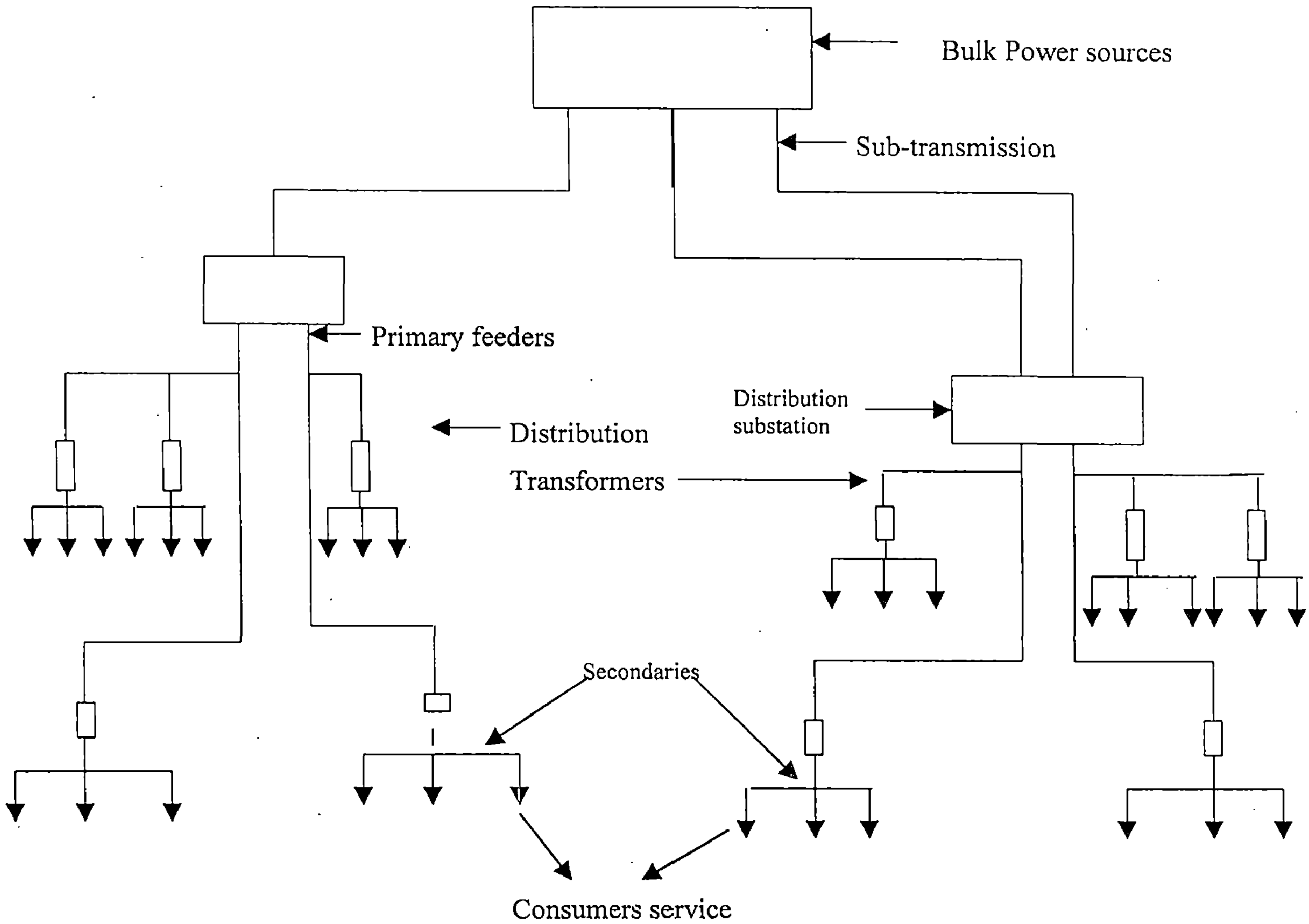



**Fig. 1.1: Components in a typical Distribution System**

Each distribution substation serves its own load area which is a sub-division of the area served by the distribution system. At the distribution sub-station, the sub-transmission voltage is reduced for general distribution through out the area. The substation consists of one or more power transformers together with the necessary voltage regulating equipment, buses, switch gear etc.

The area served by the distribution sub-station is also sub-divided and each sub-division is supplied by a distribution or primary feeder. The three-phase primary feeder emerges from the low voltage bus of the substation to its load centre where it branches into three phase sub-feeders and single phase laterals. The primary feeders and laterals may be either cable or open wire circuits operated in most cases 415 V or 230 volts.

Distribution transformers are ordinarily connected to each primary feeders and its sub-feeders and laterals. These transformers serve to step down from the distribution voltage to the utilization voltage. Each transformer or bank of transformer supplies consumers or group of consumers over its secondary circuits. Each consumer is connected to the secondary circuit through service leads and meter. The secondaries and service connectors may be either cable or open wire circuits.

The distribution plant occupies an important place in an electric power system. Briefly its function is to take electric power from the bulk power sources and deliver it to the consumers. The effectiveness with which a distribution system fulfills this function is measured in terms of voltage regulation, service continuity, flexibility, efficiency and cost. The cost of the distribution is an important factor in the delivered cost of electric power.

Briefly the problem of distribution is to design, construct, operate and maintain a distribution system that will supply adequate electric service to the load area under consideration both now and in the future at the lowest possible cost. Unfortunately, no one type of distribution system can be applied economically in all load areas, because of difference in load densities, existing distribution plant topography and other local conditions.

For different load areas or even different parts of the same load area the most effective distribution system will often take different forms. Certain principle and features however are common to almost all of these systems. The distribution system should provide service with a minimum voltage variation and a minimum of interruptions. Service interruptions should be of short duration and affect a small number of consumers. The overall system cost including construction operation and maintenance of the system should be as low as possible consistent with the quality of service required in the load area. The system should be flexible to allow its being expanded in small increments so as to meet changing load condition with a minimum amount of modification and expense. This flexibility permits keeping the system capacity close to actual load requirements and thus permits the most effective use of system investment. It also largely eliminates the need for predicting the location and magnitudes of future loads.

Two types of distribution system are commonly used. The loop system and the Radial system. The loop system is used most frequently to supply bulk loads, such as small industrial plants & medium or large commercial buildings where continuity of service is of considerable importance. The radial system is used extensively to serve the light and medium density load areas where the primary and secondary circuits are usually carried overhead on poles. Since most of the general distribution networks are radial in nature and also since the scope of present study is for the radial system, it is explained in detail below.

## 1.3 THE RADIAL SYSTEM

The Radial system gets its name from the fact that the primary feeders radiate from the distribution sub-station and branch into sub-feeder and laterals which extend into all parts of the area served. The distribution transformers are connected to the primary feeders, sub-feeders and supply the radial secondary circuits to which the consumers services are connected. The distribution substations can be supplied from the bulk power source over radial circuits as shown in figure 1.2 [5].

Fundamentally, the advantage of the radial distribution system are simplicity and low first cost. These result from a straightforward circuit arrangement, where a single or radial path is provided from the distribution substation and sometimes from the bulk power sources to the consumer. With such a circuit arrangement, the amount of switching equipment is small and the protective relaying is simple.

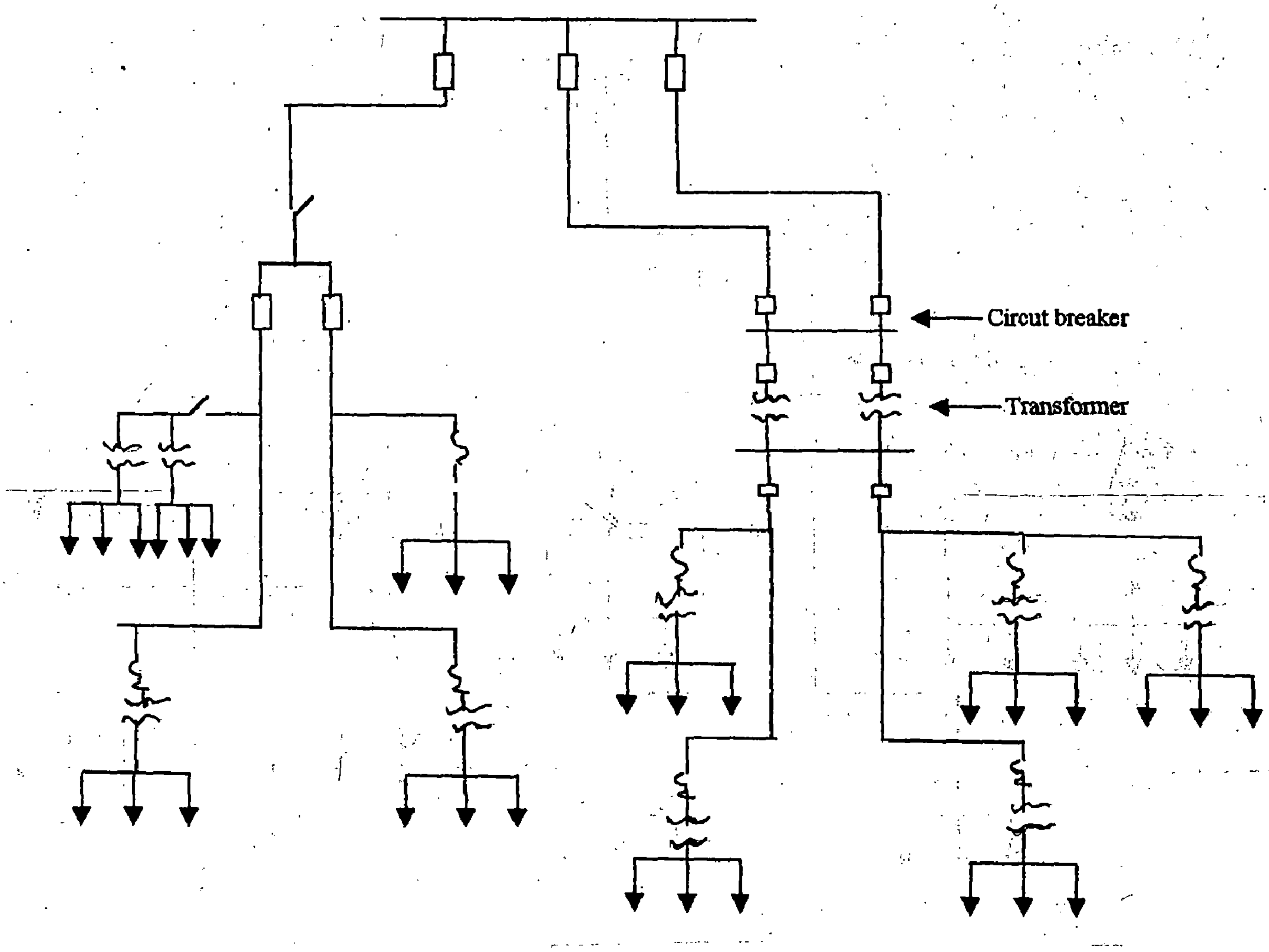



**Fig.1.2: Simple form of Radial type distribution system.**

The lack of continuity of service is the major drawback with the radial system of distribution. Attempts to overcome this drawback have resulted in many forms and arrangements of the radial system. One of these attempts is the provision of the "separation line" which consists of a normally open switch for operation at the time of fault in a particular feeder.

## 1.4 THE SEPARATION LINE

A radial arrangement of sub-transmission circuits such as that shown in Fig.1.2 results in the lowest first cost. But this form of sub-transmission is not usually employed because of the poor service reliability it provides. A fault on a radial sub-transmission circuit results in a service interruption to all loads fed over it.

The economical use of sub-transmission circuits and associated circuit breakers dictates that each sub-transmission circuit carries relatively large block of load. Then a fault on a radial sub-transmission circuit results in the loss of considerable load which usually means that a large area and many consumers are without service.

To overcome this major disadvantage of radial distribution system, normally open switches are provided between the two neighboring feeders. A typical 33 KV Sub-transmission network with provision of such switches is shown in Fig. 1.3 [2 ]. This kind of arrangement permits quick restoration of service when a radial sub-transmission circuit develops a fault. Thus with this arrangement, each radial sub-transmission circuit serves as a normal feed to certain distribution sub-station transformer and as an emergency feed to others. This switching arrangement is located along a line connecting the two feeder. This line is termed as the "separation line".

As already stated the placement of the separation line is crucial from the view point of system losses. Considerable amount of losses can be reduced and monetary savings could be accrued without any additional effort and investment by selecting the switching location optimally.

## 1.5 OBJECTIVE OF STUDY

The objective of the present study is to develop the methodology and a software to determine the optimal location of the separation line so as to minimize the overall energy losses in the system.

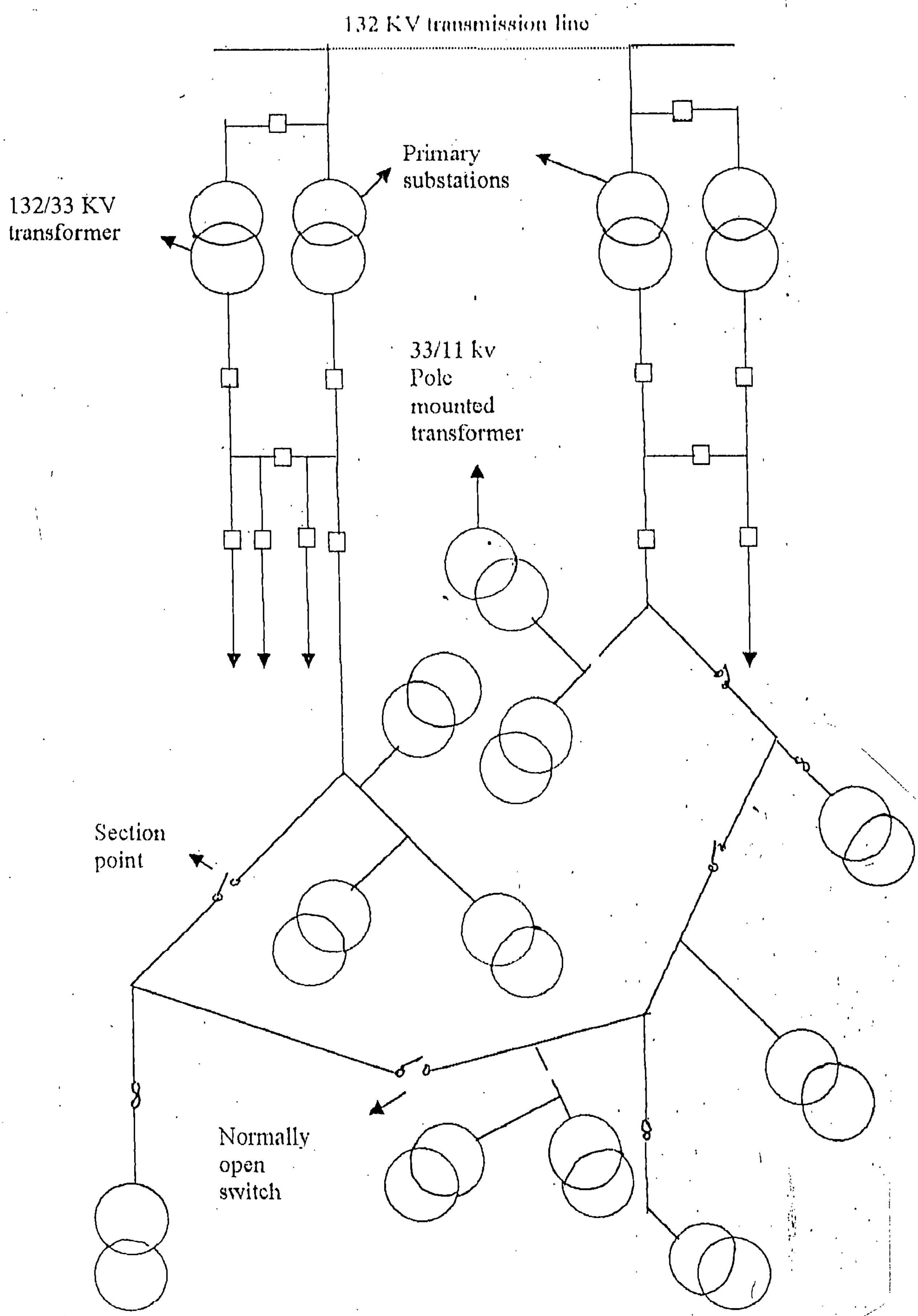



Fig. 1.3 Provision of Normally open switch in radial distribution system

## 1.6 AUTHOR'S CONTRIBUTION

Following are the author's contribution in the present thesis:

i) A software called "linknet" [10] has been developed in $C^{++}$ to store the information regarding connectivity of the network under consideration.

ii) A software in $C^{++}$ has been developed to find the load flow solution of non radial/radial network using Gauss Seidel algorithm [11].

iii) A software in $C^{++}$ has been developed to reconfigure the network after the line separation so that Z-loop algorithm [3] can be applied with the same input data.

iv) A software has been developed in $C^{++}$ to find the load flow solution of the radial distribution system using Z-loop algorithm.

v) All these software are interlinked to find that option of separation line which gives minimum energy losses for a system with a given load pattern.

## 1.7 ORGANIZATION OF THESIS

The thesis has been composed of six chapters. Chapter -1 deals with the general introduction to the power distribution system and its types especially the radial one. This chapter also deals with the significance of the "separation line" and the objective of the study. Author's contributions are also presented in this chapter.

Chapter -2 explains about the components of distribution network modelling including the representation of transformers along the distribution lines.

Chapter -3 describes the "linknet structure" which is used in the development of software for storing information regarding the network topology.

Chapter -4 presents the different methods of load flow solution and the methodology and algorithms for software development. The flow charts for each of the algorithm are also given.

In chapter -5, the developed software has been tested for two real 38 and 114 bus power distribution system of Bhopal, India. The results obtained are presented alongwith the discussions.

Chapter -6 is the last chapter of the thesis consisting of the conclusion and recommendations.

# MODELING OF DISTRIBUTION NETWORK

## 2.1 INTRODUCTION

When carrying out the calculations, it is necessary to make use of equivalent circuits for various components and then to combine these circuits in order to represent the interconnection of the components in the actual network.

The relatively short length of MV and LV distribution circuits enable simplest modelling techniques to be used for lines. The radial network configuration commonly used make it possible to simplify the network model and large matrix models are seldom necessary. It is normally sufficiently accurate to ignore the capacitance of a distribution circuit and represent it by a series impedance except when carrying out voltage calculations on a long cable.

Transformers can be represented by shunt and series impedances. The smaller distribution transformer have a larger series resistance than reactance while the large power transformers have negligible resistance compared to reactance. In the latter case, neglecting resistance has little effect on voltage drop study but resistance should be taken into account when calculating real power and energy losses.

## 2.2 NETWORK REPRESENTATION

A radial distribution system is modelled as a network of buses connected by distribution lines, switches or transformer to a voltage specified bus. Each bus may have corresponding load. The shunt capacitance are often ignored for distribution lines. Thus the distribution circuit can be represented as follows:

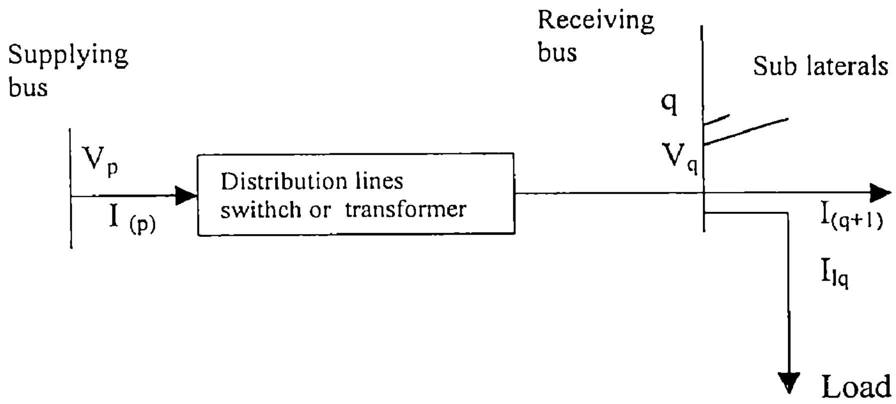



**Fig. 2.1: Distribution Network Modelling**

The voltage and current at one bus can be expressed as a function of the voltage and current of next bus. The relationship between different parameters in given below:

**In case of distribution line:**

$$I_p=I_{lq}+I_{q+1}+\sum_{j\in A} I_j \qquad (1)$$

$$V_p=V_q+I_p*Z_{pq} \qquad (2)$$

Where $I_p$ is the current flowing from bus p to bus q.

$I_{q+1}$ is the current flowing from bus q to next bus connected to it.

$I_{lq}$ is the current flowing through the load.

$I_j$ is the current flowing from bus q to the jth sub lateral.

A is the set of sub-laterals emerging out of the bus q.

$V_p$ is the voltage of bus p

$V_q$ is the voltage of bus q

$Z_{pq}$ is the impedence of the line connected between the bus p and q

**In case of switch:**

$$V_p = V_q \tag{3}$$

**In case of transformers:**

When the two buses of the system are separated by a transformer of off - nominal ratio a:1 the transformer is represented by its leakage admittance in series with an ideal auto-transformer as shown below [8].

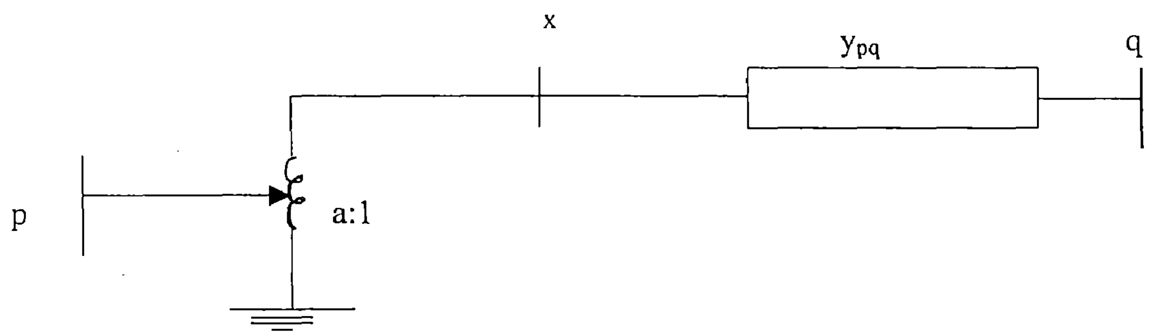



**Fig. 2.2 : Representation of Transformer in Distribution Lines**

The current flow in the transformer in the direction from x to q is given by

$$I=(V_x-V_q)y_{pq} \tag{4}$$

Now,

$$V_p/V_x=a \tag{5}$$

Since there is no power loss in the auto-transformer,

$$V_p{}^*I_p=V_x{}^*I \tag{6}$$

from which,

$$I_p=\frac{V_x^*}{V_p^*}I = \frac{1}{a}\left(\frac{V_p}{a}-V_q\right)y_{pq} \tag{7}$$

Similarly the current flow in the transformer from q to x is given by

$$I_q = (V_q - V_x) y_{pq}$$

$$= (V_q - \frac{V_p}{a}) y_{pq} \qquad (8)$$

The above expressions of current suggest a $\pi$ equivalent circuit as shown below:

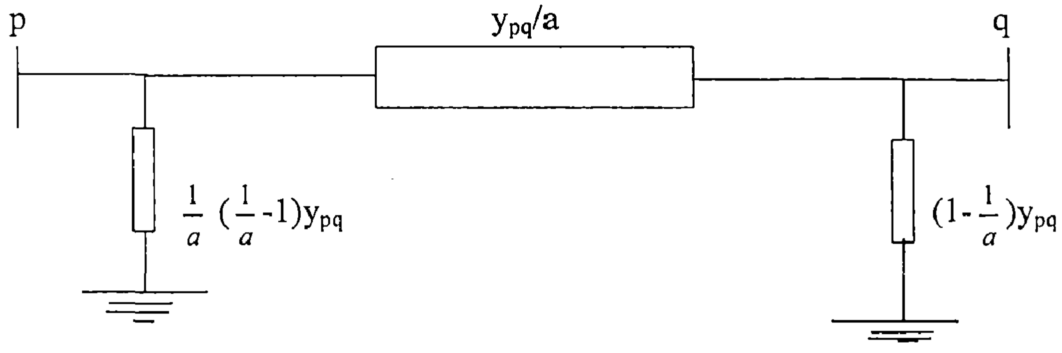



**Fig. 2.3 : Equivalent $\pi$ Circuit of Transformer Representation.**

The driving point admittance of p[th] bus becomes $y/a^2$ and that of qth bus remains y. The transfer admittance becomes( -y/a). The admittances are shown in the matrix form below:

$$\begin{bmatrix} \frac{y_{pq}}{a^2} & \frac{-y}{a} \\ \frac{-y_{pq}}{a} & y_{pq} \end{bmatrix}$$

The power flow from bus p is given by

$$P_{pq} + jQ_{pq} = V_p \left[ \frac{1}{a} \left( \frac{V_p}{a} - V_q \right) y_{pq} \right]^*$$

$$= \frac{1}{a^2} V_p^2 y_{pq}^* - \frac{1}{a} V_p V_q^* y_{pq}^* \qquad (9).$$

Similarly the power flow from bus q to p is

$$P_{qp} + jQ_{qp} = V_q\left[\left(V_q - \frac{V_p}{a}\right)y_{pq}\right]^*$$

$$= V_q^2 y_{pq}^* - \frac{1}{a} V_q V_p^* y_{pq}^* \qquad (10)$$

The voltage at the beginning of any lateral can be computed as a function of the end voltage if the currents injected into each sub-laterals are given. In a system with L levels, the level L laterals have no sub-laterals and therefore be computed first. After all level L laterals have been computed the current injected into sub-laterals of each level L-1 laterals can be computed. Next the level L-2 laterals are computed and so on until the main feeder has been computed.

# THE LINKNET STRUCTURE

In the development of any algorithms which deals with a network, provision should be made on how the network information should be stored in computer memory. The decision is particularly important for the analysis of the large sparse networks which occur in power system studies. For a wide variety of network problems there are a number of common features which are desirable in a storage structure. They are:

a) Use of small amount of computer memory.
b) Processing of the network information should be facilitated e.g. the branches and nodes connected to any given node should be easily scanned.
c) The structures should easily reflect network changes eg. The addition or removal of branches.
d) The structure should be basically simple and easy to program.

The Linknet structure incorporates each of the desirable features. The usefulness of the Linknet structure has been demonstrated by its application in the programming of a wide variety of network algorithm. It has proved to be a valuable tool for the programming of the power distribution system network in this dissertation. The structure which is directly applicable to any high level language like Fortran or $C^{++}$ is described below:

Typically in a power system network the nodes are numbered manually but the branch numbering is left to the computer. The properties of the network are divided into three types viz. node properties, branch properties and topological properties. The node and branch properties are stored in standard fashion. For each node or branch property a one dimensional array is allocated and each position in the array is identified with the node or branch having the corresponding number.

The topological properties of the network are represented by specifying the connection between the nodes and the branches. Firstly we assume that the ends of each branch are numbered as follows:

Ends of branch 1 are numbered 1 and 2 , ends of branch 2 are numbered 3 and 4 etc. Thus the branch end numbers may be derived from a branch number as

$$\begin{aligned} END &= f(BRANCH) \\ &= 2 \times BRANCH-1 \end{aligned}$$

and

$$\begin{aligned} END &= g(BRANCH) \\ &= 2 \times BRANCH \end{aligned}$$

Conversely a branch number may be derived from either of its end number using

$$\begin{aligned} BRANCH &= f(END) \\ &= (END+1)/2 \end{aligned}$$

In this relationship the integer round off is used to obtain the two to one mapping between branch ends and branches. The topology of the network can now be defined by constructing a linked list of the branch ends which are connected to each nodes.

For each node we define a pointer LIST[NODE] which is the first branch end on the list from node. For each branch we define a pointer NEXT[END], the next branch end on the list after END. The last branch end on the list for each node is indicated when NEXT[END] = 0. The LIST[NODE] and NEXT[END] pointers are sufficient to define the network topology uniquely. They allow the branches connected to any node to be directly obtained using the procedure

```
END        = LIST[NODE]
BRANCH     = f[END] = (END +1)/2
END = NEXT[END] until END = 0
```

In the network computation it is often necessary to obtain the nodes which are connected to given node. This operation is facilitated by defining an additional pointer for each branch end, the FAR[END] which is the node at the far or opposite end of the branch. The nodes connected to any given node can now be obtained using the procedure

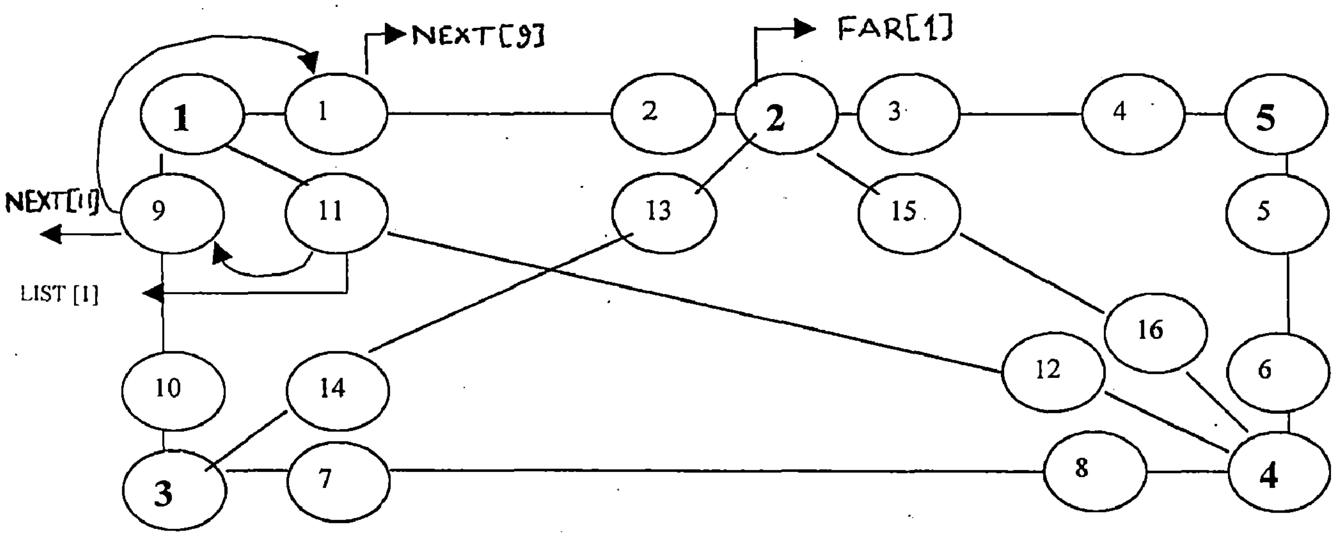
END = LIST [NODEA]

NODEB = FAR [END]

END = NEXT [END] until END = 0

The successive values of NODEB will be the nodes which are connected to NODEA.

The three pointers as explained above are illustrated with the help of Fig. 3.1.




| NODE | LIST[NODE] | END | NEXT [END] | FAR [END] |
|---|---|---|---|---|
| 1 | 11 | 1 | 0 | 2 |
| 2 | 15 | 2 | 0 | 1 |
| 3 | 14 | 3 | 2 | 5 |
| 4 | 16 | 4 | 0 | 2 |
| 5 | 5 | 5 | 4 | 4 |
| | | 6 | 0 | 5 |
| | | 7 | 0 | 4 |
| | | 8 | 6 | 3 |
| | | 9 | 1 | 3 |
| | | 10 | 7 | 1 |
| | | 11 | 9 | 4 |
| | | 12 | 8 | 1 |
| | | 13 | 3 | 2 |
| | | 14 | 10 | 3 |
| | | 15 | 13 | 4 |
| | | 16 | 12 | 2 |

**Fig. 3.1 LIST[NODE], NEXT[END], and FAR[END] Pointers in "Linknet" Structure**

When we want to obtain nodes at the ends of a given branch, following procedure can be adopted.

| | | | |
|---|---|---|---|
| ENDA | = | f[BRANCH] | = 2 x BRANCH –1 |
| ENDB | = | g[BRANCH] | = 2 X BRANCH |
| NODEA | = | FAR[ENDB] | |
| NODEB | = | FAR[ENDA] | |

The pointers LIST[NODE], NEXT[END] and FAR[END] form the framework of the "linknet" structure. These pointers enable various network scanning operations to be performed very easily.

# METHODOLOGY AND SOFTWARE DEVELOPMENT

## 4.1 INTRODUCTION

The problem of optimal line separation can be assessed in two ways.

(i) To find the optimal location of separation line for new distribution system.

(ii) To verify the correctness of the already fixed separation line of the existing system.

In first type of problem, various supply points in the system are known beforehand. The load flow solution of the system with multiple supply is then determined by Gauss Seidel method and the various options of separation line are found. When any one of the probable option is removed from the system, the network turns into the two or more radial network. The load flow solution of each individual radial system for each option is then found using Z loop algorithm to find the option giving rise to minimum system losses.

In second type of problem an existing system already having provision of separation line is considered and its optimality is checked. Among these two methods, the later is carried out in the present study. The methodology for this has been explained in section 4.3.

## 4.2 LOAD FLOW SOLUTION

The load flow solutions provide power flows and voltages for a specified power system. This information is essential for the continuous evaluation of the current performance of a power system and for analyzing the effectiveness of alternative plans for system expansion to meet the increased load demand.

The load flow problem consists of the calculation of power flows and voltages of a network for specified terminal or bus condition. Associated with each bus are four quantities: real and reactive powers, the voltage magnitude and the phase angle. Three types of buses are represented in the load flow calculation and at a bus two of the four quantities are specified. It is necessary to select one bus, called slack bus, to provide the additional real and reactive power to supply the transmission losses, since, these are unknown until the final solution is obtained. At this bus, the voltage magnitudes and phase angles are specified. The remaining buses of the system are designated either as voltage controlled buses or load buses. The real power and voltage magnitudes are specified at voltage controlled bus. The real and reactive powers are specified at a load bus.

Load flow calculation studies started with Ward and Hale method in 1956. An early approach was the Gauss Seidel iterative method using the nodal admittance matrix. Currently the Newton Raphson methods by using a nodal admittance matrix has gained wide spread popularity because of its quadratic convergence characteristic. However it has limitation for small core computer application. Extensive memory requirements in large power system calculation motivated the exploitation of sparsity with skilful programming. The problem of fast computation could be said to have been almost solved by the development of fast de-coupled load flow. Thus electronic computers are widely used in the solution of the problems of a large complex electrical distribution network.

The choice of a solution method for a particular application is frequently difficult. It requires a careful analysis of the comparative merits and demerits of the methods available in respect of storage, computation speed and convergence characteristics. The difficulties arise from the fact that no method processes all the desirable features of the others.

Two types of distribution network are dealt for the solution of optimal line separation problem. Non radial network and radial network. Out of many methods specified above the Gauss-Seidel iterative method is selected for the solution of non radial network and the Z-loop impedance method is selected for the solution of radial network . So these two methods are discussed below in details

### 4.2.1 Gaess Seidel Iterative Method [11]

The solution of the load flow problem is initiated by assuming voltages for all the buses except the slack bus where the voltage is specified and remains fixed. The bus currents are given by the equation:

$$I_i = \frac{P_i - jQ_i}{E_i^*} \tag{1}$$

i = 1,2,.....,n

$i \neq s$

where n is the number of buses in the network, $P_i$ and $Q_i$ are the real and reactive power at the bus i with voltage $E_i$. The performance of the network can be obtained from the equation

$$I_{bus} = Y_{bus}.E_{bus}$$

Selecting the ground bus as the reference bus, a set of n-1 simultaneous equations can be written in the form

$$E_i Y_{ii} = I_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} Y_{ij} E_j \tag{2}$$

i = 1,2,.....n

$i \neq s$

when $Y_{ii}$ is the driving point admittance of bus i and $Y_{ij}$ is the off diagonal admittance between bus i and j and s is the slack bus.

The bus currents calculated from equation (1), the slack bus voltages and the estimated bus voltages are substituted into equation (2) to obtain a new set of bus voltages.

These new voltages are used in equation (1) to re-calculate bus currents for a subsequent solution of equation. The process is continued until changes in all bus voltages are negligible. After the voltage solutions are obtained, the power at the slack bus and line flows can be calculated.

From equation (1) and (2), we can have

$$E_i = \frac{1}{Y_{ii}}\left(\frac{P_i - jQ_i}{(E_i *)} - \sum_{\substack{j=1 \\ j\neq i}}^{n} Y_{ij}E_j\right), i = 1,2,.....n, i \neq s \qquad (3)$$

In general,

$$E_i^{(k+1)} = \frac{1}{Y_{ii}}\left(\frac{P_i - jQ_i}{\left(E_i^{(k)}\right)^*} - \sum_{j=1}^{i-1} Y_{ij}E_j^{(k+1)} - \sum_{j=i+1}^{n} Y_{ij}E_j^{(k)}\right) \qquad (5)$$

Where k is the iteration count

As shown by equation (5), it is clear that in Gauss-Seidel method the new calculated voltage $E_i^{(k+1)}$ immediately replaces $E_i^{(k)}$ and is used in the solution of the subsequent equations.

**<u>Line flows</u>**

After the iterative solution of bus voltage is completed, line flows can be calculated. The current at bus i in the line connecting bus i to j is

$$i_{ij} = (E_i - E_j)\, y_{ij} \qquad (6)$$

where line charging admittance is not considered.

The power flow, real and reactive, from bus i to j is given by

$$P_{ij} - jQ_{ij} = E_i^* \, i_{ij}$$

or,

$$P_{ij} - jQ_{ij} = E_i^* \, (E_i - E_j) \, y_{ij} \qquad (7)$$

Where at bus i, the real power flow from bus i to j is $P_{ij}$ and the reactive is $Q_{ij}$. Similarly, at bus j the power flow from j to i is

$$P_{ji} - jQ_{ji} = E_j^* \, (E_j - E_i) \, y_{ij}$$

### 4.2.2 Z- Loop Method

The Z-loop method [ 3 ] is a new calculation method of the load flow algorithm for a radial distribution system. Figure 4.1 is the generalized configuration of a radial system. Obviously from the figure 4.1 The following equations can be formulated for a radial system of n buses by using circuit theory.

The complex power of the load branch is defined as

$$S_{bi} = V_{bi}^* \, . I_{bi} \ , i = 1,2, \ldots\ldots, n \qquad (1)$$

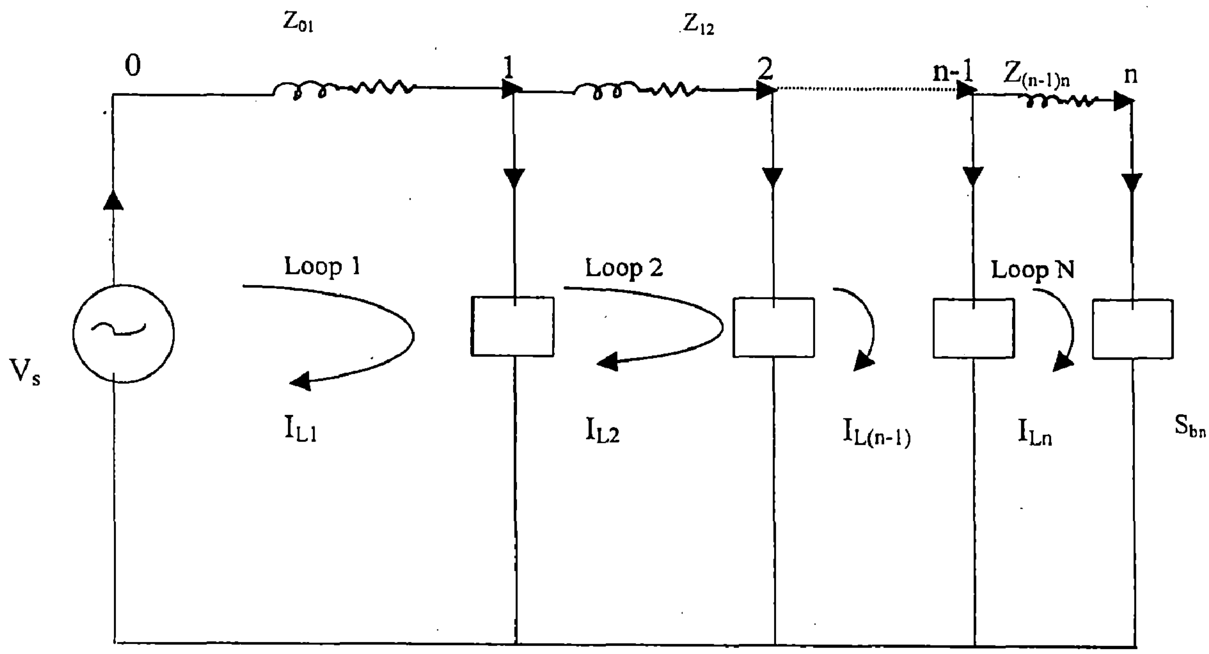



**Fig. 4.1 Generalized configuration of radial distribution system**

Where,

$V_{bi}$ & $I_{bi}$ are bus voltage and bus currents respectively.

The voltage at node 1 is

$$V_{b1} = V_0 - E_1,$$

where $V_0$ is the voltage of the slack bus and $E_1$ is the net voltage source of loop 1

Similarly voltage at node 2 is

$$V_{b2} = V_1 - E_2 \text{ and so on.}$$

In general,

$$V_{bi} = V_{i-1} - E_i \qquad i = 1,2,3,\ldots\ldots n \qquad (2)$$

Where $E_i$ is the net voltage source of loop i. Since $E_i$ is also related to the impedance of the line and the current flow through it, a set of equation can be derived as follows:

Loop 1 : $E_1 = Z_1 x\ I_{L1}$

Loop 2 : $E_2 = Z_2\ x\ I_{L2}$

:

:

Loop n : $E_n = Z_n\ x\ I_{Ln}$ (3)

Where $I_{Li}$ is the feeder section current. In matrix form, we can write

$$|E| = |Z\text{-loop}|\ x\ |I_L| \qquad (4)$$

where Z-loop is the loop impedance matrix of the network and remains constant during calculations, E is the net voltage source around the loop and $I_L$ is the loop current matrix.

By applying Kirchoff's current law (KCL) to the nodes

$$I_{Li} = \begin{cases} I_{bi} + I_{l(i+1)}, 1 < i \le n-1 \\ I_{bn} \qquad , i = n \end{cases} \qquad (5)$$

This load flow solution method can easily be implemented on a microcomputer by using the basic Gauss iteration method. Because of such a simple algorithm, the use of microcomputers for distribution system can be achieved. The loop currents found by equation (5) are used to calculate the voltage drop in the line using equation (4) which inturn is substituted in equation (2) to calculate the bus voltages.

### 4.2.3 Distribution Line Losses

After the solution of all buses have been obtained, the line flows are calculated as follows:

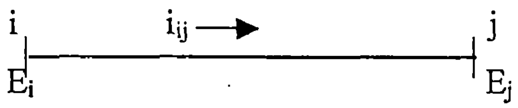



**Fig: 4.2 Distribution Line Loss Calculation**

Neglecting the line charging admittance, the current flowing from bus i to bus j is

$i_{ij} \quad = \quad (E_i - E_j)\, y_{ij}$

where $y_{ij}$ is the admittance of line connected between the buses i and j with voltages $E_i$ and $E_j$ respectively.

The power flow from i to j is given by

$P_{ij} - jQ_{ij} = \quad E_i^* (E_i - E_j) y_{ij}.$

Let

$E_i = e_i + jf_i \qquad E_j = e_j + jf_j$ and $y_{ij} = g_{ij} + jb_{ij}.$

then

$P_{ij} - jQ_{ij} = (e_i - jf_i)\ (e_i + jf_i - e_j - jf_j)\ (g_{ij} + jb_{ij})$

Equating real and imaginary part and simplifying, we get

Real power flow from bus i to bus j is

$$P_{ij} = g_{ij} \ (e_i^2 + f_i^2 - e_ie_j - f_if_j - b_{ij} \ (e_jf_i - e_if_j) \qquad (1)$$

Similarly, the real power flow from bus j to bus **i** is given by

$$P_{ji} \quad = g_{ij} \ (e_j^2 + f_j^2 - e_ie_j - f_if_j) - b_{ij} \ (e_if_j - e_jf_i) \qquad (2)$$

The active power loss in the line i - j is the algebraic sum of the real power flows of equation (1) and (2)

Hence real power loss

$$P_{los\ ij} \quad = p_{ij} + P_{ji}$$

### 4.2.4 Comparison of Solution Methods

The storage requirement of the Z - loop method is almost the same as that of the Gauss Seidel method. The Z - loop method requires extra memory to store the branch currents and loop currents but does not require impedance conversion into admittance and also does not require storage of admittance data which is required by the Gauss Seidel method. More over this method is easy to program as is the Gauss Seidel method . The main advantage of Z - loop method over the Gauss Sedel method is that it is more efficient and converges very fast compared to G.S.Method. Although the convergence pattern of this method is not so fast as that of Newton Raphson method, its total computational time is lesser than that of NR method.

## 4.3 METHODOLOGY

The methodology adopted in the present study is as follows:

(i) Already existing separation lines are reconnected and the loops in the system are formed. If there are n separation lines in the system, there will also be n loops formed.

(ii) The load flow solution of this single supply non radial system (with loops) is found out using usual Gauss Seidel algorithm.

(iii) The various pairs of lines which meet at a node where power is supplied from two or more adjacent nodes are then determined. If there are n loops, consequently, n pairs will be resulted. One line out of each pair is selected from each pair as the separation line. Because if any line other than these lines is selected, the load is not shared proportionately from two sides of the lines causing un-uniform voltage distribution and giving more $I^2R$ losses as compared to that of the line from the pair.

(iv) All the probable combinations (options) of separation lines are made. There will be $2^n$ combinations in total for n pairs of lines.

(v) For each option, the resulting radial network is renumbered, for applying Z loop algorithm of load flow solution.

(vi) The solution of the radial network for each options are found and system losses are determined.

(vii) The solutions are determined for each hour of day as per the daily load curve of the system and cumulative energy losses are found.

(viii) The option resulting in lowest daily energy loss is selected.

(ix) The energy loss of selected option is compared with the energy loss of the existing option and saving is calculated.

## 4.4 ALGORITHMS FOR SOFTWARE DEVELOPMENT

### 4.4.1 Algorithm for linknet program for storing network connectivity information:

1. Read beginning node NODEA and ending node NODEB of all branches.
2. Initialize LIST [NODEA] = 0, LIST [NODEB] = 0.
3. Set branch count = 1
4. Find ENDA and ENDB using

   ENDA = f(BRANCH) = 2 X BRANCH –1

   ENDB = g(BRANCH) = 2 X BRANCH

5. Find NEXT[ENDA], LIST[NODEA] and FAR[ENDA] pointers using following procedure

| | | |
|---|---|---|
| NEXT [ENDA] | = | LIST [NODEA] |
| LIST [NODEA] | = | ENDA |
| FAR [ENDA] | = | NODEB |

6. Find NEXT[ENDB], LIST [NODEB], FAR[ENDB] pointers using following procedure:

| | | |
|---|---|---|
| NEXT [ENDB] | = | LIST [NODEB] |
| LIST [NODEB] | = | ENDB |
| FAR [ENDB] | = | NODEA |

7. Increase branch count by 1.
8. If branch count is less or equal to number of branches go to 4, otherwise linnknet structure is completed.

### 4.4.2 Algorithms for Gauss Sediel Method of load flow solution for radial/non radial type of network:

1. Read system data: Bus powers, flat start voltage of each bus, slack bus, line impedance, beginning and ending nodes of lines.
2. Call Linknet program to store the connectivity information of the network
3. Calculate the diagonal elements of Y Bus matrix $Y_{ii}$ for each bus by summing up the line admittance of all the lines connected to that particular bus.
4. Set iteration count k = 0
5. Set maximum voltage $\Delta V_{max} = 0$ and bus count i=1
6. Test for slack bus. If it is slack bus go to 11, otherwise calculate the bus parameters $A_{ii}$ and line parameters $B_{ij}$:

$$A_{ii} = (P_i - jQ_i) / Y_{ii}$$

and

$$B_{ij} = Y_{ij}/Y_{ii} = - y_{ij} / Y_{ii}$$

Where $y_{ij}$ is the line admittance of line connected between the bus i & j

7. Calculate the voltage of $i^{th}$ bus from the relation

$$V_i^{(K+1)+} = A_{ij}/(V_i^{(K)})^* - \sum_{j=1}^{i-1} B_{ij} x V_j^{(K+1)} - \sum_{j-1+1}^{n} B_{ij} x V_j^{(K)}$$

8. Find the voltage difference

$$\Delta V = V_i^{(K+1)} - V_i^{(K)}$$

9. Test for Voltage difference $\Delta V$ . If it is greater than $\Delta V_{max}$ replace $\Delta V_{max}$ by $\Delta V$, otherwise not.

10. Replace $V_i^{(K)}$ by $V_i^{(K+1)}$

11. Advance bus count by 1 i.e. i = i+1

12. Test for calculation of voltages of all the buses. If i is less or equal to n, the number of buses go to 6, otherwise test for convergence as explained in step 13 .

13. If $\Delta V$ is greater than tolerance advance iteration count by 1 i.e K = K+1 and go to 5. Otherwise, calculate line flows, line losses and power at slack bus.

### 4.4.3 Algorithm for renumbering of the network after the lines are separated:

1 Set DIST[I]=9999; I=1,2,3,........,n, where n is the number of buses.
Initialize array DIST[1]=0;FORWARD[1]=0
Initialize I=1; IND=0; END=LIST[1]; MDISTANCE=1.

2 Find the branch connected corresponding to the end using

BRANCH=(END+1)/2

3. If BRANCH is same as the separation line, go to 5 ,otherwise find the node connected to the corresponding branch by using

J=FAR[END]

4. Replace DIST[J] by MDISTANCE i.e DIST[J]=MDISTANCE.
increase I by 1 , set FORWARD[I]=J and increase IND by 1.

Set FLAG1[IND]=J.

5. Update END i.e END=NEXT[END].

6. If END is not equal to zero, go to 2, otherwise

Set K=1, IND2=0 and increase MDISTANCE by 1.

7 Find the new END using END=LIST[FLAG[K]]

8 Find the corresponding branch using BRANCH=(END+1)/2.

9. If BRANCH falls to be any of the separation line go to 12, otherwise find the node corresponding to the BRANCH using J=FAR[END].

10. Compare DIST[J] with 9999. If they are equal increase IND2 by 1, otherwise go to 12.

11. Increase I by 1, set FORWARD[I]=J, DIST[J]=MDISTANCE and FLAG2[IND2]=J.

12. Update END by using END=NEXT[END].
If END=0 increase K by 1, otherwise go to 8.

13. If K <= IND go to 7, otherwise set IND=IND2 and M=1.

14 Set FLAG1[M]]=FLAG2[M].

15 INCREASE M BY 1.
If M <=IND2, go to 14, otherwise check IND2. If IND2 =0, stop else go to 7.

### 4.4.4 Algorithm for Z-loop method for finding solution for radial networks:

1. Read the number of buses, bus powers, flat start voltage of buses, slack bus, line impedance, beginning and ending node of all lines voltages tolerances and acceleration factor..
2. Call the Linknet program to store the information about network topology.
3. Set iteration count k = 0
4. Set the max voltage difference $\Delta V_{max} = 0$
5. Find the bus currents of each bus by using

$$I_{bi} = S_{bi}^{*} / V_{bi}^{*}, \; i = 1,2,.....n, \; i \neq s$$

6. Apply KCL and back substitution method to find the line currents

$$I_{Li} = I_{bi} + I_{L(i+1)} \quad 1 < i < n-1$$
$$= I_{bn} \quad i=n$$

7. Calculate the new voltages of the buses except slack bus.

$$V_i^{(K+1)} = V_{i-1}^{(K)} - Z_{(i-1)i}\, I_{Li} \;, \quad i = 2,3.......n$$

8. Find the voltage difference

$$\Delta V = V_i^{(K+1)} - V_i^{(K)} \;, i = 2,3,.....n$$

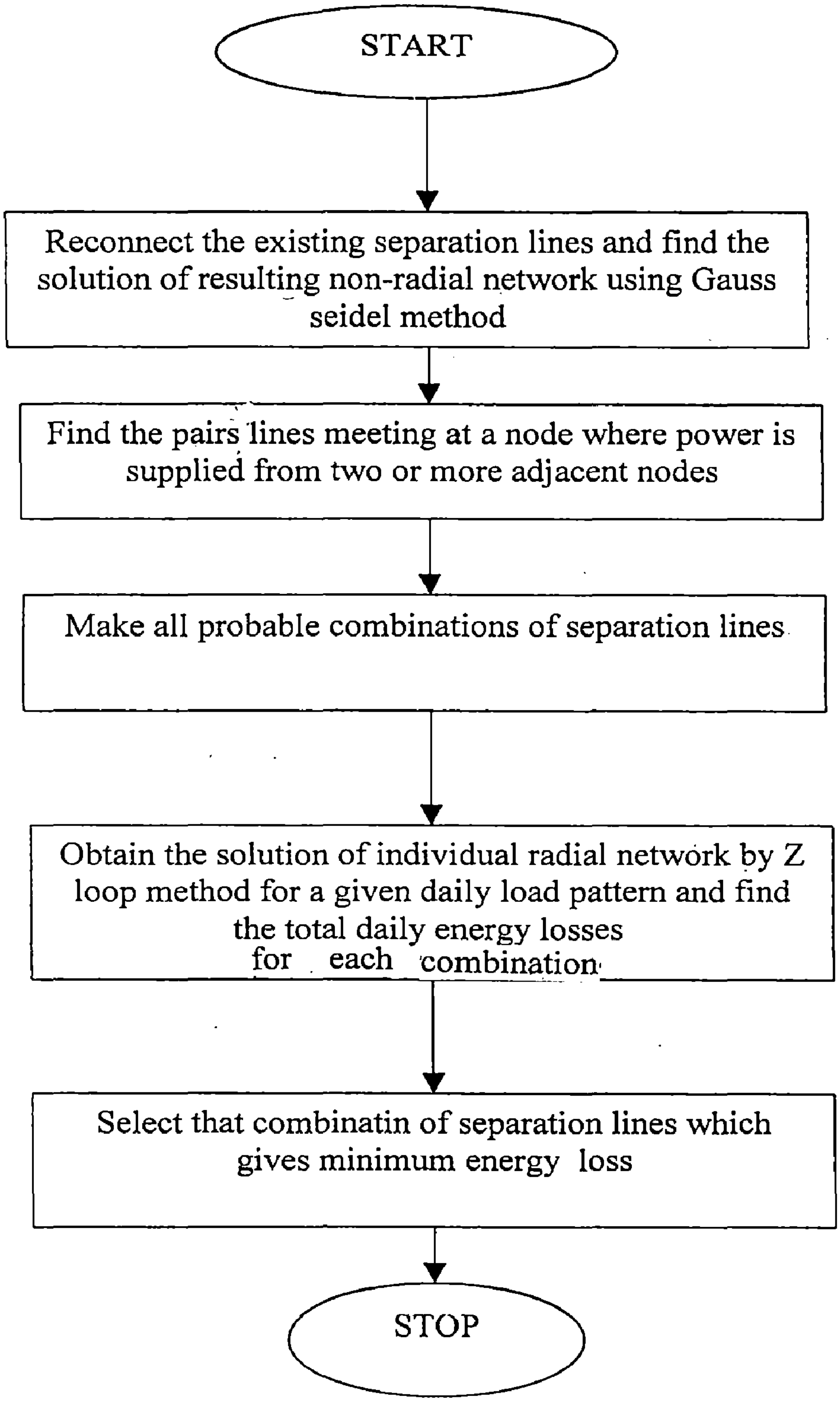
if $\Delta V$ is greater than $\Delta V_{max}$, replace $\Delta V_{max}$ by $\Delta V$, otherwise not.

9. Check for convergence.

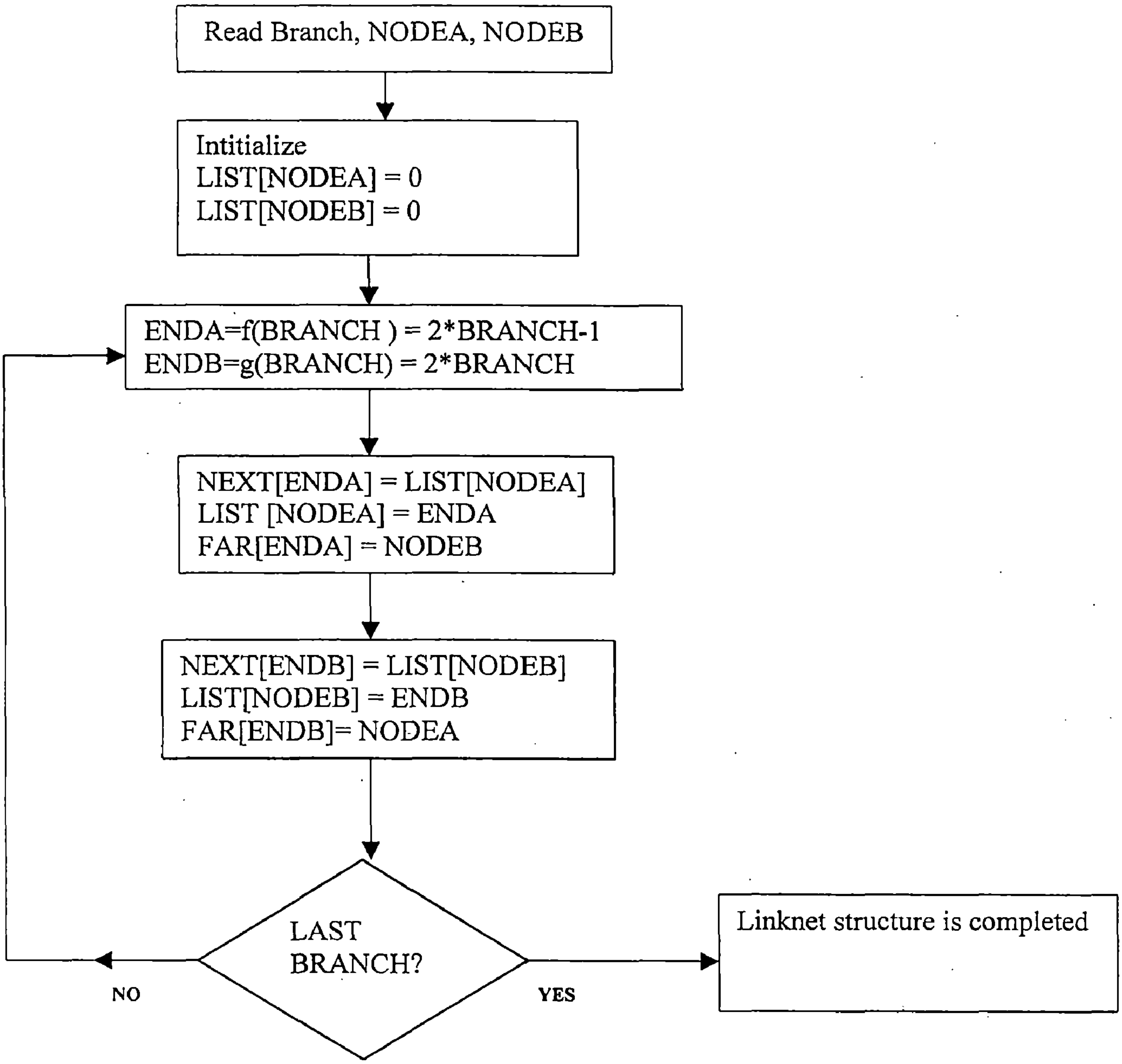
If $\Delta V_{max}$ greater than tolerance replace $V_i^{(K)}$ by $V_i^{(k+1)}$, increase iteration count $i = i+1$ and go to 5. If $\Delta V_{max}$ is less or equal to tolerance, stop iteration and calculate line flows, line losses and slack bus power.

## 4.5 FLOW CHARTS

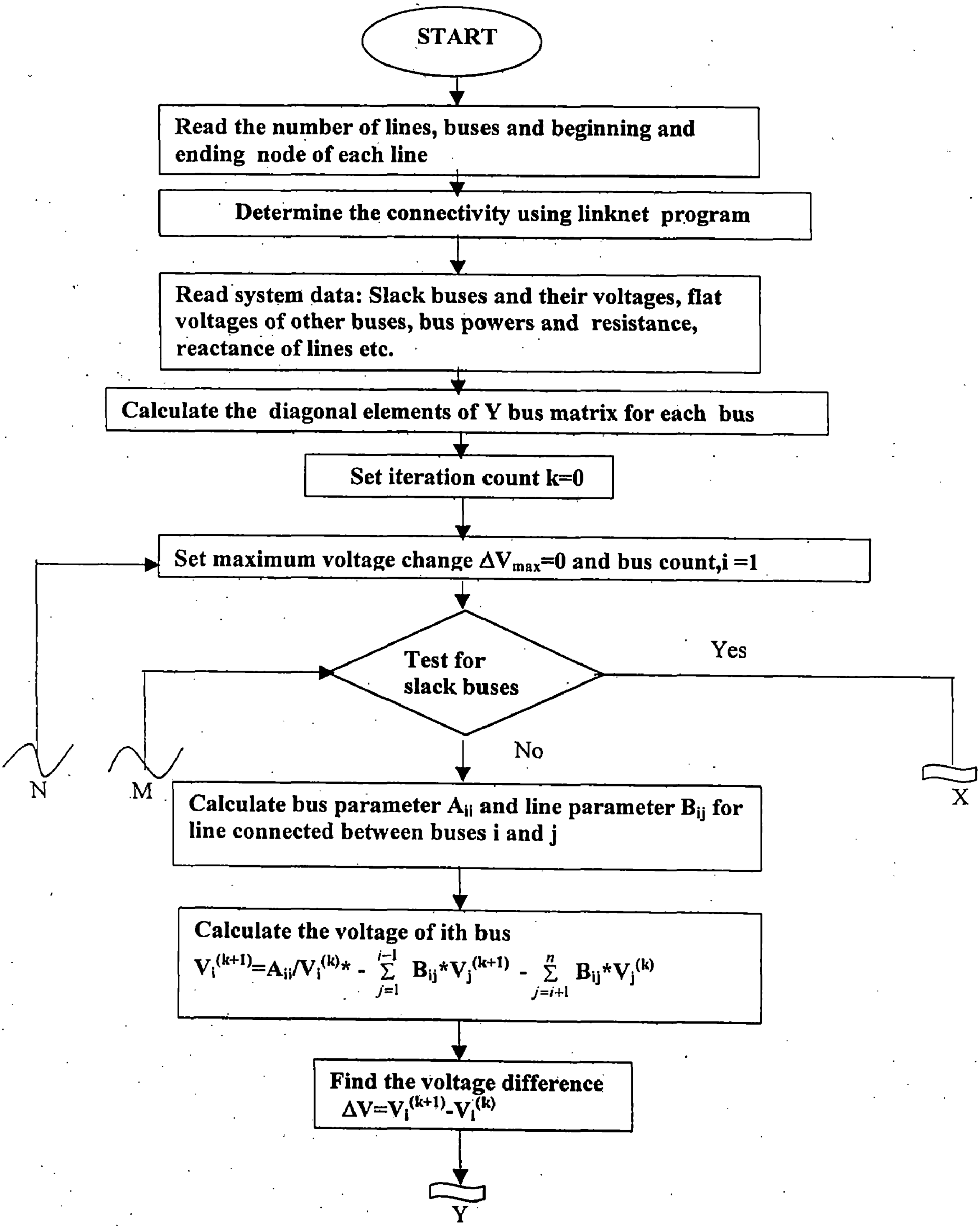
### 4.5.1 Flow Chart of Optimal Line Separation in Power Distribution System

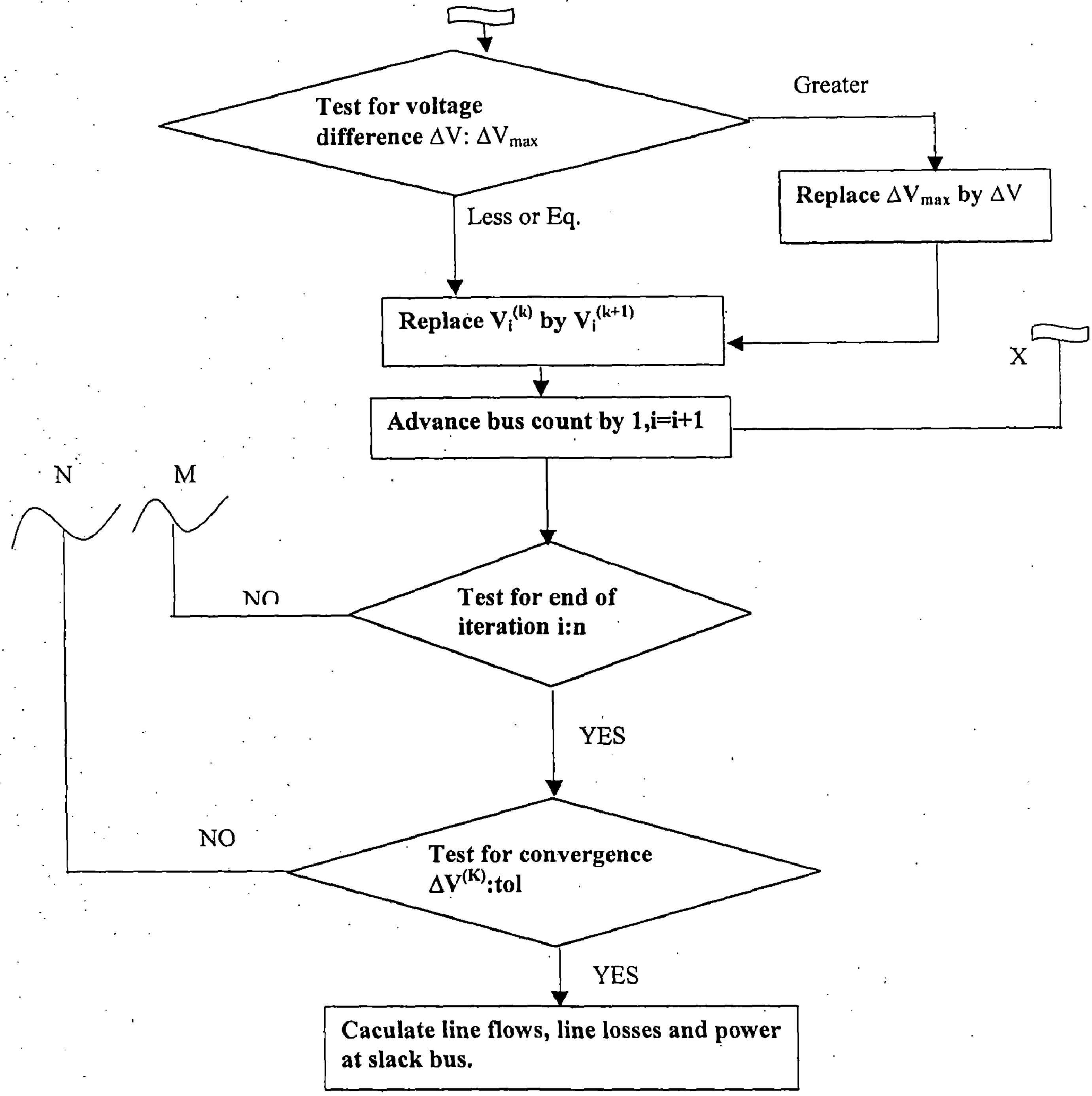


### 4.5.2 Flow Chart of Linknet Structure

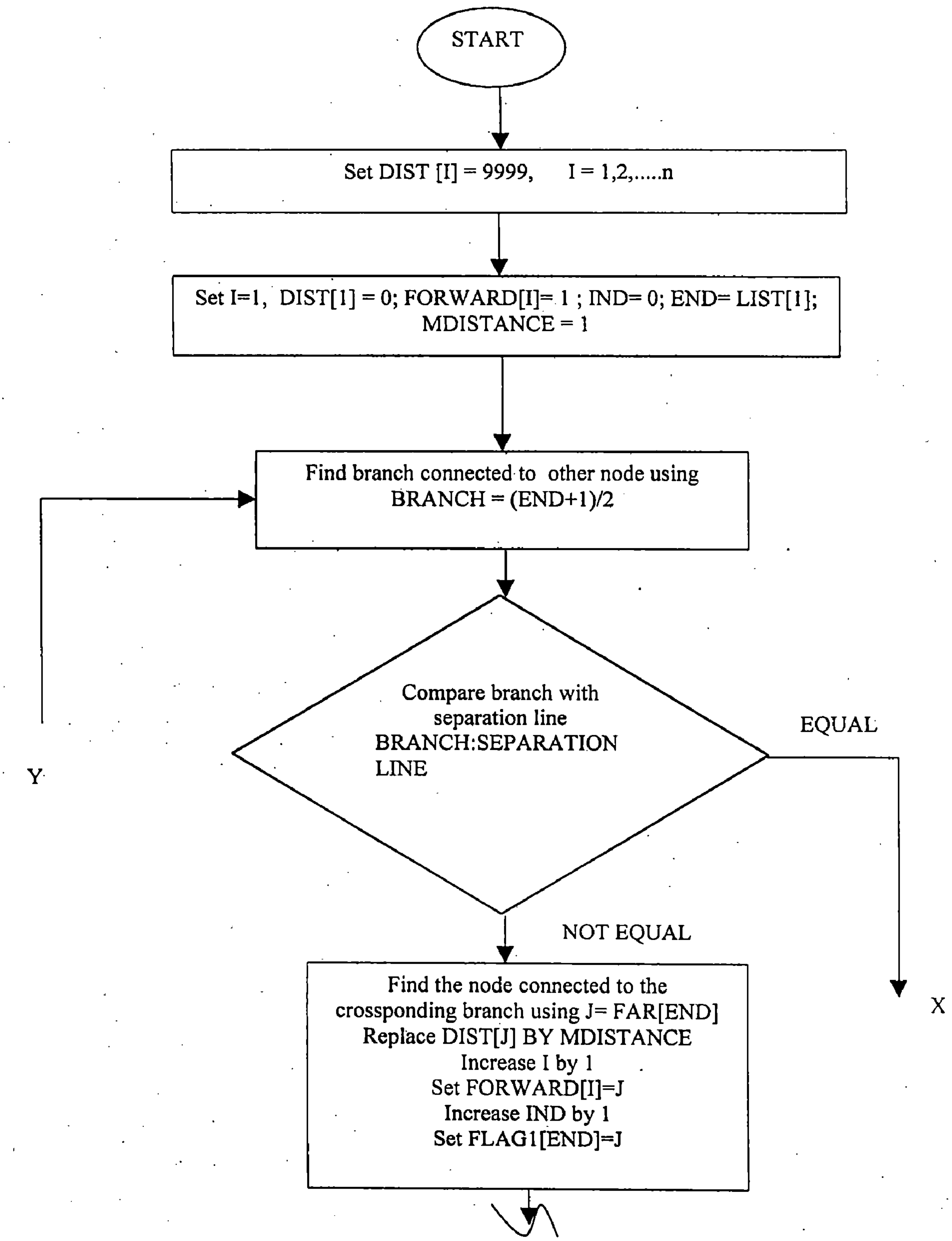


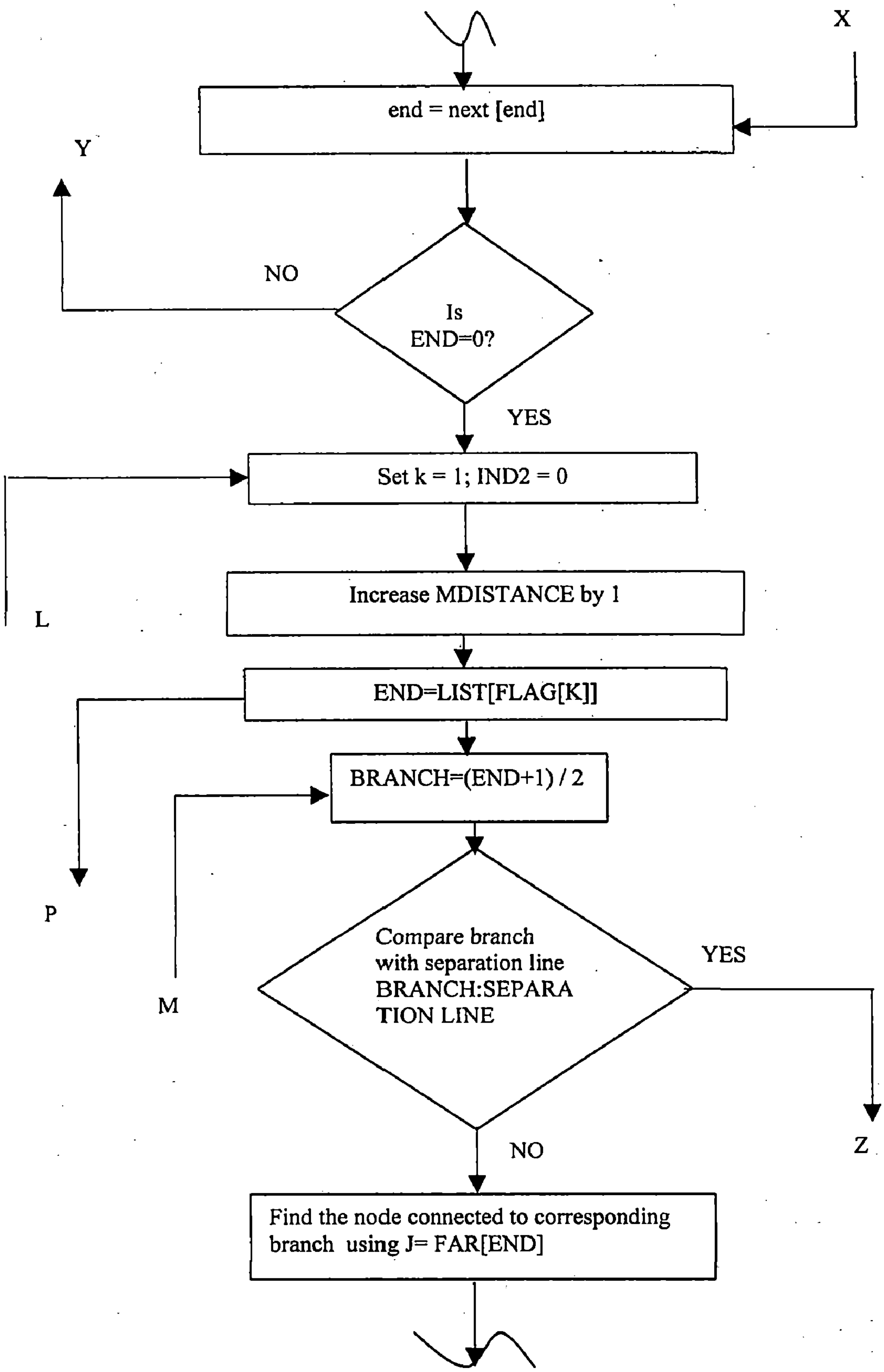
### 4.5.3 Flow chart of the gauss seidel method

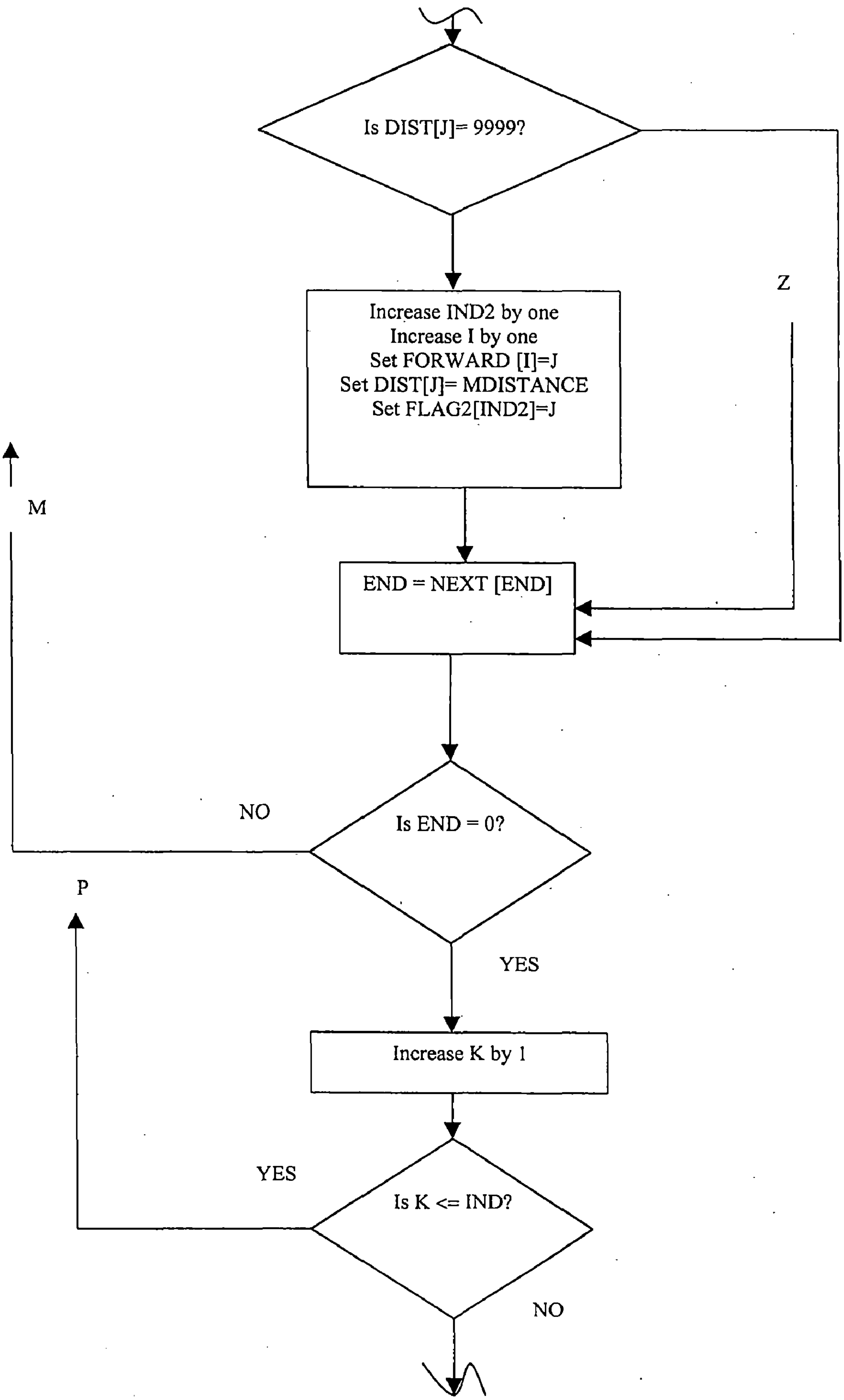


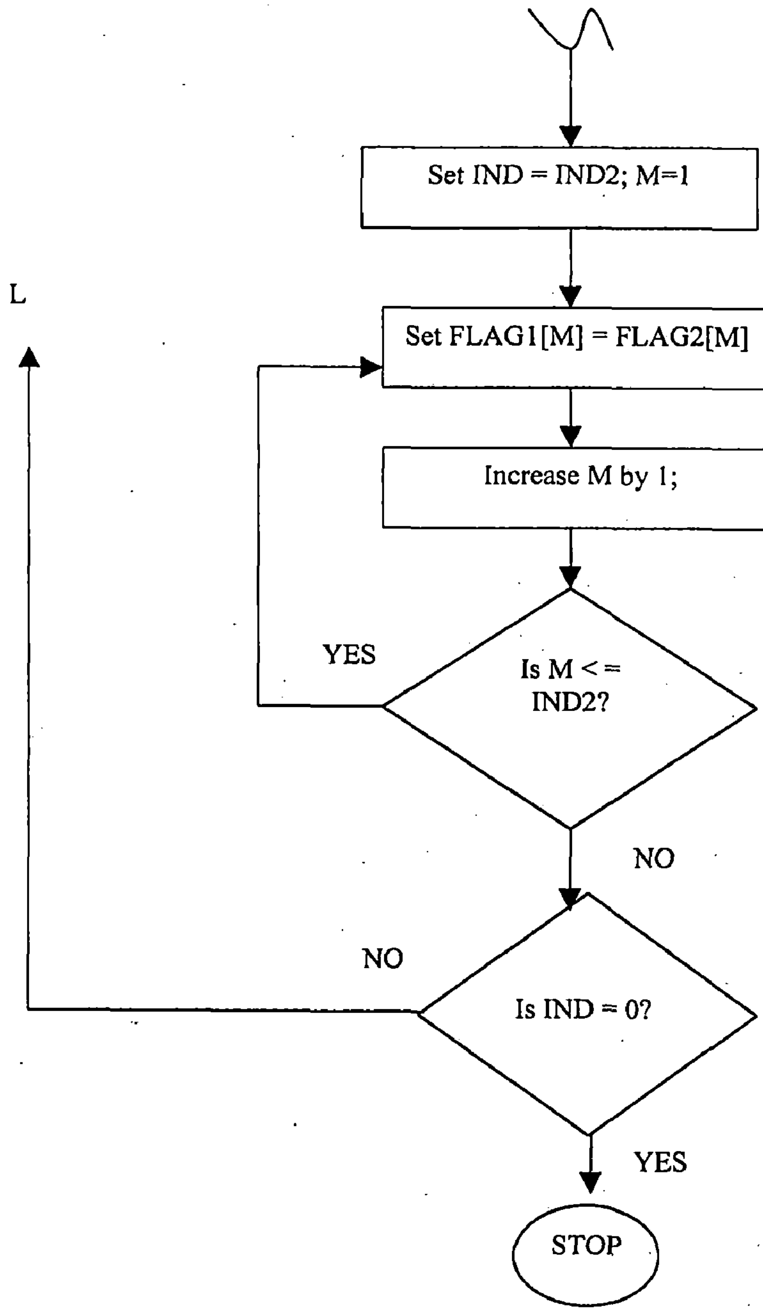

Test for voltage difference ΔV: ΔVmax
Greater
Replace ΔVmax by ΔV
Less or Eq.
Replace Vi(k) by Vi(k+1)
X
Advance bus count by 1,i=i+1
N
M
NO
Test for end of iteration i:n
YES
NO
Test for convergence ΔV(K):tol
YES
Caculate line flows, line losses and power at slack bus.

### 4.5.4 Flow Chart for the Re-Numbering of the Network after Separating the Lines



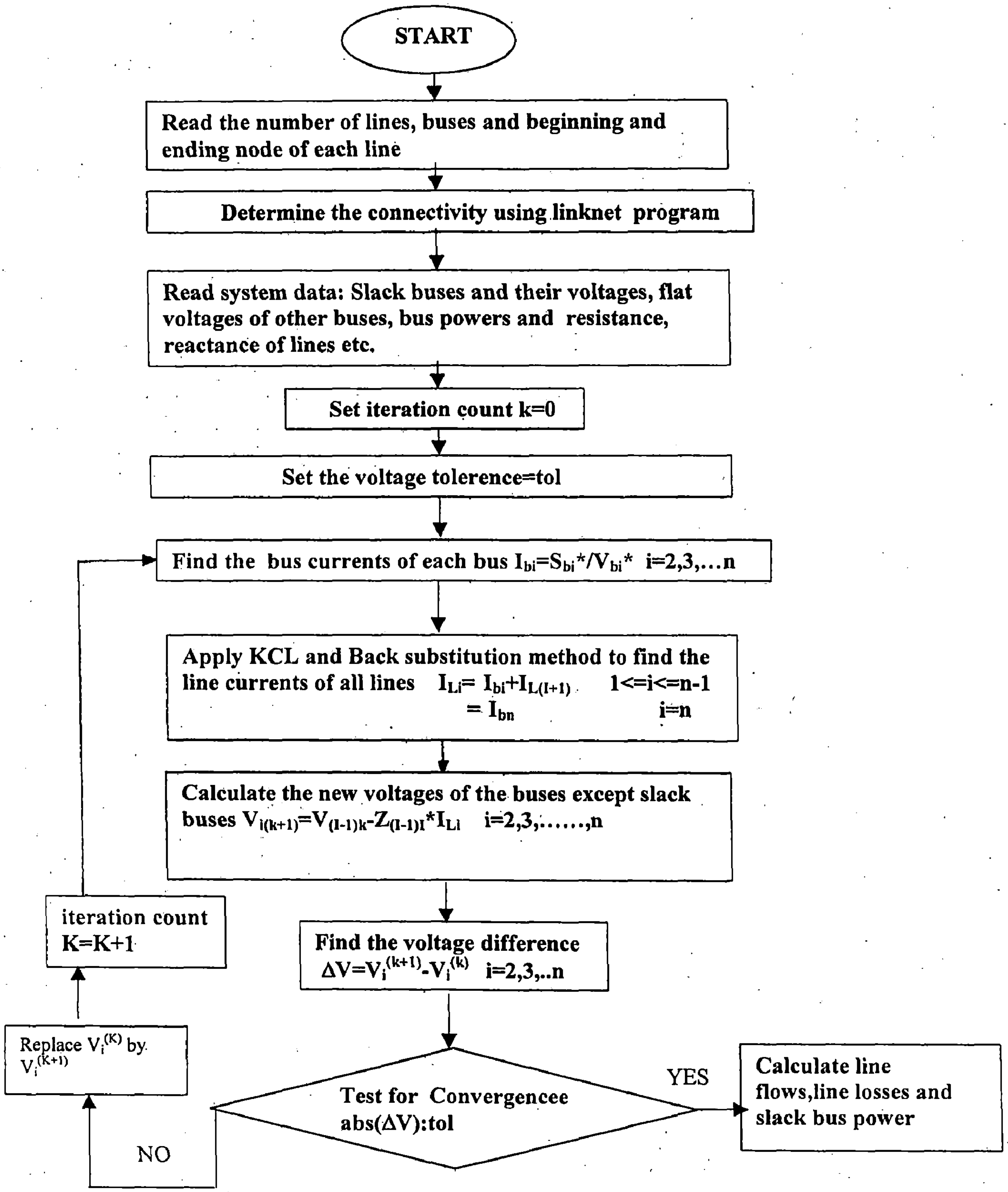

X
end = next [end]
Y
NO
Is
END=0?
YES
Set k = 1; IND2 = 0
Increase MDISTANCE by 1
L
END=LIST[FLAG[K]]
BRANCH=(END+1) / 2
P
Compare branch
with separation line
BRANCH:SEPARA
TION LINE
YES
M
NO
Z
Find the node connected to corresponding
branch using J= FAR[END]


Is DIST[J]= 9999?
Increase IND2 by one
Increase I by one
Set FORWARD [I]=J
Set DIST[J]= MDISTANCE
Set FLAG2[IND2]=J
Z
M
END = NEXT [END]
NO
Is END = 0?
P
YES
Increase K by 1
YES
Is K <= IND?
NO



Set IND = IND2; M=1
L
Set FLAG1[M] = FLAG2[M]
Increase M by 1;
YES
Is M <=
IND2?
NO
NO
Is IND = 0?
YES
STOP

4.5.5 FLOW CHART OF Z-LOOP METHOD



## RESULTS AND DISCUSSIONS

To check the validity of the methodology and developed software, two real 38 and 114 bus 33 KV power distribution networks of systems of Bhopal city have been considered. The network structure and load curve data for Bhopal city are taken as obtained from Madhya Pradesh State Electricity Board (MPSEB).

### 5.1 38 - BUS SYSTEM

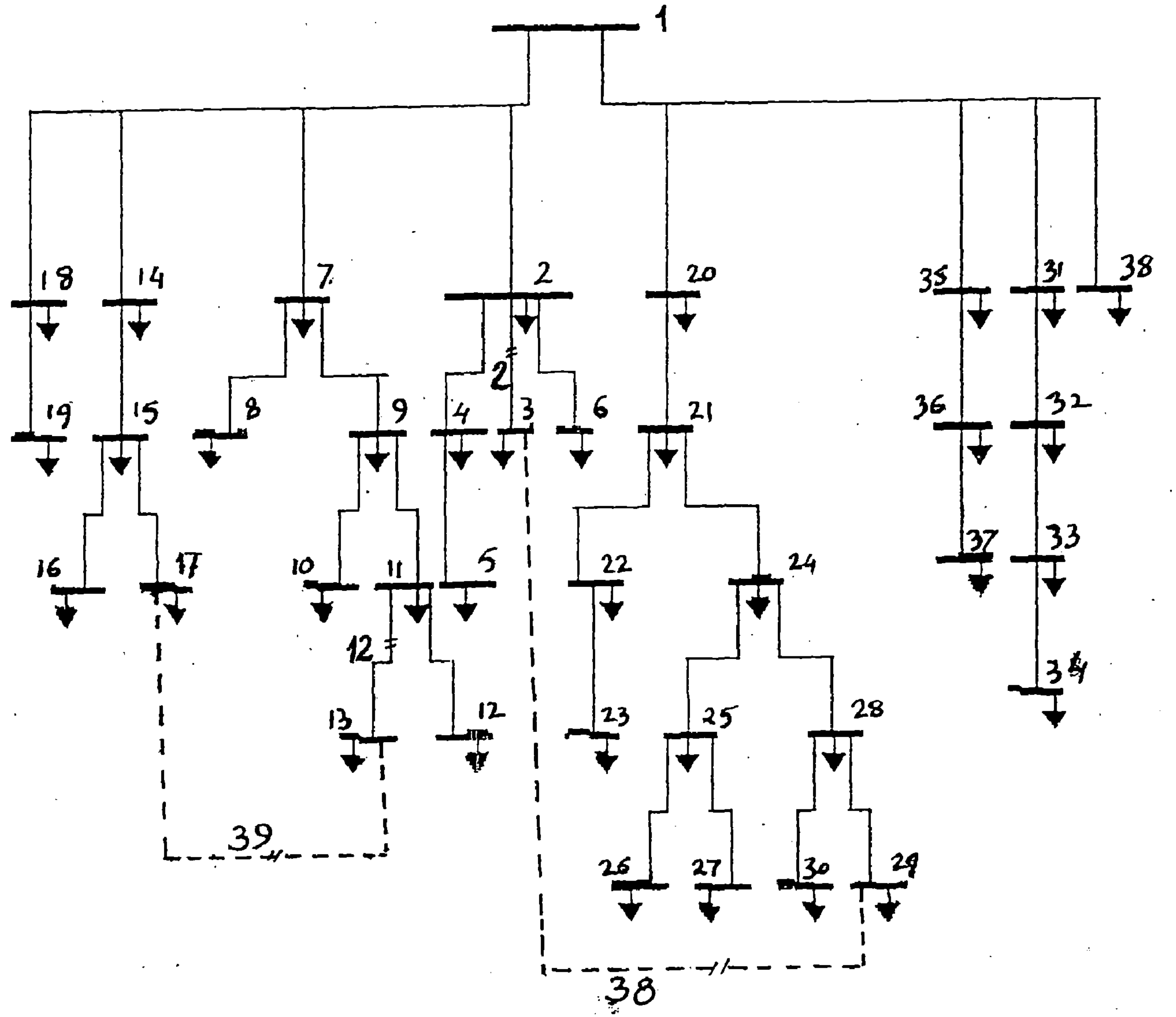
The 33 kv, 38 bus system of Madhya Pradesh State Electricity Board for Bhopal city is shown in figure 5.1. The system already has a provision of separation lines which are indicated by the dotted lines i.e. line numbers 38 and 39. The line parameters are shown in Table 5.1. the loads on each bus are shown in Table 5.2.

The dotted lines are reconnected to make the system non-radial and the load flow solution of the resulting non radial system was found using Gauss Seidel algorithm. Then the pair of lines meeting at a bus where power is fed from two or more adjacent buses are found .The voltage solutions of the said non radial network and the possible set of pairs of separation lines have been reported in Table 5.3.

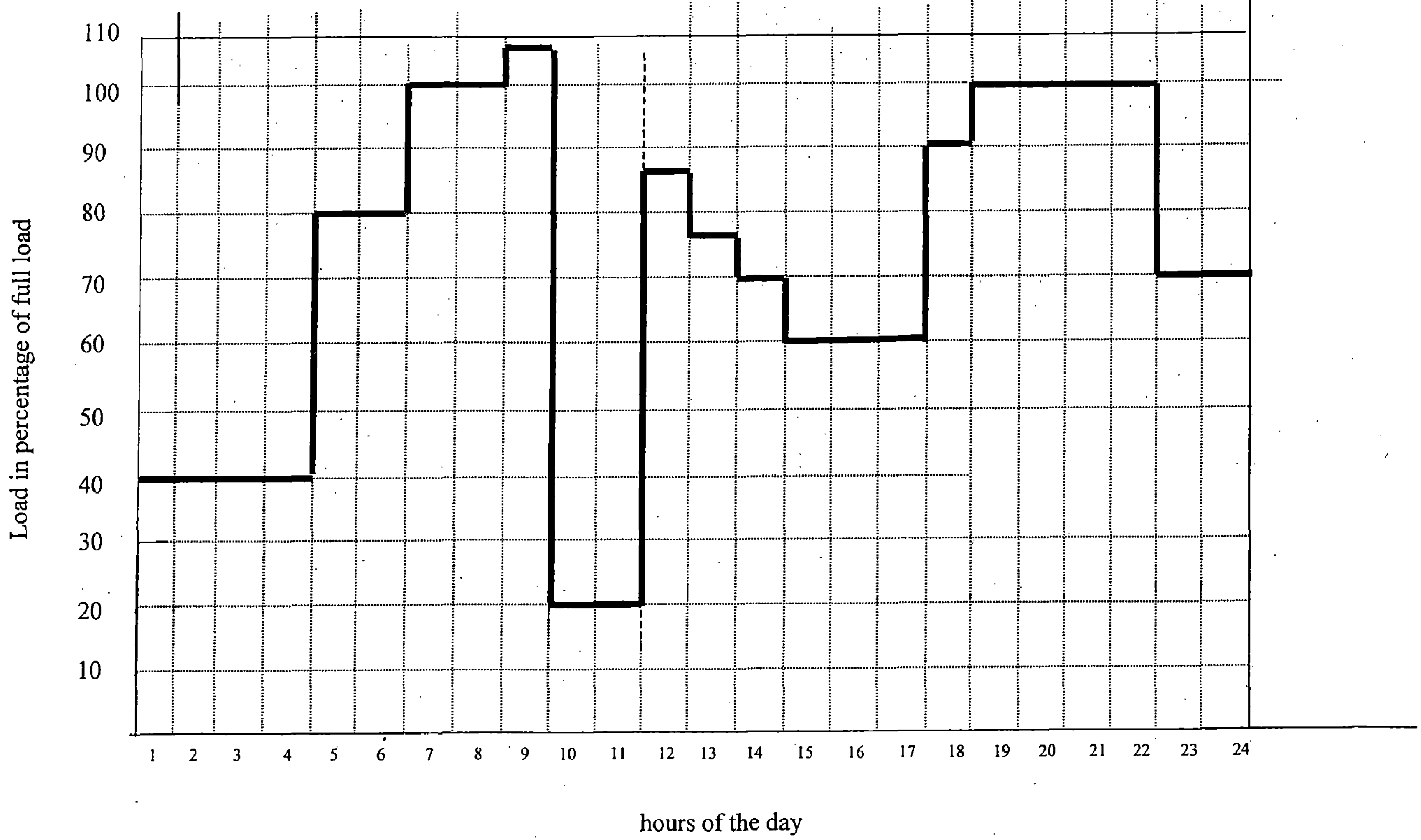
All the possible options of separation lines are then formed by picking on line from each pair. Thus if there are n pairs of lines, there will be $2^n$ combinations of possible options. Since there are two pairs of separation lines (2-38 and 12- 39), there will be 4 possible options viz. 2 -12 , 2-39, 38-12 and 38 -39. If now any of the options is removed from the network, the system gets converted into the radial one.

The program is now run with each options separated from the system with Z-loop algorithm and solution of each option is found out. The solutions are found out for each hour of the day as per the load curve of the system shown in Fig. 5.2. the option giving

FIG.5-1 - 33 KV , 38 BUS POWER DISTRIBUTION NETWORK OF BHOPAL



## FIG. 5.2 LOAD CURVE FOR 38 BUS POWER DISTIRBUTION SYSTEM OF BHOPAL CITY




Full load = 84,685 KW

**Table 5-1 Line Parameters of 33 KV, 38 Bus System of Bhopla City**

| Connection | | Resistance in ohm | Reactance in ohm |
|---|---|---|---|
| 1 | 2 | 1.2904 | 1.366 |
| 2 | 2 | 0.055 | 0.0582 |
| 2 | 4 | 0.5499 | 0.5821 |
| 4 | 5 | 0.1833 | 0.194 |
| 2 | 6 | 0.11 | 0.1164 |
| 1 | 7 | 0.9495 | 1.0051 |
| 7 | 8 | 0.0733 | 0.0776 |
| 7 | 9 | 0.3776 | 0.3997 |
| 9 | 10 | 0.055 | 0.0582 |
| 9 | 11 | 0.11 | 0.1164 |
| 11 | 12 | 0.011 | 0.0116 |
| 11 | 13 | 0.0073 | 0.0066 |
| 1 | 14 | 3.2994 | 3.4926 |
| 14 | 15 | 1.6497 | 1.7463 |
| 15 | 16 | 0.2566 | 0.2716 |
| 15 | 17 | 0.4106 | 0.4346 |
| 1 | 18 | 0.5132 | 0.5433 |
| 18 | 19 | 0.1466 | 0.1552 |
| 1 | 20 | 0.2933 | 0.3105 |
| 20 | 21 | 0.0183 | 0.0194 |
| 21 | 22 | 0.0073 | 0.0078 |
| 22 | 23 | 0.8358 | 0.8848 |
| 21 | 24 | 0.6599 | 0.6985 |
| 24 | 25 | 0.6232 | 0.6597 |
| 25 | 26 | 0.3299 | 0.3493 |
| 25 | 27 | 0.066 | 0.0699 |
| 24 | 28 | 0.0477 | 0.0504 |
| 28 | 29 | 0.011 | 0.0116 |
| 28 | 30 | 0.1466 | 0.1552 |
| 1 | 31 | 0.2933 | 0.3105 |
| 31 | 32 | 0.7332 | 0.7761 |
| 32 | 33 | 0.6232 | 0.6597 |
| 33 | 34 | 0.5646 | 0.5976 |
| 1 | 35 | 1.3491 | 1.4281 |
| 35 | 36 | 0.7405 | 0.7839 |
| 36 | 37 | 8.4318 | 8.9255 |
| 1 | 38 | 0.9165 | 0.9702 |
| 3 | 29 | 0.0158 | 0.0637 |
| 13 | 17 | 0.0648 | 0.1035 |

**Table 5.2 Bus loading of 33 KV, 38 Bus System of Bhopal City**

| Bus code | Active power in KW (X 10000) | Reactive power in KVAR (X 10000) |
|---|---|---|
| 2 | 0 | 0 |
| 3 | -0.4975 | -0.0498 |
| 4 | -0.995 | -0.0996 |
| 5 | -0.0796 | -0.008 |
| 6 | -0.2212 | -0.0221 |
| | 0 | 0 |
| 8 | -0.2212 | -0.0221 |
| 9 | 0 | 0 |
| 10 | -0.4975 | -0.0498 |
| 11 | -0.24 | -0.024 |
| 12 | -0.49 | -0.049 |
| 13 | -0.4975 | -0.0498 |
| 14 | 0 | 0 |
| 15 | -0.0796 | -0.0079 |
| 16 | -0.0342 | -0.0034 |
| 17 | -0.0696 | -0.007 |
| 18 | 0 | 0 |
| 19 | -0.249 | -0.025 |
| 20 | -0.4975 | -0.0498 |
| 21 | 0 | 0 |
| 22 | 0 | 0 |
| 23 | -0.4975 | -0.0498 |
| 24 | 0 | 0 |
| 25 | 0 | 0 |
| 26 | -0.041 | -0.0041 |
| 27 | -0.06 | -0.006 |
| 28 | 0 | 0 |
| 29 | -0.298 | -0.03 |
| 30 | -0.4975 | -0.0498 |
| 31 | -0.4975 | -0.0498 |
| 32 | 0 | 0 |
| 33 | 0 | 0 |
| 34 | -0.4975 | -0.0498 |
| 35 | -0.995 | -0.0996 |
| 36 | -0.109 | -0.0109 |
| 37 | -0.2345 | -0.0234 |
| 38 | -0.0716 | -0.0072 |

**Table. 5.3 Solutions of Non Radial network using Gauss Seidel Method for the Determination of Combination of Separation Line Options**

| Bus code | Real Voltage( P.U.) | Imaginary Voltage (P.U.) |
|---|---|---|
| 1 | 1.053 | 0 |
| 2 | 1.036221 | -0.012299 |
| 3 | 1.036216 | -0.012248 |
| 4 | 1.030315 | -0.017195 |
| 5 | 1.030169 | -0.01731 |
| 6 | 1.03598 | -0.012495 |
| 7 | 1.035691 | -0.01156 |
| 8 | 1.035533 | -0.011685 |
| 9 | 1.029661 | -0.015513 |
| 10 | 1.029391 | -0.015732 |
| 11 | 1.028469 | -0.016224 |
| 12 | 1.028419 | -0.016258 |
| 13 | 1.028457 | -0.016229 |
| 14 | 1.037714 | -0.010105 |
| 15 | 1.03008 | -0.01516 |
| 16 | 1.029996 | -0.015223 |
| 17 | 1.028661 | -0.016041 |
| 18 | 1.051764 | -0.001007 |
| 19 | 1.051411 | -0.001305 |
| 20 | 1.046206 | -0.004534 |
| 21 | 1.045871 | -0.004749 |
| 22 | 1.045835 | -0.004772 |
| 23 | 1.041754 | -0.008245 |
| 24 | 1.036975 | -0.011419 |
| 25 | 1.036358 | -0.011768 |
| 26 | 1.036224 | -0.011873 |
| 27 | 1.036319 | -0.011793 |
| 28 | 1.036379 | -0.011882 |
| 29 | 1.036295 | -0.011953 |
| 30 | 1.035656 | -0.012486 |
| 31 | 1.05015 | -0.002411 |
| 32 | 1.04656 | -0.005416 |
| 33 | 1.043509 | -0.007984 |
| 34 | 1.040745 | -0.010324 |
| 35 | 1.034936 | -0.015058 |
| 36 | 1.03234 | -0.017175 |
| 37 | 1.011953 | -0.033691 |
| 38 | 1.052366 | -0.000541 |

Number of iterations required= 243
Total load of the system= 84685 KW
Total real line losses of the system =1200.227812 KW
Set of pair of lines are 2 38
12 39

Note: Base Voltage = 33 KV

## Table. 5.4 SUMMARY OF RESULTS OF 38 BUS SYSTEM

| | Existing option (38, 39) | | | Option-1(line 2 and 12) | | | Option-2 (line 2 and 39 ) | | | Option-3 (line 38 and 12 ) | | | Option-4 (line 38 and 39) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hours of the Day | Slack bus power (KW) | Total load(KW) | Total losses (KW) | Slack bus power (KW) | Total load(KW) | Total energy losses (KWh) | Slack bus power (KW) | Total load(KW) | Total energy losses (KWh) | Slack bus power (KW) | Total load(KW) | Total energy losses (KWh) | Slack bus power (KW) | Total load(KW) | Total energy losses (KWh) |
| Ist hr | 34085.90 | 33874.00 | 211.87 | 34078.59 | 33874.00 | 204.58 | 34078.38 | 33874.00 | 204.38 | 34086.11 | 33874.00 | 212.07 | 34085.90 | 33874.00 | 211.87 |
| 2nd hr | 34085.90 | 33874.00 | 211.87 | 34078.59 | 33874.00 | 204.58 | 34078.38 | 33874.00 | 204.38 | 34086.11 | 33874.00 | 212.07 | 34085.90 | 33874.00 | 211.87 |
| 3rd hr | 34085.90 | 33874.00 | 211.87 | 34078.59 | 33874.00 | 204.58 | 34078.38 | 33874.00 | 204.38 | 34086.11 | 33874.00 | 212.07 | 34085.90 | 33874.00 | 211.87 |
| 4th hr | 34085.90 | 33874.00 | 211.87 | 34078.59 | 33874.00 | 204.58 | 34078.38 | 33874.00 | 204.38 | 34086.11 | 33874.00 | 212.07 | 34085.90 | 33874.00 | 211.87 |
| 5th hr | 68610.83 | 67748.00 | 862.84 | 68580.67 | 67748.00 | 832.51 | 68579.33 | 67748.00 | 831.20 | 68612.17 | 67748.00 | 864.15 | 68610.83 | 67748.00 | 862.84 |
| 6th hr | 68610.83 | 67748.00 | 862.84 | 68580.67 | 67748.00 | 832.51 | 68579.33 | 67748.00 | 831.20 | 68612.17 | 67748.00 | 864.15 | 68610.83 | 67748.00 | 862.84 |
| 7th hr | 86045.75 | 84685.00 | 1360.71 | 85997.58 | 84685.00 | 1312.38 | 85995.05 | 84685.00 | 1309.86 | 86048.28 | 84685.00 | 1363.23 | 86045.75 | 84685.00 | 1360.71 |
| 8th hr | 86045.75 | 84685.00 | 1360.71 | 85997.58 | 84685.00 | 1312.38 | 85995.05 | 84685.00 | 1309.86 | 86048.28 | 84685.00 | 1363.23 | 86045.75 | 84685.00 | 1360.71 |
| 9th hr | 93052.97 | 91459.79 | 1593.10 | 92996.33 | 91459.79 | 1536.27 | 92993.15 | 91459.79 | 1533.11 | 93056.14 | 91459.79 | 1596.27 | 93052.97 | 91459.79 | 1593.10 |
| 10th hr | 16989.63 | 16937.00 | 52.51 | 16987.75 | 16937.00 | 50.72 | 16987.67 | 16937.00 | 50.68 | 16989.71 | 16937.00 | 52.54 | 16989.63 | 16937.00 | 52.51 |
| 11th hr | 16989.63 | 16937.00 | 52.51 | 16987.75 | 16937.00 | 50.72 | 16987.67 | 16937.00 | 50.68 | 16989.71 | 16937.00 | 52.54 | 16989.63 | 16937.00 | 52.51 |
| 12th hr | 73829.04 | 72829.10 | 999.86 | 73793.92 | 72829.10 | 964.61 | 73792.31 | 72829.10 | 963.00 | 73830.64 | 72829.10 | 1001.48 | 73829.04 | 72829.10 | 999.86 |
| 13th hr | 65137.95 | 64360.59 | 777.29 | 65110.61 | 64360.59 | 750.03 | 65109.43 | 64360.59 | 748.89 | 65139.13 | 64360.59 | 778.42 | 65137.95 | 64360.59 | 777.29 |
| 14th hr | 59937.13 | 59279.49 | 657.60 | 59914.16 | 59279.49 | 634.62 | 59913.24 | 59279.49 | 633.72 | 59938.04 | 59279.49 | 658.50 | 59937.13 | 59279.49 | 657.60 |
| 15th hr | 51292.09 | 50811.00 | 480.96 | 51275.51 | 50811.00 | 464.24 | 51274.86 | 50811.00 | 463.65 | 51292.72 | 50811.00 | 481.55 | 51292.09 | 50811.00 | 480.96 |
| 16th hr | 51292.09 | 50811.00 | 480.96 | 51275.51 | 50811.00 | 464.24 | 51274.86 | 50811.00 | 463.65 | 51292.72 | 50811.00 | 481.55 | 51292.09 | 50811.00 | 480.96 |
| 17th hr | 51292.09 | 50811.00 | 480.96 | 51275.51 | 50811.00 | 464.24 | 51274.86 | 50811.00 | 463.65 | 51292.72 | 50811.00 | 481.55 | 51292.09 | 50811.00 | 480.96 |
| 18th hr | 77313.69 | 76216.51 | 1097.07 | 77274.85 | 76216.51 | 1058.30 | 77273.04 | 76216.51 | 1056.45 | 77315.51 | 76216.51 | 1098.91 | 77313.69 | 76216.51 | 1097.07 |
| 19th hr | 86045.75 | 84685.00 | 1360.71 | 85997.58 | 84685.00 | 1312.38 | 85995.05 | 84685.00 | 1309.86 | 86048.28 | 84685.00 | 1363.23 | 86045.75 | 84685.00 | 1360.71 |
| 20th hr | 86045.75 | 84685.00 | 1360.71 | 85997.58 | 84685.00 | 1312.38 | 85995.05 | 84685.00 | 1309.86 | 86048.28 | 84685.00 | 1363.23 | 86045.75 | 84685.00 | 1360.71 |
| 21st hr | 86045.75 | 84685.00 | 1360.71 | 85997.58 | 84685.00 | 1312.38 | 85995.05 | 84685.00 | 1309.86 | 86048.28 | 84685.00 | 1363.23 | 86045.75 | 84685.00 | 1360.71 |
| 22nd hr | 86045.75 | 84685.00 | 1360.71 | 85997.58 | 84685.00 | 1312.38 | 85995.05 | 84685.00 | 1309.86 | 86048.28 | 84685.00 | 1363.23 | 86045.75 | 84685.00 | 1360.71 |
| 23rd hr | 59937.13 | 59279.49 | 657.60 | 59914.16 | 59279.49 | 634.62 | 59913.24 | 59279.49 | 633.72 | 59938.04 | 59279.49 | 658.50 | 59937.13 | 59279.49 | 657.60 |
| 24th hr | 59937.13 | 59279.49 | 657.60 | 59914.16 | 59279.49 | 634.62 | 59913.24 | 59279.49 | 633.72 | 59938.04 | 59279.49 | 658.50 | 59937.13 | 59279.49 | 657.60 |
| **Total energy loss per day in KWH** | | | **18725.47** | | | **18064.79** | | | **18034.02** | | | **18756.24** | | | **18725.47** |

Option -2 Should be selected as it is giving lowest energy loss per day.

Saving per day in R's when compared to the existiing option @Rs 2/kwh= (18725.47-18034.02) *2=1,382.89

Saving in Rs per year-------------------------------------------- 504,754.88

## VOLTAGE SOLUTIONS OF EXISTING OPTION (38-39) AND LOWEST LOSS OPTION( 2 -39)

**Table 5.5- Solutions of Existing option ( Full load)**

| Bus code | Real Votage | Imaginary Voltage | Voltage magnitude |
|---|---|---|---|
| 1 | 1.053 | 0 | 1.053000 |
| 38 | 1.0524 | -0.0005 | 1.052400 |
| 35 | 1.0349 | -0.0151 | 1.035010 |
| 31 | 1.0501 | -0.0024 | 1.050103 |
| 20 | 1.0476 | -0.0046 | 1.047610 |
| 18 | 1.0518 | -0.0011 | 1.051801 |
| 14 | 1.0471 | -0.0051 | 1.047112 |
| 7 | 1.0344 | -0.0154 | 1.034515 |
| 2 | 1.0297 | -0.0193 | 1.029881 |
| 36 | 1.0323 | -0.0172 | 1.032443 |
| 32 | 1.0466 | -0.0055 | 1.046614 |
| 21 | 1.0473 | -0.0049 | 1.047311 |
| 19 | 1.0514 | -0.0014 | 1.051401 |
| 15 | 1.0441 | -0.0076 | 1.044128 |
| 9 | 1.0279 | -0.0209 | 1.028112 |
| 8 | 1.0343 | -0.0156 | 1.034418 |
| 6 | 1.0294 | -0.0195 | 1.029585 |
| 4 | 1.0237 | -0.0243 | 1.023988 |
| 3 | 1.0294 | -0.0196 | 1.029587 |
| 37 | 1.012 | -0.0337 | 1.012561 |
| 33 | 1.0435 | -0.0081 | 1.043531 |
| 24 | 1.0415 | -0.0098 | 1.041546 |
| 22 | 1.0473 | -0.0049 | 1.047311 |
| 17 | 1.0438 | -0.0078 | 1.043829 |
| 16 | 1.044 | -0.0077 | 1.044028 |
| 11 | 1.0265 | -0.022 | 1.026736 |
| 10 | 1.0276 | -0.0211 | 1.027817 |
| 5 | 1.0235 | -0.0244 | 1.023791 |
| 34 | 1.0407 | -0.0104 | 1.040752 |
| 28 | 1.0411 | -0.0101 | 1.041149 |
| 25 | 1.0409 | -0.0103 | 1.040951 |
| 23 | 1.0432 | -0.0084 | 1.043234 |
| 13 | 1.0265 | -0.022 | 1.026736 |
| 12 | 1.0265 | -0.0221 | 1.026738 |
| 30 | 1.0404 | -0.0107 | 1.040455 |
| 29 | 1.0411 | -0.0101 | 1.041149 |
| 27 | 1.0408 | -0.0104 | 1.040852 |
| 26 | 1.0407 | -0.0104 | 1.040752 |
| Average Voltage | | | 1.038498 |
| Standard ddeviation | | | 0.009595 |

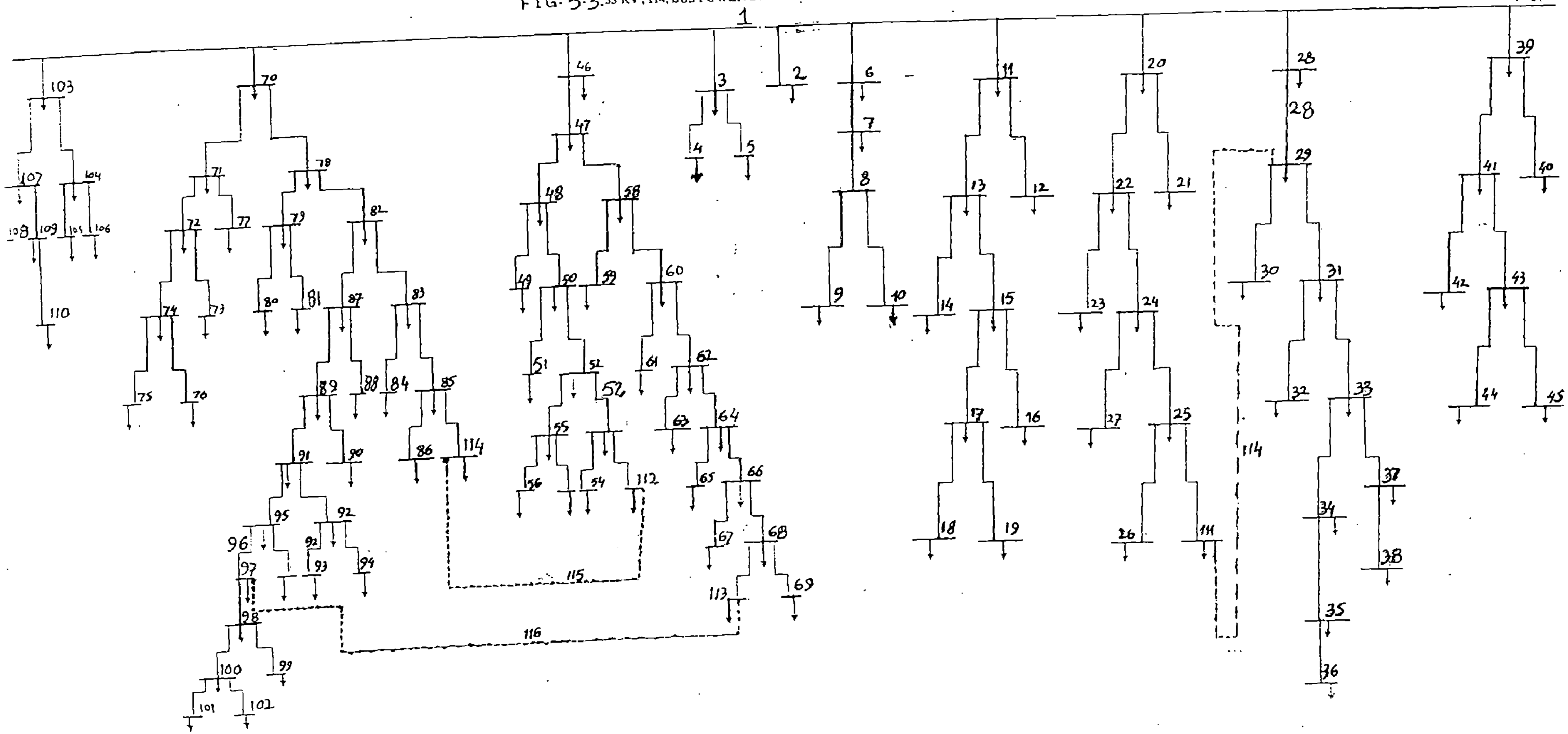
Number of iterations required=11
Lines separated are 38 39(option- 2)
Real power of slack bus 1=86045.7516 KW
Total load of the system= 84685 kw
Total real line loss of existing option is 1360.7146 KW

**Table 5.6 - Solutions of Existing option ( Full load)**

| Bus code | Real Votage | Imaginary Voltage | Voltage magnitude |
|---|---|---|---|
| 1 | 1.053 | 0 | 1.053000 |
| 38 | 1.0524 | -0.0005 | 1.052400 |
| 35 | 1.0349 | -0.0151 | 1.035010 |
| 31 | 1.0501 | -0.0024 | 1.050103 |
| 20 | 1.0461 | -0.0059 | 1.046117 |
| 18 | 1.0518 | -0.0011 | 1.051801 |
| 14 | 1.0471 | -0.0051 | 1.047112 |
| 7 | 1.0344 | -0.0154 | 1.034515 |
| 2 | 1.0363 | -0.014 | 1.036395 |
| 36 | 1.0323 | -0.0172 | 1.032443 |
| 32 | 1.0466 | -0.0055 | 1.046614 |
| 21 | 1.0457 | -0.0061 | 1.045718 |
| 19 | 1.0514 | -0.0014 | 1.051401 |
| 15 | 1.0441 | -0.0076 | 1.044128 |
| 9 | 1.0279 | -0.0209 | 1.028112 |
| 8 | 1.0343 | -0.0156 | 1.034418 |
| 6 | 1.036 | -0.0142 | 1.036097 |
| 4 | 1.0304 | -0.0189 | 1.030573 |
| 37 | 1.012 | -0.0337 | 1.012561 |
| 33 | 1.0435 | -0.0081 | 1.043531 |
| 24 | 1.0366 | -0.0138 | 1.036692 |
| 22 | 1.0457 | -0.0062 | 1.045718 |
| 17 | 1.0438 | -0.0078 | 1.043829 |
| 16 | 1.044 | -0.0077 | 1.044028 |
| 11 | 1.0265 | -0.022 | 1.026736 |
| 10 | 1.0276 | -0.0211 | 1.027817 |
| 5 | 1.0302 | -0.019 | 1.030375 |
| 34 | 1.0407 | -0.0104 | 1.040752 |
| 28 | 1.036 | -0.0143 | 1.036099 |
| 25 | 1.036 | -0.0144 | 1.036100 |
| 23 | 1.0416 | -0.0097 | 1.041645 |
| 13 | 1.0265 | -0.022 | 1.026736 |
| 12 | 1.0265 | -0.0221 | 1.026738 |
| 30 | 1.0353 | -0.015 | 1.035409 |
| 29 | 1.0359 | -0.0144 | 1.036000 |
| 27 | 1.0359 | -0.0144 | 1.036000 |
| 26 | 1.0359 | -0.0145 | 1.036001 |
| 3 | 1.0358 | -0.0147 | 1.035904 |
| Average Voltage | | | 1.038280 |
| Standard ddeviation | | | 0.008725 |

Number of iterations required=11
Lines separated are 2 39(option- 2)
Real power of slack bus 1=85995.0542 KW
Total load of the system= 84685 kw
Total real line loss of option 2 is 1309.8639 KW

FIG. 5.3. 33 KV, 114, BUS POWER DISTRIBUTION NETWORK OF BHOPAL



rise to the lowest energy losses is the selected. The lowest losses are coming for the option 2-39 in the given system. Table 5.4 shows the summary of all hourly solutions and saving per year resulting from the lowest loss option. The voltage solutions of the existing option (38 -39) are shown in Table 5.5 and the voltage solutions of the lowest loss option are shown in Table 5.6. A saving of Rs. 500,000/- has resulted when the existing location of switch is changed to the location corresponding to the lowest loss option.

## 5.2 114 - BUS SYSTEM:

Similarly, the software was run for 33 KV 114 bus system of again Bhopal which is shown in fig 5.3. The existing separation lines in this system were 114,115 and 116 as shown by the dotted lines. The lowest loss option has come to be 28,52 and 96.This option has given the lowest energy losses per day of 32326 KWH whereas the daily losses of the existing option comes to be 33135 KWH . A saving of about Rs.600,000/- per year (@ Rs 2/KWH) has resulted when the lowest loss option is compared to the existing option.

## 5.3 DISCUSSIONS

As illustrated in summary table 5.4 the energy saved per day in 38 bus system is 691.45 KWH which is about 3 percent of the daily energy loss. Although the percentage of energy saved is quite marginal, achievement of this saving can be made with no additional investment. Only the thing that has to be done is to change the position of the existing switch to the optimum location.

The above analysis is done for 33 KV distribution Sub-transmission circuit. Obviously, the energy losses at 11 KV or 400 V level will be far greater and hence the calculation for optimum separation line becomes more crucial for low voltage distribution circuits.

As we see from table 5·5 and 5·6 that the standard deviation of the voltages in the lowest loss option is 0.008725 which is less than the standard deviation of 0.009595 in case of the existing option. This indicates that the voltage profile has become more smooth in the lowest loss option and hence giving less losses.

# CONCLUSION AND RECOMMENDATIONS

## 6.1 CONCLUSION

The provision of separation line for the alternate supply in situations of fault already exists in almost all of the Indian electric utilities but it just solves the problem of service restoration without focussing on economy. In other words it just fulfills the consumer's need without applying any engineering. The present study has given way out on how to manage for the supply continuity economically in case of fault.

The economical solution of the problem involves finding the optimum location of the separation line so as to optimize the total active losses in the system. A methodology and a software has been developed for this purpose in this study. Two real 38 and 114 bus systems of Bhopal have been considered to check the validity of the software and hence to verify the correctness of the location of already existing separation line in these systems. The existing separation lines were reconnected and loops in the systems were formed. The Gauss Seidel algorithm was then applied to find all the probable options of separation lines of the looped system. Since the removal of any of these options turns the system into radial one, the Z-loop algorithm was applied to find the solution corresponding to each of the options for a given load curve and total energy losses per day were found out. The option resulting into lowest energy losses was then selected as the optimum separation line.

As we see from the results, locating the separation line optimally saves, though marginal, some energy and hence the money. By adopting the suggested option, a sum of Rs.500,000/- in 38 bus system and a sum of 600,000/- in 114 bus system per year can be saved. As already stated, though the saving is not much, it is achieved without any additional investment and it is of quite significance as every little thing is counted in today's competitive age. The existing location of the interconnecting switch just needs to be changed to get the said benefit.

## 6.2 RECOMMENDATIONS

Following recommendations are made for further improvement of the software prepared in the present study:

1. Though the software applies for verifying the optimality of existing location of separation line, it can be modified to determine the location of the separation line for a new distribution line to be constructed.

2. No particular search method has been applied to select the best separation line out of total available options from load flow solution. In the present study all of the possible combinations of separation lines have been considered and the program has been run for all these options to find the lowest - loss option. If some search method e.g. Genetic algorithm could be applied, the software would be more general and would be applicable for any system irrespective of its size.

# REFERENCES


1. A.S.Pabla, "Electric Power Distribution System", 3rd Edition.
2. B.M.Weedy, "Electric Power System",2nd Edition, John Wiley & Sons.
3. C.E Lin, Y.W, Huang & C.L.Huang, "Distribution System Load Flow Calculation with Micro- Computer", Electrical Power System Research, 13, (1987), pp.139-145.
4. E.L. Lakervi & E.J. Holmes, " Electricity Distribution Network Design", Peter Peregrinus Ltd., London, U.K., 1989.
5. John. S. Parson & B.G. Barnett, " Electrical Transmission and Distribution Reference Book", pp. 666-689 ( Distribution System).
6. Mesut. E .Baran and Felix, F.Wu, " Network Reconfiguration in Distribution System for Loss Reduction and Load Balancing", IEEE-Transaction, Power Delivery, Vol.4, No.2, April 1982.
7. R.M. Bundele, & S.R. Nam Joshi, "Distribution Automation- SCADA in BEST", 3rd International Conference on Power Development in Afro-Asian Countries, Kathmandu, Nepal, 4-7 March,1996.
8. R.N.Dhar, "Computer Added Power System Operation and Analysis", Tata Mc-Graw Hill, Publishing Company, 1982.
9. Robert, Lafore, "Object Oriented Programming in $C^{++}$", 3rd Edition, Waite Group.
10. R.Podmore, "Digital Computer Analysis of Power System Networks", Ph. D Thesis (Appendix-A), University of Centerbury, (New Zealand), 1972.'
11. Stagg. & E.L. Abiad, " Computer Methods in Power System Analysis", Mc-Graw Hill Book Company, pp. 258-267.
12. Turan, Gonen, "Electric Power Distribution System Engineering",Mc-Graw Hill Book Company, 1986.

**SOFTWARE FOR OPTIMAL LINE SEPARATION IN POWER DISTRIBUTION SYSTEM**

```
//THIS PROGRAM FINDS THE OPTIMAL SEPARATION LINE IN A RADIAL POWER
//DISTRIBUTION NETWORK

//BASE KV=33.0, BASE MVA=10.0

#include<fstream.h>
#include<iostream.h>
#include<string.h>
#include<iomanip.h>
#include<math.h>
#include<conio.h>

//****************FUNCTION DECLARATION**************************

void gauss_seidel();
void linknet(int,int);
void zloop(int);
float complexangle(float,float,float,float);

//***************INPUT AND OUTPUT FILES*************************

ifstream infile("gsrdin38.dat");
ofstream outfile("gsrdou38.dat");

//**************GLOBAL VARIABLES DECLARATION********************

const int size=300;
const float pi=3.1416;
int   list[size], next[size], lfrom[size], lto[size], ifar[size],
      i, j, s1, s2, s3, s4, k, nbbus, nbline, enda, endb, nodea,
      nodeb, end, sepline1[16], sepline2[16], sepline3[16], opt,hr;

float srev[size], simv[size], rev[size], imv[size], revs[size],
      imvs[size], rep[size], imp[size], cond[size], sucp[size],
      res[size], reac[size], retol, imtol, af, magadm, angadm,
      maxdelre, maxdelim, delrev, delimv, reflowij, imflowij,
      reflowji, imflowji, totalload, reloss[10], imloss[10],
      realloss[10], temprep[size], tempimp[size], minimumloss;

//************MAIN PROGRAM STARTS*******************************

void main()
 {
  outfile<<"\nOUTPUT RESULTS";
  //cout<<"\nEnter the number of buses"<<endl;
    infile>>nbbus;
  //cout<<"Enter the number of lines"<<endl;
    infile>>nbline;

for(i=1;i<=nbline;i++)
  {
  //cout<<"enter beginning and ending node for line  "<<i<<endl;
    infile>>lfrom[i]>>lto[i];
  }
```

```
outfile<<endl;
outfile<<endl;

for(i=1;i<=nbline;i++)
{
 //cout<<"resistance and reactance of line "<<i<<" ?"<<endl;
   infile>>res[i]>>reac[i];
   res[i]=res[i]/108.9;
   reac[i]=reac[i]/108.9;
   magadm=1/(sqrt(pow(res[i],2)+pow(reac[i],2)));
   angadm=complexangle(1.0,0.0,res[i],reac[i]);
   cond[i]=magadm*cos((pi/180)*angadm);
   sucp[i]=magadm*sin((pi/180)*angadm);
 }

//cout<<"Enter the slack buses"<<endl;
         infile>>s1;
//cout<<"Enter the real and imaginary voltages of slack bus"<<endl;
  infile>>rev[s1]>>imv[s1];
  revs[s1]=rev[s1];imvs[s1]=imv[s1];

//*************READING STARTING VOLTAGES OF BUSES****************

for(i=1;i<=nbbus;i++)
 {
  if(i!=s1)
   {
    //cout<<"Enter starting real and imaginary voltage of bus  "
    //     <<i<<endl;
    infile>>rev[i]>>imv[i];
    revs[i]=rev[i];imvs[i]=imv[i];
   }
 }

//cout<<"Enter real and imaginary bus voltage tolerences"<<endl;
  infile>>retol>>imtol;
  cout<<"Enter the accleration factor"<<endl;
  infile>>af;

//**********READING SCHEDULED POWER OF BUSES******************

for(i=1;i<=nbbus;i++)
 {
  if(i!=s1)
  {
   //cout<<"Enter active and reactive scheduled power for bus  "
   //     <<i<<endl;
   infile>>rep[i]>>imp[i];
   temprep[i]=rep[i];tempimp[i]=imp[i];
  }
 }

outfile<<endl<<"LINE PARMETERS OF 38 BUS SYSTEM"<<endl;
outfile<<setw(20)<<"Connection"<<setw(24)<<"Resistance(ohm)"
       <<setw(16)<<"Reactance(ohm)"<<endl;
outfile<<setw(20)<<"----------"<<setw(24)<<"---------------"
       <<setw(16)<<"--------------"<<endl;
```

```
outfile<<endl;

for(i=1;i<=nbline;i++)
outfile<<setw(13)<<lfrom[i]<<setw(5)<<lto[i]<<setprecision(6)
       <<setw(21)<<108.9*res[i]<<setprecision(6)<<setw(17)
       <<108.9*reac[i]<<endl;
outfile<<endl<<endl;
outfile<<setw(8)<<"Bus No."<<setw(14)<<"active p"<<setw(14)
       <<"Reac p"<<setw(14)<<"Real V"<<setw(14)<<"im V"<<endl;
outfile<<setw(8)<<"-------"<<setw(14)<<"------"<<setw(14)
       <<"------"<<setw(14)<<"------"<<setw(14)<<"----"<<endl;
outfile<<endl;

for(i=1;i<=nbbus;i++)
{
 if(i!=s1)
 outfile<<setw(8)<<i<<setprecision(4)<<setw(14)<<rep[i]
        <<setprecision(4)<<setw(14)<<imp[i]
        <<setprecision(4)<<setw(14)<<rev[i]
        <<setprecision(4)<<setw(14)<<imv[i]<<endl;
}

gauss_seidel();              //calling of gauss seidel function

for(opt=1;opt<5;opt++)
 {
  infile>>sepline1[opt]>>sepline2[opt];
  realloss[opt]=0;

  for(hr=1;hr<25;hr++)
  {
   zloop(opt);
   realloss[opt]=realloss[opt]+reloss[opt];
  }
 }

for(opt=1;opt<5;opt++)
outfile<<"\nTotal daily energy loss for option "<<opt
       <<"( lines "<<sepline1[opt]<<" and "
       <<sepline2[opt]<<") is "<<10000.00*realloss[opt]<<" KWH";

minimumloss=100000;
i=0;
for(opt=1;opt<5;opt++)
 if (realloss[opt]<minimumloss)
  {
   minimumloss=realloss[opt];i=opt;
  }
outfile<<endl;
outfile<<"\noption "<<i<< " is giving the lowest daily  energy losses"
       <<"\nHence this option should be selected";}
```

```
//*******FUNCTION TO DETERMINE NETWORK SOLUTION USING GAUSS
//SEIDEL METHOD**************

void gauss_seidel()
{
int   a[size], b[size], x, branch, count, flag;
float reybusd[size], imybusd[size], rei[size], imi[size],
      rebij, imbij, magai, angai, reai, imai, magbij, angbij,
      magft, angft, reft, imft, resum1, resum2, imsum1, imsum2,
      reloss, imloss, slackpowers1, reflows1j, magvj,
      magv1fromi,magv1toi;

linknet(nbbus,nbline);              //calling of linknet function

//*************DIAGONAL ELEMENTS OF Y-BUS MATRIX****************

for(i=1;i<=nbbus;i++)
 {
 end=list[i];
 reybusd[i]=0;imybusd[i]=0;
 do
  {
   branch=(end+1)/2;
   end=next[end];
   reybusd[i]=reybusd[i]+cond[branch];
   imybusd[i]=imybusd[i]+sucp[branch];
  }
  while(end!=0);
 }

//******************ITERATION STARTS*************************

clrscr();
cout<<"PROGRAM IS RUNNING, WAIT FOR 10 SECONDS";

for(i=1;i<=nbbus;i++)
 {
  srev[i]=rev[i];
  simv[i]=imv[i];
 }

  k=0;
  do
   {
    maxdelre=0;maxdelim=0;
    k=k+1;
    for(i=1;i<=nbbus;i++)
    {
     if(i!=s1)
     {
       magai=sqrt(pow(rep[i],2)+pow(imp[i],2))
            /sqrt(pow(reybusd[i],2)+pow(imybusd[i],2));
       angai=complexangle(rep[i],-imp[i],reybusd[i],imybusd[i]);
        reai=magai*cos((pi/180)*angai);
        imai=magai*sin((pi/180)*angai);
       magft=sqrt(pow(reai,2)+pow(imai,2))/sqrt(pow(srev[i],2)
```

```
            +pow(simv[i],2));
        angft=complexangle(reai,imai,srev[i],-simv[i]);
        reft=magft*cos((pi/180)*angft);
        imft=magft*sin((pi/180)*angft);

         end=list[i];
         resuml=0;imsuml=0;
    do
     {
      j=ifar[end];
      branch=(end+1)/2;
      end=next[end];
      magbij=sqrt(pow(-cond[branch],2)+pow(-sucp[branch],2))
             /sqrt(pow(reybusd[i],2)+pow(imybusd[i],2));
      angbij=complexangle(-cond[branch],-sucp[branch],
             reybusd[i],imybusd[i]);
      rebij=magbij*cos((pi/180)*angbij);
      imbij=magbij*sin((pi/180)*angbij);
      resuml=resuml+(rebij*srev[j]-imbij*simv[j]);
      imsuml=imsuml+(rebij*simv[j]+imbij*srev[j]);
      }
 while(end!=0);

      rev[i]=reft-resuml;
      imv[i]=imft-imsuml;
      delrev=sqrt(pow((rev[i]-srev[i]),2));
      delimv=sqrt(pow((imv[i]-simv[i]),2));
      if(delrev>maxdelre) maxdelre=delrev;
      if(delimv>maxdelim) maxdelim=delimv;
      srev[i]=srev[i]+af*(rev[i]-srev[i]);
      simv[i]=simv[i]+af*(imv[i]-simv[i]);
    }
   }
 }
while(maxdelre>retol||maxdelim>imtol);

//************OUTPUT OF BUS VOLTAGES***********************
outfile<<"\nSOLUTIONS OF LOOPED 38 BUS NETWORK ";
outfile<<endl<<endl;;
outfile<<setw(6)<<"Bus no"<<setw(16)<<"Re Vol"
       <<setw(16)<<"Im Vol"<<endl;
outfile<<setw(6)<<"------"<<setw(16)<<"------"
       <<setw(16)<<"------"<<endl;
outfile<<endl;

for(i=1;i<=nbbus;i++)
outfile<<setw(6)<<i<<setprecision(6)<<setw(16)<<rev[i]
         <<setprecision(6)<<setw(16)<<imv[i]<<endl;
outfile<<endl;
outfile<<"   Number of iterations required="<<k<<endl;
totalload=0;
for(i=2;i<=nbbus;i++)
totalload=totalload-rep[i];
outfile<<"\nTotal load of the system= "
       <<10000*totalload<<" kw";
```

```
//*************CALCULATION OF LINE LOSSES**********************

reloss=0;imloss=0;
for(i=1;i<=nbbus;i++)
 {
  end=list[i];
  do
    {
     j=ifar[end];
     branch=(end+1)/2;
     end=next[end];
     if(j>i)
      {

       reflowij=cond[branch]*(pow(rev[i],2)+pow(imv[i],2)
                -rev[i]*rev[j]-imv[i]*imv[j])-sucp[branch]
                *(rev[j]*imv[i]-rev[i]*imv[j]);

      imflowij=sucp[branch]*(pow(rev[i],2)+pow(imv[i],2)
               -rev[i]*rev[j]-imv[i]*imv[j])+cond[branch]
               *(rev[j]*imv[i]-rev[i]*imv[j]);

      reflowji=cond[branch]*(pow(rev[j],2)+pow(imv[j],2)
               -rev[j]*rev[i]-imv[j]*imv[i])-sucp[branch]
               *(rev[i]*imv[j]-rev[j]*imv[i]);

      imflowji=sucp[branch]*(pow(rev[j],2)+pow(imv[j],2)
               -rev[j]*rev[i]-imv[j]*imv[i])+cond[branch]
               *(rev[i]*imv[j]-rev[j]*imv[i]);


      reloss=reloss+reflowij+reflowji;
      imloss=imloss+imflowij+imflowji;
     }
   }
  while(end!=0);
 }
outfile<<"\ntotal real line losses="<<10000.00*reloss
       <<" KW"<<endl;

//*************CALCULATION OF SLACK BUS POWER******************

  end=list[s1];
  slackpowers1=0;
  do
    {
     j=ifar[end];
     branch=(end+1)/2;
     end=next[end];
     if(j>s1)
     reflows1j=cond[branch]*(pow(rev[s1],2)+pow(imv[s1],2)
               -rev[s1]*rev[j]-imv[s1]*imv[j])
               -sucp[branch]*(rev[j]*imv[s1]-rev[s1]*imv[j]);
     slackpowers1=slackpowers1+reflows1j;
    }
while(end!=0);
```

```
outfile<<"Real power of slack bus 1="
        <<10000.00*slackpowers1<<"  KW"<<endl;

x=0;
for(i=1;i<=nbline;i++)
 {
  end=list[lto[i]];
  if(next[end]!=0)
   {
     magvltoi=sqrt(pow(rev[lto[i]],2)+pow(imv[lto[i]],2));
     magvlfromi=sqrt(pow(rev[lfrom[i]],2)+pow(imv[lfrom[i]],2));
     do
       {
        j=ifar[end];
        magvj=sqrt(pow(rev[j],2)+pow(imv[j],2));
        branch=(end+1)/2;
        end=next[end];
        if(magvj>magvltoi&&magvlfromi>magvltoi)
        {
         x=x+1;
         count=0;
         a[count]=0;
         b[count]=0;
         do
           {
            count=count+1;

            if((i!=a[count]&&i!=b[count])&&(branch!=a[count]
                         &&branch!=b[count]))
            flag=1;else flag=0;
           }
while(flag==1&&count<x);

     if(i!=branch&&flag==1)
        {
        a[x]=i;
        b[x]=branch;
        outfile<<"\nSET OF POSSIBLE PAIRS OF LINES ARE   "
                <<a[x]<<"    "<<b[x];
        }
     }
   }
while(end!=0);
   }
  }
 }

//******FUNCTION TO DETERMINE THE SOLUTIONS OF RADIAL NETWORK***

void zloop(int opt)
 {

   int    renbus[size], renline[size], nbusrad, nlinerad, branch,
          signal, m, k, bus[size], count, flag, line[size], dist[size],
          forward[size], ind,mdistance, flag1[size], ind2, flag2[size];
   float prev[size], pimv[size], magib, angib,reib[size],
```

```
        imib[size], reil[size],imil[size], ree[size], ime[size],
        rei, imi, rec[size], imc[size], slackpowers, reflowsj;

//**** THIS PART OF THE PROGRAM RENUMBERS THE NETWORK BUSES
// AFTER SEPARATING A PAIR OF LINES**************

for(i=1;i<=nbbus;i++)
dist[i]=9999;
dist[1]=0;
forward[1]=1;
i=1;
ind=0;
end=list[1];
mdistance=1;
do
 {
  branch=(end+1)/2;
  if(branch!=sepline1[opt]&&branch!=sepline2[opt])
   {
    j=ifar[end];
    dist[j]=mdistance;
    i=i+1;
    forward[i]=j;
    ind=ind+1;
    flag1[ind]=j;
   }
  end=next[end];
 }
while(end!=0);

do
 {
  k=1;
  ind2=0;
  mdistance=mdistance+1;
  do
   {
    end=list[flag1[k]];
    branch=(end+1)/2;
    do
      {
       if(branch!=sepline1[opt]&&branch!=sepline2[opt])
        {
         j=ifar[end];
         if(dist[j]==9999)
          {
           ind2=ind2+1;
           i=i+1;
           forward[i]=j;
           dist[j]=mdistance;
           flag2[ind2]=j;
          }
        }
      end=next[end];
      branch=(end+1)/2;
     }
    while(end!=0);
```

```
    k=k+1;
   }
  while(k<=ind);
  ind=ind2;
  m=1;
  do
   {
    flag1[m]=flag2[m];
    m=m+1;
   }
   while(m<=ind2);
   }
  while(ind!=0);

//***********ITERATION STARTS**************************************

 k=0;
 for(i=1;i<=nbbus;i++)
  {
   srev[i]=revs[i];simv[i]=imvs[i];
   if(i!=s1)
    {
     rep[i]=temprep[i];imp[i]=tempimp[i];
    }
   }

switch(hr)
  {
   case 1:for(i=2;i<=nbbus;i++){rep[i]=0.40*rep[i];
                          imp[i]=0.40*imp[i];}break;

   case 2:for(i=2;i<=nbbus;i++){rep[i]=0.40*rep[i];
                          imp[i]=0.40*imp[i];}break;

   case 3:for(i=2;i<=nbbus;i++){rep[i]=0.40*rep[i];
                          imp[i]=0.40*imp[i];}break;

   case 4:for(i=2;i<=nbbus;i++){rep[i]=0.40*rep[i];
                          imp[i]=0.40*imp[i];}break;

   case 5:for(i=2;i<=nbbus;i++){rep[i]=0.80*rep[i];
                          imp[i]=0.80*imp[i];}break;

   case 6:for(i=2;i<=nbbus;i++){rep[i]=0.80*rep[i];
                          imp[i]=0.80*imp[i];}break;

   case 7:for(i=2;i<=nbbus;i++){rep[i]=1.00*rep[i];
                          imp[i]=1.00*imp[i];}break;

   case 8:for(i=2;i<=nbbus;i++){rep[i]=1.00*rep[i];
                          imp[i]=1.00*imp[i];}break;

   case 9:for(i=2;i<=nbbus;i++){rep[i]=1.08*rep[i];
                          imp[i]=1.08*imp[i];}break;

   case 10:for(i=2;i<=nbbus;i++){rep[i]=0.20*rep[i];
                          imp[i]=0.20*imp[i];}break;
```

```
  case 11:for(i=2;i<=nbbus;i++){rep[i]=0.20*rep[i];
                         imp[i]=0.20*imp[i];}break;

  case 12:for(i=2;i<=nbbus;i++){rep[i]=0.86*rep[i];
                         imp[i]=0.86*imp[i];}break;

  case 13:for(i=2;i<=nbbus;i++){rep[i]=0.76*rep[i];
                         imp[i]=0.76*imp[i];}break;

  case 14:for(i=2;i<=nbbus;i++){rep[i]=0.70*rep[i];
                         imp[i]=0.70*imp[i];}break;

  case 15:for(i=2;i<=nbbus;i++){rep[i]=0.60*rep[i];
                         imp[i]=0.60*imp[i];}break;

  case 16:for(i=2;i<=nbbus;i++){rep[i]=0.60*rep[i];
                         imp[i]=0.60*imp[i];}break;

  case 17:for(i=2;i<=nbbus;i++){rep[i]=0.60*rep[i];
                         imp[i]=0.60*imp[i];}break;

  case 18:for(i=2;i<=nbbus;i++){rep[i]=0.90*rep[i];
                          imp[i]=0.90*imp[i];}break;

  case 19:for(i=2;i<=nbbus;i++){rep[i]=1.00*rep[i];
                          imp[i]=1.00*imp[i];}break;

  case 20:for(i=2;i<=nbbus;i++){rep[i]=1.00*rep[i];
                          imp[i]=1.00*imp[i];}break;

  case 21:for(i=2;i<=nbbus;i++){rep[i]=1.00*rep[i];
                          imp[i]=1.00*imp[i];}break;

  case 22:for(i=2;i<=nbbus;i++){rep[i]=1.00*rep[i];
                          imp[i]=1.00*imp[i];}break;

  case 23:for(i=2;i<=nbbus;i++){rep[i]=0.70*rep[i];
                          imp[i]=0.70*imp[i];}break;

  case 24:for(i=2;i<=nbbus;i++){rep[i]=0.70*rep[i];
                          imp[i]=0.70*imp[i];}break;
 }

do
  {
   maxdelre=0;maxdelim=0;
   k=k+1;
   for(i=1;i<=nbbus;i++)
    {
     if(forward[i]!=s1)
      {
        magib=sqrt(pow(rep[forward[i]],2)+pow(imp[forward[i]],2))
              /sqrt(pow(srev[forward[i]],2)+pow(simv[forward[i]],2));
        angib=complexangle(rep[forward[i]],-imp[forward[i]],
              srev[forward[i]],-simv[forward[i]]);
        reib[forward[i]]=magib*cos((pi/180)*angib);
```

```
            imib[forward[i]]=magib*sin((pi/180)*angib);
          }
        }

   for(i=nbbus;i>=1;i--)
    {
     end=list[forward[i]];
     rei=0;imi=0;
    do
     {
      branch=(end+1)/2;
      if(branch!=sepline1[opt]&&branch!=sepline2[opt])
        {
          j=ifar[end];
          if(dist[j]>dist[forward[i]])
           {
            rei=rei+rec[branch];
            imi=imi+imc[branch];
           }
        else
          {
           rec[branch]=rei-reib[forward[i]];
           imc[branch]=imi-imib[forward[i]];
           ree[branch]=res[branch]*rec[branch]-reac[branch]*imc[branch];
           ime[branch]=res[branch]*imc[branch]+reac[branch]*rec[branch];
          }
        }
      end=next[end];
      }
     while(end!=0);
     }

    for(i=1;i<=nbbus;i++)
     {
      end=list[forward[i]];
      branch=(end+1)/2;
      do
       {
        if(branch!=sepline1[opt]&&branch!=sepline2[opt]){
        j=ifar[end];
        if(dist[j]>dist[forward[i]])
        {
         rev[j]=rev[forward[i]]-ree[branch];
         imv[j]=imv[forward[i]]-ime[branch];
        }
       }
      end=next[end];
      branch=(end+1)/2;
     }
     while(end!=0);
    }

    for(i=1;i<=nbbus;i++)
     {
      if(forward[i]!=s1)
       {
        delrev=sqrt(pow((rev[forward[i]]-srev[forward[i]]),2));
```

```
    delimv=sqrt(pow((imv[forward[i]]-simv[forward[i]]),2));
    if(delrev>maxdelre) maxdelre=delrev;
    if(delimv>maxdelim) maxdelim=delimv;
    srev[forward[i]]=srev[forward[i]]+
                    af*(rev[forward[i]]-srev[forward[i]]);
    simv[forward[i]]=simv[forward[i]]+
                    af*(imv[forward[i]]-simv[forward[i]]);
   }
  }
 }
 while(maxdelre>retol||maxdelim>imtol);

 outfile<<endl;
 outfile<<"\nFINAL VOLTAGES"<<endl;

 outfile<<endl<<endl;;
 outfile<<setw(6)<<"Bus no"<<setw(10)<<"Re Vol"
        <<setw(10)<<"Im Vol"<<endl;
 outfile<<setw(6)<<"------"<<setw(10)<<"------"
        <<setw(10)<<"------"<<endl;
 outfile<<endl;

for(i=1;i<=nbbus;i++)
outfile<<setw(6)<<i<<"-"<<"("<<forward[i]<<")"
       <<setprecision(4)<<setw(10)<<rev[forward[i]]
       <<setprecision(4)<<setw(10)<<imv[forward[i]]<<endl;
outfile<<endl;
outfile<<"   Number of iterations required="<<k<<endl;

outfile<<"\Lines separated are "<<sepline1[opt]<<" "<<sepline2[opt]
       <<"(option- "<<opt<<"  hour -"<<hr<<" )";

 end=list[s1];
 slackpowers=0;
 do
   {
    branch=(end+1)/2;
    if(branch!=sepline1[opt]&&branch!=sepline2[opt])
      {
       j=ifar[end];
       reflowsj=cond[branch]*(pow(rev[s1],2)+pow(imv[s1],2)
                -rev[s1]*rev[j]-imv[s1]*imv[j])
                -sucp[branch]*(rev[j]*imv[s1]-rev[s1]*imv[j]);
       slackpowers=slackpowers+reflowsj;
      }
   end=next[end];
   }
while(end!=0);

outfile<<"\nReal power of slack bus 1="
       <<10000.00*slackpowers<<"  KW";

totalload=0;
for(i=2;i<=nbbus;i++)
totalload=totalload-rep[i];
outfile<<"\nTotal load of the system= "
       <<10000.00*totalload<<" kw";
```

```
reloss[opt]=0;imloss[opt]=0;
for(i=1;i<=nbbus;i++)
 {
  end=list[forward[i]];
  do
   {
    j=ifar[end];
    branch=(end+1)/2;
    if(branch!=sepline1[opt]&&branch!=sepline2[opt])
   {
    if(dist[j]>dist[forward[i]])
     {
      reflowij=cond[branch]*(pow(rev[forward[i]],2)
               +pow(imv[forward[i]],2)-rev[forward[i]]*rev[j]
               -imv[forward[i]]*imv[j])-sucp[branch]*(rev[j]
               *imv[forward[i]]-rev[forward[i]]*imv[j]);
      imflowij=sucp[branch]*(pow(rev[forward[i]],2)
               +pow(imv[forward[i]],2)-rev[forward[i]]*rev[j]
               -imv[forward[i]]*imv[j])+cond[branch]*(rev[j]
               *imv[forward[i]]-rev[forward[i]]*imv[j]);
      reflowji=cond[branch]*(pow(rev[j],2)+pow(imv[j],2)
               -rev[j]*rev[forward[i]]-imv[j]*imv[forward[i]])
               -sucp[branch]*(rev[forward[i]]*imv[j]-rev[j]
               *imv[forward[i]]);
      imflowji=sucp[branch]*(pow(rev[j],2)+pow(imv[j],2)
              -rev[j]*rev[forward[i]]-imv[j]*imv[forward[i]])
              +cond[branch]*(rev[forward[i]]*imv[j]-rev[j]
              *imv[forward[i]]);
      reloss[opt]=reloss[opt]+reflowij+reflowji;
      imloss[opt]=imloss[opt]+imflowij+imflowji;
     }
   }
   end=next[end];
  }
 while(end!=0);
 }
 outfile<<"\ntotal real line losses of option "<<opt
        <<" is "<<10000.00*reloss[opt]<<" KW"<<endl;
}

//********FUNCTION TO CALCULATE THE ANGLE OF COMPLEX NUMBERS****

float complexangle(float renum,float imnum,
               float reden,float imden)
 {
  float anglenum,angleden,angle;
  const float pi=3.1416;

  if(renum>=0&&imnum>=0)
     anglenum=(180/pi)*(atan(imnum/renum));
  else if(renum<=0&&imnum>=0)
          anglenum=180+(180/pi)*(atan(imnum/renum));
  else if(renum<=0&&imnum<=0)
          anglenum=180+(180/pi)*(atan(imnum/renum));
  else
      anglenum=(180/pi)*(atan(imnum/renum));
```

```
if(reden>=0&&imden>=0)
   angleden=(180/pi)*(atan(imden/reden));
else if(reden<=0&&imden>=0)
        angleden=180+(180/pi)*(atan(imden/reden));
else if(reden<=0&&imden<=0)
          angleden=180+(180/pi)*(atan(imden/reden));
else
     angleden=(180/pi)*(atan(imden/reden));
     angle=anglenum-angleden;
return angle;
}

//*****LINKNET FUNCTION TO DETERMINE CONNECTIVITY***************

void linknet(int nbbus,int nbline)
 {
  for(i=1;i<=nbbus;i++)
  list[i]=0;
  for(i=1;i<=nbline;i++)
   {
    endb=2*i;
    enda=endb-1;
    nodea=lfrom[i];
    nodeb=lto[i];
    next[enda]=list[nodea];
    list[nodea]=enda;
    ifar[enda]=nodeb;
    next[endb]=list[nodeb];
    list[nodeb]=endb;
    ifar[endb]=nodea;
   }

 outfile<<endl;
 outfile<<endl;
 outfile<<setw(8)<<"node"<<setw(16)<<"list[node]"<<setw(8)
        <<"end"<<setw(16)<<"next[end]"<<setw(16)
        <<"ifar[end]"<<endl;
 outfile<<setw(8)<<"----"<<setw(16)<<"----------"<<setw(8)
        <<"---"<<setw(16)<<"---------"<<setw(16)
        <<"---------"<<endl;
 outfile<<endl;

 for(i=1;i<=nbbus;i++)
 outfile<<setw(8)<<i<<setw(16)<<list[i]<<setw(8)<<i
        <<setw(16)<<next[i]<<setw(16)<<ifar[i]<<endl;

 for(i=nbbus+1;i<=2*nbline;i++)
 outfile<<setw(32)<<i<<setw(16)<<next[i]<<setw(16)
        <<ifar[i]<<endl;
}

//*************** END OF THE PROGRAM*****************************
```