THE UNIVERSITY OF NAIROBI

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

OPTIMAL POWER FLOW FOR THE KENYA POWER SYSTEM USING THE STEEPEST DESCENT METHOD

A Thesis in

Electrical and Electronics Engineering

By

Gerald Muriithi Maina

Submitted in partial fulfillment of the requirements for the award of Degree of Master of Science in Electrical Engineering of the University of Nairobi

February 1995

/e approve the thesis of Gerald Muriithi Maina and certify that it is his original work and has not

een presented elsewhere for the award of a Master of Science Degree.

gnature | Date of Signature

1/11/1996

r. Maurice K. W. Mang'oli
rst Supervisor
D in Electrical Engineering
ept. of Electrical and Electronics Eng.
niversity of Nairobi

14/11/96

. Mucemi Gakuru
cond Supervisor
D in Electrical Engineering
ept. of Electrical and Electronics Eng.
niversity of Nairobi

ACKNOWLEDGEMENTS

I would like to thank my thesis supervisors Dr M. Mango'li. and Dr M. Gakuru for introducing me to this exciting field of research and further going on to offer excellent guidance in my thesis. I greatly appreciated their logical comments, guidance and criticism; without which the successful completion of this thesis would have been hard to accomplish.

A lot of thanks also go to other lecturers who laid a firm foundation for me during my course work including Prof. Ijumba, Dr. Mbuthia, Misters Orero, Bukhala and Walkade all of the Department of Electrical and Electronics Engineering, University of Nairobi.

I'd be extremely ungrateful if I failed to thank the German Academic Exchange Service (DAAD) who, in co-operation with the University of Nairobi, sponsored my two year Msc. programme and even agreed to extend the duration of my scholarship to enable me finish my research. DAAD's financial support was highly appreciated.

Much thanks also go to among others computer laboratory assistants Misters Chomba, Otieno and Owalla who willingly availed the computer facilities throughout my period of research, sometimes even at odd hours. Their patience and willingness to assist were extremely encouraging.

Finally, and most important, I'm immensely indebted to my parents, brothers and sisters for their unselfish contribution to my education.

ABSTRACT

Economy and reliability of service are important considerations in power system operation and planning. The primary objective in power system planning and operation is to minimise the cost of meeting the power requirements of the system over some appropriate period of time and in a manner consistent with reliable service.

The appropriate time could be short (say afew minutes) or long (say one year) depending on the nature of the energy resources available to the system. Obviously, the aim in the utilization of energy resources is to realise the greatest possible value during the period of operation in terms of fuel usage. Thermal or gas stations rely on fossil fuels like coal, diesel, gas etc for their operation. In addition to being expensive, fossil fuels are exhaustible and therefore thermal stations should be set to operate with minimum cost so as to reduce fuel usage. Hydro and geothermal stations have negligible operating costs but should nonetheless be operated efficiently if the available hydro and geothermal resources are to be exploited to the maximum.

The limiting factor is of course system constraints like voltage levels, power flow on transmission lines, generator output limits, etc., which should be maintained within an acceptable operating range to ensure reliability of supply.

An optimal power flow problem formulates an appropriate objective function (cost, real power loss, load shedding etc.) which is then minimised subject to satisfying system constraints like voltage levels, transformer taps, line flows etc. The solution can be obtained using several non - linear programming techniques which include the steepest descent method, Quadratic Programming Method (QPM) and the Gradient Projection Method (GPM) among others. In this thesis, the optimization problem is solved using the steepest descent method and targets the Kenya Power

System.

The solution technique is based on the Newton Raphson load flow formulation, a first order gradient adjustment algorithm for minimising the objective function and use of penalty functions to account for violation of dependent variables. These include system security constraints such as voltage levels on load buses and real and reactive power flows on transmission lines which, when satisfied, ensure a reliable and secure supply of power. Only the short term operational problem is addressed. The long term hydro-thermal co-ordination planning problem is recommended for future research.

TABLE OF CONTENTS

LIST OF SYMBOLS

OPF	Optimal Power Flow
VAR	Volt Ampere Reactive
QPM	Quadratic Programming Method
GPM	Gradient Projection Method
SUMT	Sequential Unconstrained Minimization Technique
LP	Linear Programming
δ	Vector of bus voltage angles
V	Vector of bus voltage magnitudes
ΔP	Vector of changes in bus real power
ΔQ	Vector of changes in bus reactive power
$\Delta\delta$	Vector of changes in bus voltage angles
ΔV	Vector of changes in bus voltage magnitudes
i	Bus number
n	Total number of buses in the network
I_i	Current injected into bus i
Y_{ij}	Bus admittance between bus i and j
θ_{ij}	Angle of bus admittance between i and j
S_i	Complex power at bus i
P_i	Net real power at bus i

Real power output of generator at bus i

Reactive power output of VAR source at bus i

Jacobian sub-matrices

Total cost of real power generation

Cost of generating real power at generator connected at bus i

Augmented Lagrangian function

Vector of equality constraints

Vector of inequality constraints

dependent variable

control or decision variable

independent variable

Lagrange multiplier

Changes in a control variable

Penalty function

weighting factor for penalty function Wj

step length in steepest descent direction

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVE

The objective of this thesis is to minimise the operational cost and / or real power loss of a power system network subject to meeting the expected load demand and maintaining reliable supply.

1.2. STATEMENT OF WORK:

The objective stated above is achieved by formulating an objective function (operational cost or real power loss) and minimising it subject to meeting both operational and equipment loading constraints, using the steepest descent method. In case of a cost objective function, optimal real power generations, voltage magnitudes at PV buses and transformer tap settings are obtained by minimising the fuel cost function. On the other hand, optimal reactive power generations, transformer taps and voltage magnitudes at PV buses are determined by minimising the real power loss function. In both cases, generator real power output, voltage magnitude at PV buses and transformer tap settings serve as control variables.

An initial load flow solution, using an assumed feasible operating point (by specifying real power generations, transformer tap settings and voltage magnitudes at PV buses) is obtained using the Newton Raphson method. A vector of gradients giving the sensitivity of the objective function with respect to changes in the control variables is determined and used to adjust the control variables from one operating point to another in the negative direction until the minimum value of the objective function is obtained. Penalty functions are used to account for violations of limits of dependent variables.

HISTORICAL REVIEW OF OPTIMAL POWER FLOW

The traditional approach to optimal power flow solution was by trial and error, where a nber of normal and contingency loading conditions like line and generator outages were evaluated ng load flows and an engineering judgement based on experience. Reactive power was allocated correct for any voltage violations that may arise, and more load flows run until an acceptable rating point was obtained. Recent approaches to OPF solution [1 - 4] are more sophisticated all point to the need of minimising a given objective function (generation cost, real power loss,) while at the same time improving system performance in terms of security and reliability of ply. In these methods, the optimal power flow problem is formulated as a mathematical mization problem and then solved, in most cases, using a non - linear programming technique LP). The non-linear programming problem has a non-linear objective function with functional ality and inequality constraints. A practical approach is to augment the objective function with equality constraints using Lagrange multipliers and then minimise the augmented objective ction using one of the optimization schemes e.g. Quadratic Programming Technique, the uential Unconstrained Minimization Technique (SUMT), the Gradient Projection Method (GPM e steepest descent method etc.

Dopazo et al [1] presented a method of minimising the generation cost by co-ordinating real reactive power allocation in the system. The procedure at first determines the real power patch based on Lagrangian multipliers, and then proceeds to optimize the reactive power cation by a gradient approach. The required gradient vector is obtained from the system real ver loss , which is the objective function.

Peschon et al [2] presented a method of minimizing the system losses by judiciously selecting reactive power injections into the system and tansformer tap settings. A suitable adjustment algorithm to drive the solution towards a feasible optimal point is presented.

Dommel and Tinney [3] developed and presented a practical non-linear optimization technique to determine the optimal power flow solution. The method minimises a non-linear objective function of production costs or losses using the Kuhn-Tucker conditions.

Gungor et al [4] formulated an algorithm that minimises the production costs by equating the total differential of the cost with respect to real and reactive power to zero. Network power equations are satisfied but voltages are not treated as essential control parameters. The algorithm also lacks logic to cater for power flow limit violations.

Some more recent techniques approach the optimal power flow problem by decomposing the general problem into real power P and reactive power Q sub-problems [19]. Here, the real power subproblem has the real power generation PG as the decision variable whereas reactive power generation and tap settings are the decision variables in the Q subproblem. The cost function is then minimised using any of the non-linear programming techniques mentioned above.

Variations of this P - Q decomposition approach solve the P and Q subproblems using a linear programming technique. Mang'oli [20], presents a solution methodology in which the P and Q subproblems are solved using a linear programming approach. Sensitivity matrices obtained from the network Jacobian matrix are used to unify the P and Q modules and linearlize the optimal power flow problem. In the P subproblem, the short term operational problem is solved, giving the optimal production costs. It uses real power generation, PG, as the decision variable. In the Q subproblem, the long term reactive power investment planning problem is solved using reactive power Q, transformer taps and voltage levels on PV buses as the decision variables.

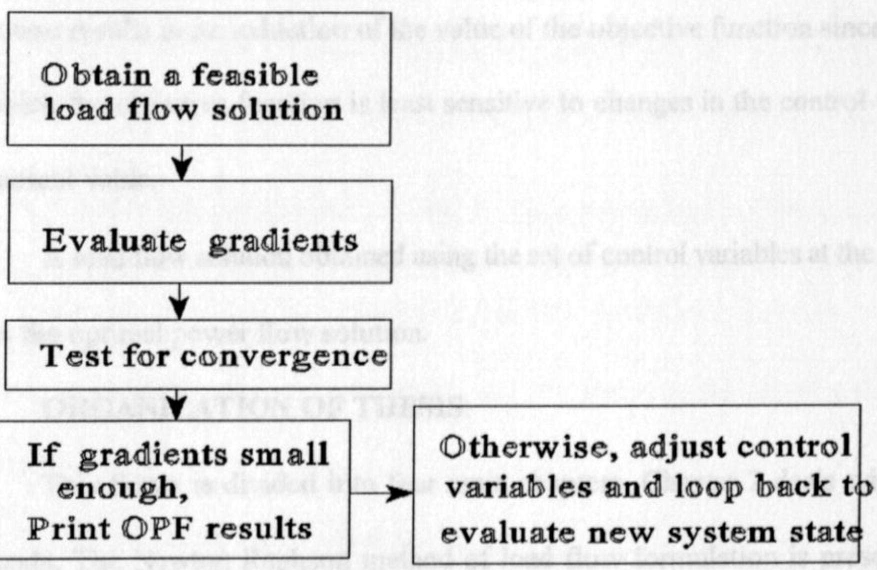
1.4. PROPOSED APPROACH TO OPF PROBLEM

Perhaps it is important to mention that the need to undertake this thesis arose as a result of realising that the Kenya Power System does not have an optimal power flow monitoring program for either the short term operational planning or the long term reactive power planning.

The solution approach adopted in this thesis is an extension of Dommel and Tinney's formulation [3] who developed and presented a practical non-linear optimization technique to determine the optimal power flow solution. The proposed solution approach recognises two types of objective functions; the operational cost objective function and the real power loss objective function, that is, both P and Q optimizations are achieved simultaneously. Formulations to cater for both equality and inequality constraints are developed and incorporated into the solution algorithm using Lagrange multipliers. Equality constraints are the load flow equations, which must be satisfied at the optimum to give a feasible operating point. Violations on both control and dependent variables form the inequality constraints and must also be satisfied at the optimum to ensure a secure and reliable operating point. Control (or decision variables) are real power generations, transformer tap settings and voltage magnitude at PV buses. Dependent variables are the resultant real and reactive power flows on transmission lines, reactive power generation from reactive generators (capacitors, inductors, synchronous condensors, etc) and voltage levels on load buses.

In the solution approach adopted in this thesis, the equality constraints (load flow or supply-demand equations) are augmented into an objective function using Lagrange multipliers and the resultant objective function minimised using the steepest descent method. Violation of limits of control variables is accounted for using the Kuhn-Tucker conditions [3]. Penalty functions are used to correct for violation on the limits of dependent variables.

The first step in the steepest descent algorithm is to obtain a feasible load flow solution using ssumed set of control variables. The Newton Raphson method of load flow formulation is loyed. Using Lagrange multipliers and the Jacobian matrix of the converged load flow solution, adient vector giving the sensitivity of the augmented objective function with respect to the rol variables is evaluated. Should a resultant dependent variable violate its limits (minimum or imum value allowed), then a penalty function is added to the augmented objective function, h serves to move the solution closer to a feasible region at the optimum.



.1 : Main steps involved in the steepest descent algorithm.

A first order gradient adjustment algorithm is then used to move the set of control variables new operating point with a lower value of objective function. This is accomplished by ucting a one dimensional search which determines a step length that moves the control variables negative direction (steepest descent) to a point with the least value of the objective function.

gradient vector at this system state again evaluated. If the gradient components are small enough (less than or equal to a specified tolerance), the solution has converged and the current set of control variables, that is, real power generation, transformer tap settings and voltage magnitudes at PV buses represents the most economical or optimal operating point with the least value of objective function (generation cost or real power loss). At the same time all the dependent variables, which actually characterise the sytem operating parameters like voltage levels on load buses, real and reactive power flows on transmission lines etc, are within the allowed operating range.

At the optimal point, any further attempt to adjust the control variables in the negative direction results in no reduction of the value of the objective function since it represents the point at which the objective function is least sensitive to changes in the control variables that affect its numerical value.

A load flow solution obtained using the set of control variables at the optimal point therefore gives the optimal power flow solution.

1.5 ORGANIZATION OF THESIS.

This thesis is divided into four main chapters. Chapter 2 deals with the load flow study methods. The Newton Raphson method of load flow formulation is presented and analysed. In Chapter 3, the optimal power flow problem is formulated and its solution using the steepest descent method presented. In Chapter 4 test results obtained using the IEEE 6 bus and 30 bus sample power systems together with those of the Kenya Power System are presented and extensively analysed. Conclusions and recommendations for future research are presented in Chapter 5 and finally, the appendices contain the main programming steps, a one dimensional search technique for obtaining the optimum step length and a listing of the FORTRAN 77 programs developed.

CHAPTER 2

LOADFLOW ANALYSIS

2.1. Introduction

A load flow study of a power system network is the steady state solution of the network. The main information obtained from this study comprises the magnitudes and phase angles of load bus voltages, reactive powers at generator buses and real and reactive power flows on transmission lines. This information is essential for the continuous monitoring of the current state of the power system and for analysing the effectiveness of alternative plans for future system expansion to meet increased load demand [5, 6, 7]. A power system network consists of bus bars inter connected by several transmission lines. Each bus bar is characterised by four parameters; real and reactive powers, P_i, Q_i, injected into the bus, voltage magnitude and phase angle, V_i, δ_i, respectively. Practical considerations allow two of the variables to be specified at each bus, and depending on which two are known, the buses can be classified into three categories:

i) PQ buses or load buses;

Here, the net real and reactive powers, P_i and Q_i injected into the bus are specified. The load flow solution then seeks to find the voltage magnitude, V_i and angle δ_i at these buses. It is important to note that at each bus;

$$P_i = P_{Gi} - P_{Di} \tag{2.1(a)}$$

$$Q_i = Q_{Gi} - Q_{Di} \tag{2.1(b)}$$

Where:

P_i, Q_i are the net real and reactive powers injected into the ith bus respectively.

P_{Gi}, Q_{Gi} are real and reactive power generations at bus i respectively.

P_{Di}, Q_{Di} are real and reactive power demands at bus i

For a pure load bus, P_{Gi} and Q_{Gi} are zero i.e. there is no power generation, meaning that

$$P_i = -P_{Di} \quad (2.2(a))$$

$$Q_i = -Q_{Di} \quad (2.2(b))$$

On the Kenya power system, such a bus would represent power demand centres like Nairobi, Nanyuki, Eldoret, Kisumu etc., with no power generating stations. PQ buses form the majority in practical power systems. Power is generated from power stations which are physically far away (due to the fact that most of our generation is by hydro generation) and transported to these centres through transmission lines.

(ii) PV buses;

At these buses, both the real power injected into the bus P_i , and the bus voltage magnitude; V_i are specified. Unknown variables are therefore the reactive power Q_i, and phase angle δ_i, at the bus. Generating stations e.g. Kiambere, Gitaru, Masinga, Kipevu, etc on the Kenya power system are mostly classified as PV buses. Such stations are scheduled to generate a specified amount of real power P_{Gi}, according to a given hydro-generation potential and the load flow solution then seeks to find the reactive power generation; Q_{Gi}, that would maintain the required voltage magnitude V_i, at several load buses. A special case of a PV bus is when there is real power generation, in which case a reactive power source e.g. a synchronous condensor, a pacitor bank etc is installed to improve the voltage profile of a power system by either creasing or increasing the bus voltage to a specified setting. A good example on the Kenya ower system is the Juja road switching station where synchronous condensors are installed to gulate the voltage in Nairobi and other surrounding load centres. In practise, only afew PV uses exist in a given power system. Power is generated at these buses and transmitted to buses eficient in power via high voltage lines.

(iii) Swing or slack bus

At this bus, which is always only one for a chosen power system, the voltage agnitude, V_i and angle δ_i are specified. Unknown variables are real and reactive power eneration P_i, Q_i respectively. A slack bus serves a special purpose in a power system. It pplies that power that could not be scheduled plus system losses which are unknown at the utset of a load flow solution. In practise, the swing bus is normally the station with the highest stalled capacity. The power demanded from the swing bus is calculated from bus voltages btained in the load flow solution. Sometimes, the figure obtained could exceed the generator ting in, which case a trial and error approach is used to reschedule generation from the other nits to offset the limit violations at the slack bus. Such a trial and error approach could be very dious and erratic and might require an operator with experience in running the system oncerned. Equipment limit violations and other security constraints such as line flows on ansmission lines, voltages at load demand centres etc. are automatically accounted for in the ormulation presented in this work.

2.2 Mathematical formulation of Load flow problem

In a system of n nodes, the complex apparent power injected into node i is,

$$S_i = P_i + jQ_i = V_i I_i^* \quad (2.3)$$

for $i = 1, 2, 3, \text{------}, n$

Where

P_i, Q_i : net real and reactive powers at the node respectively.

V_i : complex node-to-datum voltage

I_i : complex node current

$*$: complex conjugate

Taking the complex conjugate of eq. (2.3),

$$P_i - jQ_i = V_i^* I_i \quad (2.4)$$

for $i = 1, 2, 3, \text{-------}, n$

The relationship between the complex node current I_i and node-to-datum voltage V_j in a network of n nodes is given by the linear equation [5]

$$I_i = \sum_{j=1}^{n} Y_{ij} V_i \quad (2.5)$$

for $i = 1, 2, \ldots\ldots\ldots, n$

Where Y_{ij} is an element of the bus admittance matrix, an n by n matrix built from system branch impedances and line charging admittances [5, 6, 7].

Substituting equation (2.5) in equation (2.4) we get

$$P_i - jQ_i = V_i^* \sum_{j=1}^{n} Y_{ij} V_i \qquad (2.6)$$

for i = 1, 2 -----, n

In polar co-ordinates,

$$V_i = |V_i|.\, e^{j\delta i} \qquad (2.7)$$

$$Y_{ij} = |Y_{ij}|.e^{j\theta ij} \qquad (2.8)$$

Expressing V_i and Y_{ij} in this form and substituting this in equation 2.6 and equating real and imaginary parts respectively, we get the supply - demand equations below:

$$P_i = V_i . \sum_{j=1}^{n} Y_{ij} V_j \cos(\theta_{ij} + \delta_j - \delta_i) \qquad (2.9(a))$$

$$Q_i = - V_i . \sum_{j=1}^{n} Y_{ij} V_j \sin(\theta_{ij} + \delta_j - \delta_i) \qquad (2.9(b))$$

for i = 1, 2, 3, -------, n

Equations (2.9(a)) and (2.9(b)) are indeed the load flow equations, which express the relationship between the four variables, P_i, Q_i, V_i, δ_i that characterise a bus. They are non-linear equations (note the cosine and sine terms) and can only be solved iteratively. Several methods exist for solving this set of non-linear equations, like Gauss-Siedel method [5], the Newton Raphson method [5, 8], decoupled method [9, 10], fast decoupled method [10] etc. Each of these methods has its own advantages in terms of speed of computation, accuracy, reliability of convergence, computer storage requirements and ease or complexity of programming.

The Newton - Raphson method was chosen to solve the load flow problem for several reasons. It is reasonably fast, more accurate and more reliable than the other methods when applied to solve large power systems. It is also more insensitive to ill- conditioned systems (eg. series capacitance in transmission lines). The Newton - Raphson method has quadratic convergence and will always yield a solution in 3-5 iterations for practically sized systems [8]. Despite these merits, the Newton - Raphson method has some disadvantages. One, it is far more complex to formulate and program than, for example, the linearly convergent Gauss-Siedel method. In addition, the Newton-Raphson method requires more computer memory. For large systems of more than 500 buses, special programming techniques have to be adopted to improve the speed of computation and reduce computer core storage requirements. These techniques almost invariably exploit the sparsity of the bus admittance matrix [5, 8, 12] (which is reflected in the associated Jacobian matrix). Tinney and Hart [8], using optimally ordered Gaussian elimination and a special programming technique in which only the non-zero elements of the Jacobian matrix are stored in computer memory, succeeded in presenting a very efficient and fast approach that could solve problems of upto 1000 buses in approximately 52 seconds on an IBM 7040 computer and requiring only about 32KB of core memory (RAM). With the evolution of faster computers with enhanced memory capacities (e.g. VAX VMS, etc), the deficiencies of the method, that is, speed and storage requirements, can only be viewed as minor when the method is used for small and medium sized systems (50-300 buses). Infact, today, for system planning and expansion studies, these requirements are not prohibitive and solutions are possible for systems of up to 2000 buses with hundreds of components and transmission lines. It can be seen, therefore, that the advantages of this method far outweigh its short comings which can be adequately solved by incorporating special features in the programme. Most importantly, only a small modification is required to incorporate the steepest descent optimization algorithm as will be explained in a latter chapter.

2.3 The Newton-Raphson Formulation:

This method evolves from a Taylor series expansion of a multivariable function [5] and involves repeated direct solutions of a system of linear equations derived from the load flow equations (2.9) [8]. The Jacobian matrix of equation (2.9) gives the linearlized relationship between small changes in voltage angle and magnitude, $\Delta\delta_j$ and ΔV_j, and small changes in real and reactive powers, ΔP_j and ΔQ_j respectively. This is expressed as:

$$\begin{bmatrix} \Delta P \\ \cdots\cdots \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & : & N \\ \cdots\cdots & . & \cdots\cdots \\ J & : & L \end{bmatrix} \begin{bmatrix} \Delta\delta \\ \cdots\cdots \\ \Delta V \end{bmatrix} \qquad (2.10)$$

Where

ΔP: Vector of changes in real power

ΔQ: Vector of changes in reactive power

H, N, J, L: Submatrices of the Jacobian matrix

$\Delta\delta$: A vector of phase angle corrections

ΔV: A vector of voltage magnitude corrections

The submatrices H, N, J and L are derived from equations (2.9) and are defined below.

$$[H] = [\partial P/\partial \delta]$$

$$[N] = [\partial P/\partial V] \qquad (2.11)$$

$$[J] = [\partial Q/\partial \delta]$$

$$[L] = [\partial Q/\partial V]$$

i.e they represent the sensitivity of bus powers with respect to voltage and phase angles in the whole power system.

The terms ΔPi and ΔQi are residuals of equations 2.9(a) and 2.9(b) respectively as defined below.

$$\Delta Pi = Pi \text{ sched.} - Pi \text{ calc.} \qquad (2.12(a))$$

$$\Delta Qi = Qi \text{ sched.} - Qi \text{ calc.} \qquad (2.12(b))$$

for i= 1, 2, 3, ---, n $\quad i \neq$ slack bus

Where

$P_{i\ sched}$, $Q_{i\ sched}$ = real and reactive powers specified at bus i

$P_{i\ calc.}$, $Q_{i\ calc.}$ = real and reactive powers computed from equations (2.9)

For an n-bus power system, equation (2.10) can be expanded as follows:

$$
\begin{bmatrix} \Delta P1 \\ \Delta P_2 \\ \vdots \\ \vdots \\ \Delta P_n \\ \cdots\cdots \\ \Delta Q_1 \\ \Delta Q_2 \\ \vdots \\ \vdots \\ \Delta Q_n \end{bmatrix}
=
\begin{bmatrix}
\frac{\partial P1}{\partial \delta 1} & \frac{\partial P1}{\partial \delta 2} & \cdots & \frac{\partial P1}{\partial \delta n} & : & \frac{\partial P1}{\partial V1} & \frac{\partial P1}{\partial V2} & \cdots & \frac{\partial P1}{\partial Vn} \\
\frac{\partial P2}{\partial \delta 1} & \frac{\partial P2}{\partial \delta 2} & \cdots & \frac{\partial P2}{\partial \delta n} & : & \frac{\partial P2}{\partial V1} & \frac{\partial P2}{\partial V2} & \cdots & \frac{\partial P2}{\partial Vn} \\
: & : & \cdots & : & : & : & : & \cdots & : \\
: & : & \cdots & : & : & : & : & \cdots & : \\
\frac{\partial Pn}{\partial \delta 1} & \frac{\partial Pn}{\partial \delta 2} & \cdots & \frac{\partial Pn}{\partial \delta n} & : & \frac{\partial Pn}{\partial V1} & \frac{\partial Pn}{\partial V2} & \cdots & \frac{\partial Pn}{\partial Vn} \\
\cdots\cdots & \cdots\cdots & \cdots & \cdots\cdots & . & \cdots\cdots & \cdots\cdots & \cdots & \cdots\cdots \\
\frac{\partial Q1}{\partial \delta 1} & \frac{\partial Q1}{\partial \delta 2} & \cdots & \frac{\partial Q1}{\partial \delta n} & : & \frac{\partial Q1}{\partial V1} & \frac{\partial Q1}{\partial V2} & \cdots & \frac{\partial Q1}{\partial Vn} \\
\frac{\partial Q2}{\partial \delta 1} & \frac{\partial Q2}{\partial \delta 2} & \cdots & \frac{\partial Q2}{\partial \delta n} & : & \frac{\partial Q2}{\partial V1} & \frac{\partial Q2}{\partial V2} & \cdots & \frac{\partial Q2}{\partial Vn} \\
: & : & \cdots & : & : & : & : & \cdots & : \\
: & : & \cdots & : & : & : & : & \cdots & : \\
\frac{\partial Qn}{\partial \delta 1} & \frac{\partial Qn}{\partial \delta 2} & \cdots & \frac{\partial Qn}{\partial \delta n} & : & \frac{\partial Qn}{\partial V1} & \frac{\partial Qn}{\partial V2} & \cdots & \frac{\partial Qn}{\partial Vn}
\end{bmatrix}
\begin{bmatrix} \Delta\delta_1 \\ \Delta\delta_2 \\ \vdots \\ \vdots \\ \Delta\delta_n \\ \cdots\cdots \\ \Delta V_1 \\ \Delta V_2 \\ \vdots \\ \vdots \\ \Delta V_n \end{bmatrix}
\tag{2.13}
$$

Using reactangular complex arithmetic for numerical evaluation, the diagonal and off-diagonal elements of the submatrices H, N, J, and L can be evaluated as follows:

Case 1: off-diagonal elements i.e. $i \neq j$

$$H(i,j) = \partial Pi / \partial \delta j = am.fi - bm.ei \tag{2.14(a)}$$

$$N(i,j) = (\partial Pi / \partial Vj).Vj = amei + bmfi \tag{2.14(b)}$$

$$J(i,j) = \partial Qi / \partial \delta j = -(\,amei + bmfi\,) \qquad (2.14(c))$$

$$L(i,j) = (\partial Qi / \partial Vj).Vj = am.fi - bm.ei \qquad (2.14(d))$$

for $i = 1,2,3,\text{------}n$

for $j = 1,2,3,\text{------}n$

$i \neq j$

here

$$Y_{ij} = G_{ij} + B_{ij},$$

$$V_i = e_i + jf_i$$

$$(a_m + jb_m) = (G_{ij} + B_{ij})(e_j + jf_j) \qquad (2.15)$$

ase 2: Diagonal elements i.e. i = j

$$H(i,i) = \partial Pi / \partial \delta i = -Qi - Bii\,.\,|Vi|^2 \qquad (2.16(a))$$

$$N(i,i) = (\partial Pi / \partial Vi).Vi = -Pi + Bii\,.\,|Vi|^2 \qquad (2.16(b))$$

$$J(i,i) = \partial Qi / \partial \delta i = -Qi - Bii\,.\,|Vi|^2 \qquad (2.16(c))$$

$$L(i,i) = (\partial Qi / \partial Vi).Vi = -Qi - Bii\,.\,|Vi|^2 \qquad (2.16(d))$$

Vhere

Pi, Qi are the net real and reaction bus powers as calculated from equations (2.9(a)) and 2.9(b)) respectively.

and $Y_{ii} = G_{ii} + j B_{ii}$ (2.17)

2.3.1 The Newton - Raphson procedure

The Newton-Raphson solution procedure is as follows:

Step 1: An initial approximation is made of the voltage solution. In case of no other available state of the voltages, a flat start is assumed, where voltage magnitudes at all the load buses are set equal to that of the slack bus. Phase angles are assumed zero.

Step 2: Using the assumed set of bus voltages, the bus power mismatch vector is calculated from equation (2.11(a)) and (2.11(b)).

Step 3: If the power mismatches, ΔPi, ΔQi, at all buses are sufficiently small, the process is stopped and the solution printed out.

Step 4: Otherwise, the Jacobian matrix is formed from equations (2.13) and (2.15) and used to find the angle and voltage magnitude corrections from equation (2.12).

$$\begin{bmatrix} \Delta\delta \\ \cdots\cdots \\ \frac{\Delta V}{V} \end{bmatrix} = \begin{bmatrix} H & \vdots & N \\ \cdots\cdots & & \cdots\cdots \\ J & \vdots & L \end{bmatrix}^{-1} \begin{bmatrix} \Delta P \\ \cdots\cdots \\ \Delta Q \end{bmatrix} \quad (2.18)$$

Step 5: The voltage magnitudes and phase angles are updated as:

$$\delta_i^{\,k+1} = \delta_i^{\,k} + \Delta\delta_i^{\,k} \quad (2.19)$$

$$V_i^{\,k+1} = V_i^{\,k} + \Delta V_i^{\,k} \quad (2.20)$$

Vhere k = Iteration number

tep 6: Repeat from step 2 above until the power mismatch is less than specified tolerance.

Equation (2.18), can be solved using several methods. In this thesis, it was opted to use he Gaussian forward elimination and backward substitution [8, 12]. Here, the inverse of the acobian matrix, J need not be explicitly determined. This improves the efficiency and speed of omputation. A more efficient approach incorporates optimal ordering schemes that further take dvantage of sparsity characteristic of the Jacobian matrix [8] to improve speed and storage equirements. Optimal bus ordering was not implemented in this work mainly because the argeted system, the Kenya power system is a relatively small system of less than 500 buses.

.3.2 Variations of Newton Raphson methods

Decoupled load flow and fast decouped load flow methods are variations of the Newton-Raphson approach. In these formulations, advantage is taken of the loose coupling observed between P-V and Q-δ relationships in practical power systems. In this case, the sub matrices N and J are assumed zero. That is, voltage magnitudes remain constant when real power varies and he bus angle is constant when the reactive power varies.

In the decoupled load flow, equation 2.10 thus becomes:

$$\begin{bmatrix} \Delta P \\ \cdots\cdots \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & : & 0 \\ \cdots\cdots & . & \cdots\cdots \\ 0 & : & L \end{bmatrix} \begin{bmatrix} \Delta\delta \\ \cdots\cdots \\ \Delta V \end{bmatrix} \qquad (2.21)$$

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H \\ L \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta V \end{bmatrix} \qquad (2.22)$$

ast decoupled load flow, more assumptions are used to further simplify equation (2.22) with a w to improve speed of solution [10]. These variations of the Newton-Raphson method are nmonly used for fast load flow and are often used for a quick check on the operating point of a ver system if high accuracy is not required. Another advantage is that they also minimise on nputer storage requirements since the elements of submatrices J2 and J3 of the Jacobian need be stored. However, it was decided to employ the exact Newton - Raphson formulation for it nore accurate. Also, for small and medium systems like the one under study, the computation e using exact formulation compares well to the approximated one.

CHAPTER 3

OPTIMAL POWER FLOW

.1. Introduction

In the load flow problem introduced in Chapter 2, two variables are specified at each bus and he solution is then obtained for the other two variables. The specified variables are real and eactive power at PQ buses, real powers and voltage magnitudes at PV buses and voltage magniude and angle at the slack bus. In addition, tap settings of regulating transformers, if present, hould also be specified. Now, if the specified variables are allowed to vary in a region contrained by practical considerations, [upper and lower limits of real and reactive power generaions, bus voltage limits, tie line power angles, transformer taps, e.t.c.], there result an infinite umber of load flow solutions each pertaining to one set of values of specified variables. The est choice in some sense of the values of specified variables leads to the best load flow solution. he traditional approach is to find these set of values by trial and error until a suitable system tate is obtained corresponding to a given criteria. Evidently, this approach is not only tedious ut also prone to inaccuracies. In contrast, an optimal power flow has the capability to adjust oltage levels at PV buses, power outputs from generators, transformer tap positions and switchble capacitor or reactor banks to minimise the operating costs [1, 3, 4, 15], system real power osses [15] or some other appropriate objective function such as improved system security [13 , optimum load shedding [14], e.t.c.

In addition to fulfilling a certain objective, the optimal power flow ensures that both equipnent loading and system security constraints such as power output from generators, voltage s, transformer tap positions, line flows on transmission lines and reactive power outputs of citors and/ or reactors fall within practical limits. The usefulness and applicability of such a is apparent for both system planning and operating purposes. In this chapter, a step by step int of how the optimal power flow problem is formulated and solved is presented. This s considers two objective functions that feature prominently in the planning and operation of ical power systems like the one under study.

1) Real power generating cost. This is the main economic factor in power system opera- particularly if the system is predominantly thermal.

2) Real power loss minimisation. This assumes a known real power generation schedule the problem reduces to one of optimally allocating the available VAR resources to result the least real power transmission losses. Since the Kenya power system is predominantly o, it is assumed in this thesis that an appropriate hydro-generation schedule for commited o stations is known for a given load demand. The optimal power flow program is then used inimise transmission losses by optimally varying the settings of available capacitors, reactors tap changing transformers.

Mathematical formulation of the optimal power flow problem : cost objective function

he objective function to be minimised is the operating cost, which is a function of real power;

$$F(PG)= \sum_{i=1}^{NG} f_i (P_{Gi}) \qquad (3.1)$$

ject to the load flow equations

$$P_{i\ sched} - P_{icalc} = 0$$

$$(3.2)$$

$$Q_{i\ sched} - Q_{icalc} = 0$$

d operating and equipment loading constraints:

$$V_{i\ min} \leq V_i \leq V_{i\ max}, \text{ for } i = 1, 2, 3, ...,N \qquad (3.3)$$

$$P_{Gi\ min} \leq P_{Gi} \leq P_{Gi\ max}, \text{ for } i = 1, 2, 3, ...,NG \qquad (3.4)$$

$$Q_{Gi\ min} \leq Q_{Gi} \leq Q_{Gi\ max}, \text{ for } i = 1, 2, 3, ...,NG \qquad (3.5)$$

$$t_{i\ min} \leq t_i \leq t_{i\ max} \qquad \text{for } i = 1, 2, 3, ...,NT \qquad (3.6)$$

$$S_i \leq S_{i\ max} \qquad \text{for } i = 1, 2, 3, ..., NB \qquad (3.7)$$

here

F(PG) : Total operating cost

$f_i(P_{Gi})$: Cost of generating real power, P_{Gi} at unit i

$P_{i\ sched}$: Scheduled net real power at bus i

$Q_{i\ sched}$: Scheduled net reactive power

$P_{i\ calc}$: Calculated net real power

$Q_{i\ calc}$: Calculated net reactive power

V_i : bus voltage magnitude

P_{Gi} : real power generated at bus i

Q_{Gi} : reactive power generated at bus i

t_i : tap setting of transformer i

S_i : complex power flow in branch i

N : total number of buses

NG : number of generators

NT : number of tap changing transformers

NB : number of network branches in the system.

$(.)_{min,max}$: lower and upper limits respectively.

The load flow equations are equality constraints and can be expressed by the vector.

$$\mathbf{g(x,u,p)} = 0 \tag{3.8}$$

ere X: A vector of dependent variables and comprises voltage magnitudes at all PQ buses

angles at all network buses.

$$\begin{bmatrix} \mathbf{x} \end{bmatrix} = \begin{bmatrix} V \\ \delta \end{bmatrix} \tag{3.9}$$

u: A vector of control variables and consists of real and reactive power outputs of generators, voltage magnitudes of PV buses and tap settings of tap changing transformers.

$$\begin{bmatrix} \mathbf{u} \end{bmatrix} = \begin{bmatrix} P_{Gi} \\ Q_{Gi} \\ V_i \\ t_i \end{bmatrix} \tag{3.10}$$

p: vector of independent or uncontrollable variables. In this case, the slack bus power, voltage magnitude and angle at the slack bus are included in the vector P.

$$\begin{bmatrix} \mathbf{p} \end{bmatrix} = \begin{bmatrix} P_{G1} \\ Q_{G1} \\ V_1 \\ \delta_1 \end{bmatrix} \tag{3.11}$$

assuming slack bus is denoted 1

The inequality constraints (3.3) - (3.7) can be expressed as a vector:

$$[\, h(\mathbf{x}, \mathbf{u}) \,] \leq 0 \tag{3.12}$$

For example, equation (3.4) which defines limits on bus voltage magnitudes can be split into

o equations, each representing a boundary:

$$V_{i\,min} - V_i \leq 0 \tag{3.13}$$

And

$$V_i - V_{i\,max} \leq 0 \tag{3.14}$$

Similarly, all the other inequalities can be broken down into two equations as in equation

13) and (3.14).

his explains the notation adopted in equation (3.12).

The objective function thus becomes a function of **x**, **u** and **p**

$F(P_G) = F(\mathbf{x,u,p}) = F(\mathbf{x,u})$, since

P is a function of [**x, u**] (3.15)

sing the notation introduced above, the optimization problem can therefore be defined as:

Minimise

$$\underset{[U]}{F(\mathbf{x,u})} \tag{3.16}$$

Subject to the equality constraints

$$[g(\mathbf{x,u,p})] = 0 \tag{3.17}$$

And inequality constraints

$$[h(\mathbf{x,u})] \leq 0 \tag{3.18}$$

SOLUTION APPROACH:

Equations (3.16) - (3.18) represent a multivariable, non-linear objective cost function with -linear constraints and can only be solved iteratively.

For clarity, the optimization problem is first solved in the absence of inequality constraints, ch will be introduced in a latter section. Using the classical optimization method of Lagrange ltipliers [16], the minimum of F, with U as control variables,

$$\underset{[u]}{Min}\ F(\mathbf{x,u}) \tag{3.19}$$

Subject to equality constraints

$$[\ g(\mathbf{x,u,p})\ ] = 0 \tag{3.20}$$

Found by introducing as many auxilliary variables λ_i as there are equality constraints in equa- n (3.17) and minimising the unconstrained Lagrangian function:

$$L(x, u, p,\lambda) = F(x,u) + [\lambda]^T [g(x,u,p)] \tag{3.21}$$

here [λ] is a vector of Lagrange multipliers.

denotes the transpose.

om equation 3.21, the neccessary conditions for an unconstrained minimum are:

$$L/\partial X] = [\partial F/\partial X] +[\partial g/\partial X]^T.[\lambda] = 0 \tag{3.22}$$

$$L/\partial U] = [\partial F/\partial U] +[\partial g/\partial U]^T.[\lambda] = 0 \tag{3.23}$$

$$L\ /\partial\lambda] = [g(X,U,P)] =0 \tag{3.24}$$

ation (3.24) is indeed the load flow equations.

To obtain the minimum of equation (3.21), equations (3.22 - 3.24) must be satisfied. Closely mined, it is observed that equation (3.24) is the power flow equations which solves for the set dependent variables X, given a set of control variables, U. Equations (3.22) and (3.23) have as the unknown with all the other parameters known. Therefore, equation (3.22) will be used solve for $[\lambda]$ once a feasible, but not yet optimal solution point is found from equation (3.24).

$$] = -[[\partial g/\partial X]^T]^{-1}.[\partial F/\partial X] \quad (3.25)$$

The $[\lambda]$ obtained from equation (3.25) is inserted into equation (3.23) to obtain the gradient the objective function with respect to the control variables, U. Up to this point, all the equations with the exception of equation (3.23) are satisfied, and iterations are required to move the ution point to a minimum.

.1 The steepest descent method:

In this method of approach, the basic idea is to move from one feasible solution point, in the rection of steepest descent (negative gradient) to a new feasibe solution point with a lower lue of F. By repeating these moves in the direction of negative gradient, the minimum value of is found. The solution algorithm is as follows and is shown in fig. 3.1:

1) Assume an initial set of control variables, U_o

2) Solve for X by solving the power flow equations.

This yields a feasible but not yet optimal solution point, g_o, and from it the Jacobian m

$[\partial g/\partial X]$ is obtained. This is possible when the power flow equations are solved by New

aphson's method.

3) Solve for the Lagrangian multipliers, $[\lambda]$ from equation (3.25)

This only amounts to one repeat solution of a linear system for which the Jacobian

is already available from step 2. The ease with which the Newton-Raphson formulation

tates the computation of Lagrangian multipliers is one of the main reasons that influenc

choice as a method of solving the load flow equations.

4) Compute the gradient, $[\nabla F]$, by inserting the $[\ \lambda\ ]$ obtained from equation (3.26

equation (3.23).

The gradient $[\nabla F]$ measures the sensitivity of the objective function with respect to c

in U subject to the equality (or load flow equations) constraints.

5) If $[\nabla F]$ is sufficiently small, according to a specified tolerance, the minimum has

eached and current set of control variables represents the optimum.

6) Otherwise, find a new set of control parameters from

$$[U]_{new} = [U]_{old} + [\Delta U] \qquad (3.28)$$

with

$$[\Delta U] = -\alpha\ [\nabla F]$$

here

α : positive scalar that adjusts the step size [ΔU]

Go to step 2.

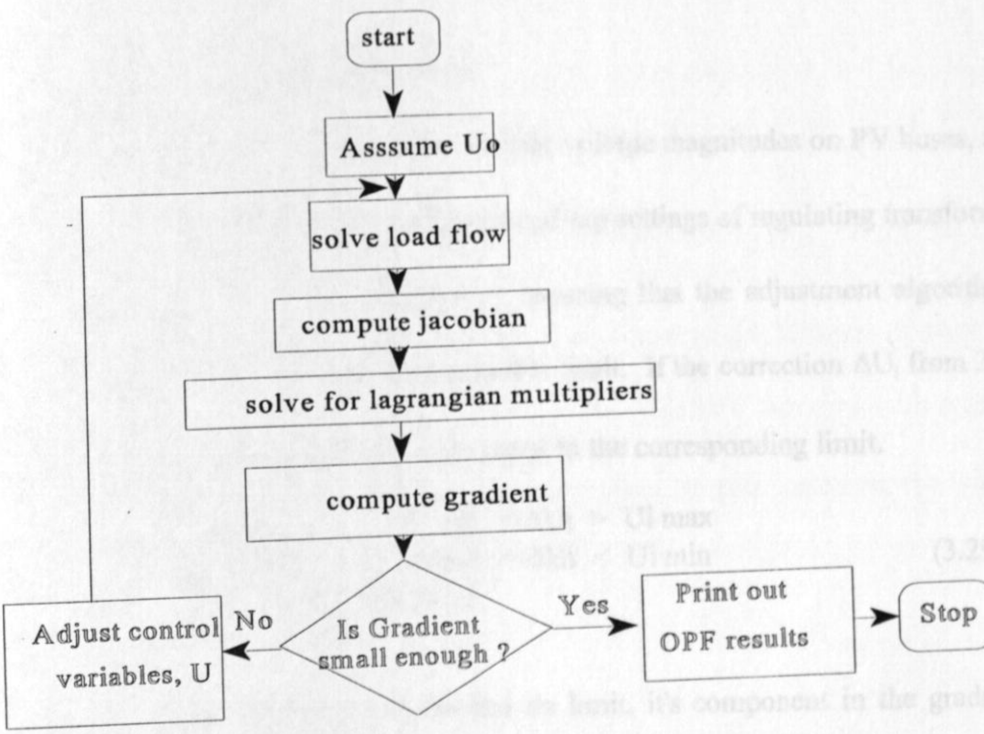


Figure 3.1 Flow chart for OPF algorithmn

In the derivation of the solution algorithm, it has so far been assumed that both the con- parameters and functional constraints can assume any value. However, practical consider- ons on equipment loading, thermal rating of transmission lines and operating levels of voltages oad buses constrain these parameters to permissible limits. In the next two sections, we shall k into how the constraints on

1) control parameters

2) functional constraints

enforced in this thesis. Alternative approaches as advocated elsewhere are given where ap-

propriate.

.2 Constraints on control parameters:

As mentioned earlier, control parameters include voltage magnitudes on PV buses, real power neration at buses with dispatchable real power and tap settings of regulating transformers.

Limits on these parameters are enforced by ensuring that the adjustment algorithm in 3.28 es not send any parameter beyond its permissible limit. If the correction ΔU_i from 3.28 would use U_i to exceed one of its limits, U_i is set equal to the corresponding limit.

$$U_{i\,new} = \begin{cases} U_{i\,max}, & \text{if } U_{i\,old} + \Delta U_i > U_{i\,max} \\ U_{i\,min}, & \text{if } U_{i\,old} + \Delta U_i < U_{i\,min} \\ U_i + \Delta U_i, & \text{otherwise} \end{cases} \qquad (3.29)$$

Even when a control parameter has reached its limit, it's component in the gradient vector ust still be computed in the following cycles because it might eventually back off the limit. hen a control parameter hits a limit, step (5) of the solution algorithm, that is checking for nvergence, is modified to comply with Kuhn-Tucker conditions [17]. At the minimum, the mponents of $[\nabla F]$ will be:

$/\partial U_i = 0, \quad \text{if} \quad U_{i\,min} < U_i < U_{i\,max}$ (3.30)

$/\partial U_i \leq 0, \quad \text{if} \quad U_i = U_{i\,max}$ (3.31)

$/\partial U_i \geq 0, \quad \text{if} \quad U_i = U_{i\,min}$ (3.32)

Previously, it has been shown [18] that the lagrangian multipliers and non-zero gradient nponents have a significant meaning at the optimum. The Lagrangian multipliers measure the sitivity of the objective function with respect to consumption P_L, Q_L or generation P_G, Q_G l hence provide a rational basis for tarrification. The non-zero gradient components of control ameters at their limits measure the sensitivity of the objective function with respect to the its U_{imin} or U_{imax} and consequently show the price being paid for imposing the limits or savs obtainable by relaxing them.

.3 Limits on functional(or dependent) variables:

Real and reactive power flows on transmission lines, voltage magnitudes on PQ (load) buses, l and reactive power generation at the swing bus and reactive power dispatch at PV buses are classified as functional constraints.

The handling of functional constraints is the principle barrier for success of most non-linear ge scale optimization algorithms. Poor handling of functional constraints can even make the ution fail to converge. In this thesis, the outside penalty method was chosen although other ethods exist for solving this involving problem [13, 17, 18]. In this approach, the objective tion is augmented by penalties whenever a functional constraint is violated, which forces the tion back sufficiently close to the constraint. The penalty method was chosen for the following reasons:

) Functional constraints are rarely rigid limits in the strict mathematical sense but are, rather, limits. For instance, $V \leq 1.0$ on a PQ node really means V should not exceed 1.0 by too h and V = 1.01 may still be acceptable. The penalty function method produces just such limits.

2) Another important consideration is that this method adds very little to the algorithm as mply amounts to adding terms to [$\partial F/\partial X$] (and also to [$\partial F/\partial U$] if the functional constraint so a function of [U]).

3) The penalty function method is also more insensitive to the initial starting point, Uo, ong as it's not unfeasible by a wide margin, in which case an optimal solution is hard to in.

With the penalty funtion method, the objective function must be replaced by

$$_{w} = F(X,U) + \Sigma W_j \tag{3.34}$$

ere a penalty term, W_j, is added for each violated functional constraint.

The design of the penalty function, W_j, is to distort the shape of the 'Constant F' contours h that the minimum of the augmented function exists in a region where all functional inequal- constraints are within their prescribed limits.

The penalty function, as used in this thesis, is defined as;

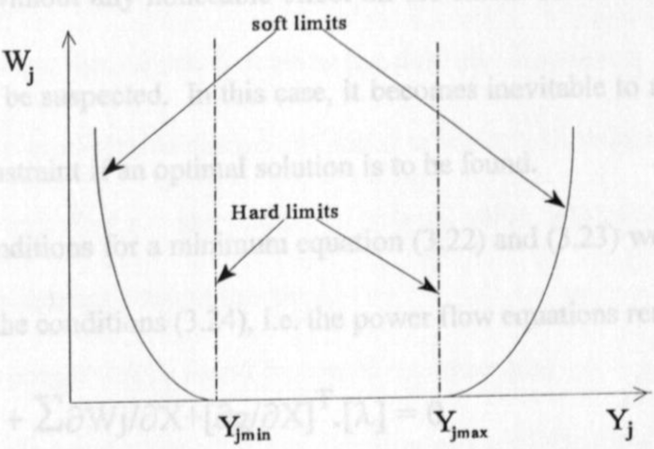
$$
j = \begin{cases} \sigma_j(y_j - y_{jmin})^2, & \text{for } y_j < y_{j\,min} \\ \sigma_j(y_j - y_{jmax})^2, & \text{for } y_j > y_{j\,max} \\ 0 & \text{Otherwise} \end{cases} \qquad (3.35)
$$

here y_j : A functional inequality constraint e.g. voltage level on a PQ bus.

σ_j : A weighting factor which determines the degree of penalty.

$(.)_{min,\,max}$: Lower and upper limits of functional constraints, respectively

g. 3.2 below shows this penalty function which replaces the rigid limit by a soft limit.



gure 3.2 Penalty function for a violated functional constraint

The use of penalty functions invariably retards the rate of convergence of the steepest descent gorithm. If the initial starting point is unfeasible by a wide margin, it may be difficult to re-over feasibility and perhaps never find an optimal solution. In some cases, a power system odel may be over constrained to the extent where Kirchoff's laws have no solution. One apach to this problem, which has been adopted in this research, is to systematically increase the

luence of the particular penalty term by increasing the weighting factor in equation (3.34). It

hown in ref [18] that if the weighting factor σ_j is increased as follows:

$$= \sigma_o . Z^l$$

ere σ_o and Z are constants and l is the number of iterations that the penalized variable y, has

lated its limit, then it is possible to improve the rate of convergence.

If σ_i grows large without any noticeable effect on the extent of violation, then an over con-

ained situation may be suspected. In this case, it becomes inevitable to relax the limits on the

ected functional constraint if an optimal solution is to be found.

The neccessary conditions for a minimum equation (3.22) and (3.23) would now be modified

given below while the conditions (3.24), i.e. the power flow equations remain unchanged.

$$L/\partial X] = [\partial F/\partial X] + \sum \partial Wj/\partial X + [\partial g/\partial X]^T.[\lambda] = 0 \qquad (3.36)$$

$$L/\partial U] = [\partial F/\partial U] + \sum \partial Wj/\partial U + [\partial g/\partial U]^T.[\lambda] = 0 \qquad (3.37)$$

As was mentioned in an earlier section, the penalty function method adds very little to the

constrained algorithm developed earlier. Equations (3.36) and (3.37) attest to this, since the

ly modification is in introducing terms related to the penalty term W_j to $[\partial F/\partial X]$ and $[\partial F\partial/U]$

his makes it relatively easy to program even when a large number of constraints are being

ndled, as is normally the case in practical power systems.

3.4 Step length determination:

With the gradient components of the objective function available as obtained fro

(3.37), the feasible direction of steepest descent [r], is formed with:

$$r_i = \begin{cases} 0, & \text{if } \nabla Fi < 0 \text{ and } U_i = U_{i\,max} \\ 0, & \text{if } \nabla Fi > 0 \text{ and } U_i = U_{i\,min} \\ -\nabla Fi, & \text{otherwise} \end{cases} \qquad (3.3$$

Then the adjustments in this direction follow from

$$\Delta U_i = \alpha . r_i$$

Where α = positive scalar that adjusts ΔUi along the descent direction r_i

It is neccessary to precisely determine the scalar α which will reduce the obj

tion by the greatest amount. Since α is a scalar, it's optimal value determination can

of as a subproblem of optimizing a scalar function.

The optimal step length can be found by one of two methods:

(1) approximate the augmented objective function as a mathematical function

late an approximate minimum [3] or

(2) conduct a one dimensional search [19].

The search technique, which was chosen in this research, has two principal adv

(1) it is extremely efficient

(2) it automatically incorporates the constraint violations.

This technique models the augmented objective function as a quadratic in ste

along the descent direction at three known points from where the optimum value of t is found. See appendix A.

Real power loss objective function.

The solution procedure outlined above solves the optimal power flow problem assuming

ost objective function. In this case it is taken that all the generators are thermal and that a

dratic cost function expressing the cost of generating a specified amount of real power can be

ained. This is repeated below for convenience:

$$_G) = \Sigma (f_i(P_{Gi}) = f_1(P_{G1}) + f_2(P_{G2}) + \ldots\ldots\ldots\ldots + f_n(P_{Gn}) \qquad (3.39)$$

where:

$F(P_G)$: Total cost of real power generation

$f_i(P_{Gi})$: Cost of generating real power , P_{Gi} , at unit i

rther, the cost of generating real power at unit i can be expressed as a quadratic function:

$$f_i(P_{Gi}) = (a_i + b_i P_{Gi} + c_i P_{Gi}^2) \qquad (3.40)$$

here a_i, b_i and c_i are cost coefficients for generator i and depend on the nature and price of the

el used to run the generator. Hydro and geothermal power stations have negligible operating

sts which in essence means that their cost coefficients are zero. On the other hand, thermal

ations are expensive to operate and have cost coefficients which determine the cost of generat-

g a specified amount of real power.

It should be recalled that the swing bus supplies both the system losses plus that power

at could not be scheduled at the start of a loaf low solution. This means that if the swing bus ower generation is minimised, then by extension the real power lost along the transmission s also minimised. This fact forms the basis of OPF solution using real power loss as the ive function.

If all the cost coefficients are set equal to zero apart from the b coefficient of the swing which is set equal to one, then the objective function reduces to

$$= P_{G1} \qquad (3.41)$$

ion (3.41) serves as a suitable dummy for real power loss minimization because minimising ject to both the equalty and inequality constraints as detailed in section 3.3 of this chapter to an optimal operating point with the least loss of real power. Switching from cost optimi- to real power loss minimization requires only a minor alteration of data input to the opti- ower flow program developed in this work. This easy transition is found extremely suitable situation where the real power generation for all the generators is known in advance (in dance with a specified generation policy) and the objective is to optimise on the available sources e.g. capacitors, transformer tap settings, synchronous condensors, etc. to minimise mission losses and obtain a good voltage profile.

In case of the Kenya Power System, generation is predominantly hydro, where op- onal costs are negligible. The OPF problem thus reduces to one of optimally allocating the able Var sources to minimise transmission line real power loss and at the same time main- a good voltage profile.

CHAPTER 4

TEST RESULTS AND ANALYSIS

A close examination of the preceding chapter reveals that the OPF algorithm is basically a series of transitions from one load flow solution to another via the gradient steps. The overall performance of the algorithm in terms of speed of solution, accuracy of results, reliability of convergence etc. therefore relies to a large extent on the performance of the load flow solution. It is as a result of this that the portion of the algorithm that computes load flow solutions received significant attention. A detailed description of the computer program developed is presented in Appendix B.

The following sample power systems were chosen for testing the optimal power flow program :

i) IEEE 6 bus power system [20]

ii) IEEE 30 bus power system [13]

The choice of these systems was not arbitrary. They were carefully selected so that comparison can be made to studies carried out by other researchers elsewhere. Similar test systems have been set aside by the IEEE Power System Working Committee like the IEEE 57 bus, IEEE 118 bus etc. These standard systems have been chosen carefully so that different researchers carrying out different power system studies have a common base to compare their results. It was however, not possible to obtain system data for the 57 and 118 bus systems. Despite this, it was felt that the available 6 bus and 30 bus systems would serve as sufficient test examples.

It should be noted that no two power systems are exactly similar in size, layout, length of mission lines, loading etc. The program would undoubtedly perform differently when applied olve different power systems, though similar in some aspects like size (number of buses), erators, etc. In some cases an optimal operating point could even be impossible to obtain ending on the loading condition of the system being solved. It is therefore necessary to apply the gram to a wide range of test systems of various layouts, sizes and loadings before one can rtain its reliability as a computational aid. All these aspects were borne in mind when subjecting program to tests. Sufficient attention was paid to the following performance criteria of the OPF gram:

i) Reliability of convergence.

ii) Speed of solution.

ll the test cases presented below, these two aspects of performance evaluation are well illustrated n in tabular and graphical form and where appropriate explanations advanced to clarify the ervations noted.

TEST RESULTS FOR IEEE 6 BUS SYSTEM

1 DESCRIPTION OF THE 6 BUS SYSTEM

Figure 4.1 below shows the single line diagram of the 6 bus system. This small system sists of 7 transmission lines with two tap setting transformers on lines 4 and 7 which are used to trol (or regulate) voltages at the load buses 5 and 3 respectively. There are two 100 MW, 100 ar generators connected at buses 1 and 2. Buses 3, 4, 5 and 6 are load buses with an assumed k demand of 135 MW and 36 Mvar of real and reactive power respectively. Two shunt capacitors installed at load buses 4 and 6 to control voltages at their respective buses in case the two nerators fail to dispatch sufficient reactive power to maintain a good voltage profile. For nsmission lines, loading etc. The program would undoubtedly perform differently when applied solve different power systems, though similar in some aspects like size (number of buses), nerators, etc. In some cases an optimal operating point could even be impossible to obtain pending on the loading condition of the system being solved. It is therefore necessary to apply the ogram to a wide range of test systems of various layouts, sizes and loadings before one can certain its reliability as a computational aid. All these aspects were borne in mind when subjecting e program to tests. Sufficient attention was paid to the following performance criteria of the OPF ogram:

i) Reliability of convergence.

ii) Speed of solution.

all the test cases presented below, these two aspects of performance evaluation are well illustrated oth in tabular and graphical form and where appropriate explanations advanced to clarify the oservations noted.

1 TEST RESULTS FOR IEEE 6 BUS SYSTEM

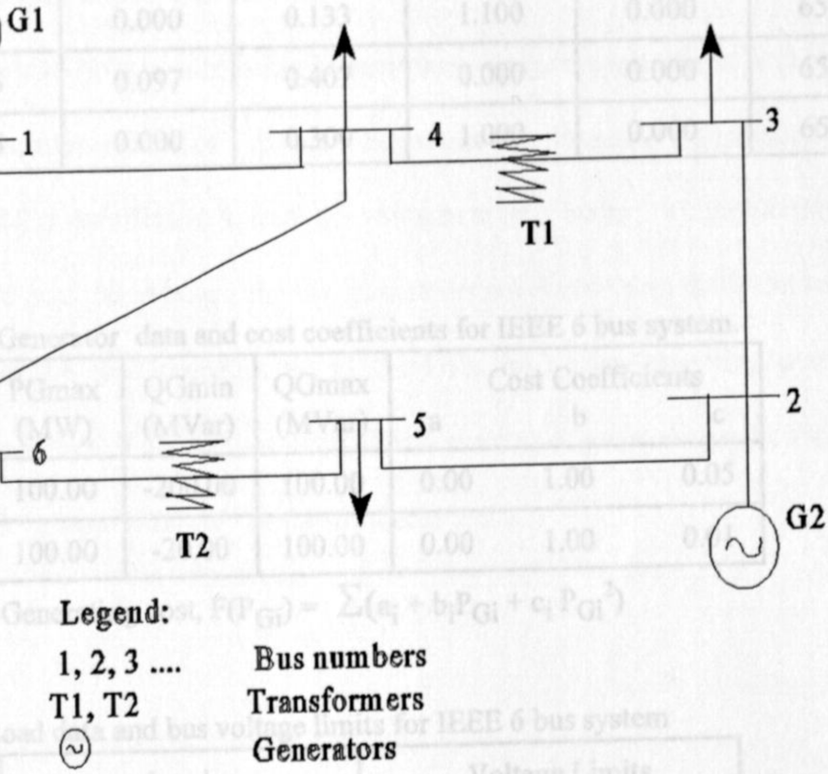
1.1 DESCRIPTION OF THE 6 BUS SYSTEM

Figure 4.1 below shows the single line diagram of the 6 bus system. This small system onsists of 7 transmission lines with two tap setting transformers on lines 4 and 7 which are used to ontrol (or regulate) voltages at the load buses 5 and 3 respectively. There are two 100 MW, 100 Ivar generators connected at buses 1 and 2. Buses 3, 4, 5 and 6 are load buses with an assumed eak demand of 135 MW and 36 Mvar of real and reactive power respectively. Two shunt capacitors re installed at load buses 4 and 6 to control voltages at their respective buses in case the two enerators fail to dispatch sufficient reactive power to maintain a good voltage profile. For been chosen to serve as the swing bus.

low shows line and transformer data for the 6 bus system. Generator data

r cost coefficients are presented in Table 4.1(b). Load data and bus voltage

Table 4.1(c)



gram for the IEEE 6 bus system*

Table 4.1 (a) : Line and transformer data for IEEE 6 bus system

Line No.	Bus Nos.	Line Impedance R (p.u)	X (p.u)	Tap setting (p.u)	Line char B (p.u)	Rating (MVA)
1	1 - 4	0.080	0.370	0.000	0.000	65.00
2	1 - 6	0.123	0.518	0.000	0.000	65.00
3	2 - 3	0.723	1.050	0.000	0.000	65.00
4	2 - 5	0.282	0.640	0.000	0.000	65.00
5	3 - 4	0.000	0.133	1.100	0.000	65.00
6	4 - 6	0.097	0.407	0.000	0.000	65.00
7	5 - 6	0.000	0.300	1.000	0.000	65.00

Table 4.1 (b) : Generator data and cost coefficients for IEEE 6 bus system.

Bus No.	PGmin (MW)	PGmax (MW)	QGmin (MVar)	QGmax (MVar)	Cost Coefficients a	b	c
1	0.00	100.00	-200.00	100.00	0.00	1.00	0.05
2	0.00	100.00	-20.00	100.00	0.00	1.00	0.01

Generating cost, $F(P_{Gi}) = \sum(a_i + b_i P_{Gi} + c_i P_{Gi}^2)$

Table 4.1 (c) Load data and bus voltage limits for IEEE 6 bus system

Bus No.	Load (MW)	(MVar)	Voltage Limits Vmin (p.u)	Vmax (p.u)
1	0.00	0.00	1.00	1.15
2	0.00	0.00	1.00	1.15
3	55.00	13.00	0.95	1.05
4	0.00	0.00	0.95	1.05
5	30.00	18.00	0.95	1.05
6	50.00	5.00	0.95	1.05

4.1.2 Load flow results for IEEE 6 bus system .

The initial load flow is run with no compensation for load buses 3, 4, 5 and 6 apart from reactive power generation at buses 1 and 2. The tap setting transformers on lines 4 and 7 are set at their initial values of 1.025 and 1.1 p.u respectively. Table 4.1 (d) below shows the load flow solution which results from this system state.

Examining the load flow results, it can be seen that voltage magnitudes at load buses 3, 4, 5 and 6 are all below their lower limit of 0.95 p.u .This means that the reactive generation obtained from generators 1 and 2 is insufficient to hold the voltages at load buses 3 - 6 within the allowed limits of 0.95 and 1.05 p.u. So although the two generators are dispatching sufficient real power (147.86 MW) to meet the load requirements (135 MW), the system operating point is still unacceptable due to violation of the voltage limits. Extra reactive sources are required to regulate the voltages at buses 3 through 6 to fall within allowable limits.

Table 4.1 (d) Initial load flow results for IEEE 6 bus system

lo.	Generation MW	Generation MVar	Voltage | V | (p.u)	Phase angle δ (°)
	97.8600	57.4600	1.0000	0.0000
	50.0000	22.9900	1.0000	-4.1770
	0.0000	0.0000	0.8356	-15.3850
	0.0000	0.0000	0.8575	-11.2260
	0.0000	0.0000	0.8063	-14.8380
	0.0000	0.0000	0.8359	-14.2630

tal System Generation (MW)	147.860
Total System Losses (MW)	12.860
otal Generation Cost ($ / Hr)	651.690
No. of Load Flow iterations	3

Fig 4.1 (e) below shows the real and reactive power flows as well as real and reactive power along the seven (7) transmission lines interconnecting the six (6) buses. If a transmission line ying power in excess of its MVA rating, thermal failure may occur due to excessive heating conductors. From Table 4.1 (e), it can be seen that no transmission line is carrying power in ion of its MVA rating.

Further, it can be noted from the same table that different lines are differently loaded. From vation, this depends on the nature of the buses that they interconnect and also on their line lances. For example, line number 1 that interconnects buses 1(a generator bus) and 4 (a load carries 60.89 MVA of apparent power as compared to line number 7 that connects bus 5 (a ous) to bus 6 (a load bus) which carries only 8.27 MVA of apparent power.

NB: apparent power refers to the vector sum of real and reactive powers.

s expected since as explained in chapter 2, load buses, normally deficient in power, obtain generating stations via transmission lines and hence the heavy power flow between two such a power system. In contrast, load buses exchange very little power between themselves in rk because they are infact demand centres in need of power. This kind of information, as d from load flow analysis during the planning stage of a power system, is vital in determining gn specifications of transmission lines.

Table 4.1 (e) real and reactive power flow and line losses for IEEE 6 bus system

o.	Line Flows			Line Losses	
o.	Real (MW)	React.(MVAR)	Total(MVA)	(MW)	(MVAR)
	51.98	31.71	60.89	2.97	13.72
	45.88	25.75	52.61	3.40	14.34
	18.52	4.43	19.04	2.62	3.81
	31.49	18.57	36.55	3.77	8.55
	-15.90	-0.62	15.91	2.62	3.81
	-39.07	-12.37	40.98	0.00	3.20
	-49.01	-17.99	52.21	2.97	13.72
	39.07	15.57	42.06	0.00	3.20
	9.91	2.43	10.21	0.14	0.58
	-27.72	-10.02	29.47	3.77	8.55
	-2.25	-7.96	8.27	0.00	0.32
	-42.47	-11.41	43.98	3.40	14.34
	-9.77	-1.85	9.95	0.14	0.58
	2.25	8.27	8.58	0.00	0.32

The cost of generating real power, as explained in chapter 3 of this report, is an important
eration in power system operation. The hourly cost of generating real power for the 6 bus
operating in the state as shown in Table 4.1(d) is $651.69. In practise, attempts should be
to lower this figure while at the same time maintaining an acceptable operating point where
perational and equipment loading constraints are satisfied. The traditional approach was by
nd error where several operating points (selected by varying real power generation, voltage
buses, tap positions on regulating transformers etc) were chosen until a reasonably good
state was found. This was tedious and prone to inaccuracies.

In the next section, it is shown how the OPF program developed is used to obtain an optimal
ing point which not only requires the least cost of generating the real power but also maintains
ing constraints like voltage limits, power flow on transmission lines, reactive power generation
quipment loading contraints within preselected limits.

OPF results for the IEEE 6 bus system .

From an initial operating point as shown in Table 4.1(d), the optimal power flow program
voked taking recognition of shunt capacitors on buses 4 and 6 and also the two regulating
formers on lines 4 and 7. The main column 2 of Table 4.1(f) shows the OPF results obtained
the steepest descent method as developed in this work. Main column (3) of the same table
s the OPF results for the same system obtained using a linear programing approach as
loped by Mang'oli [20].

From an initial generating cost of $651.69 per hour, the OPF program automatically transits
one system state to another (by adjusting the control parameters) to an optimal operating point

ith a generating cost of \$393.76 per hour representing a net saving of 39.57 %. At the optimal perating point, generators 1 and 2 are set to dispatch 53.05 MW and 100 MW of real power spectively. Their combined generation of 153.25 MW is sufficient to to serve the 135 MW system ad. The two shunt capacitors at load buses 4 and 6 are automatically set to dispatch 27.79 and 5.63 MVAR of reactive power respectively. The transformers on lines 4 and 7 are adjusted to 0.97 nd 1.1 p.u respectively. All the load bus voltages are now within the desired operating range of 0.95 nd 1.05 p.u. with the lowest of 0.9767 p.u. registered at bus 5. These results compare well with hose on main column 2 which are obtained using a different approach, the linear programming method.

Here, the cost of generation at the optimal operating point is \$395.776 per hour, a difference f 0.51 % as compared to that obtained using the steepest descent method.

Table 4.1 (f) Optimal Power Flow results for IEEE 6 bus system

	OPF results using steepest descent				OPF results using Linear Programming			
Bus No.	Generation MW	MVar	Voltage | V | (p.u)	Angle δ (º)	Generation MW	MVar	Voltage | V | (p.u)	Angle δ (º)
1	53.05	4.21	1.075	0.000	53.37	18.08	1.000	0.000
2	100.00	9.91	1.150	15.408	100.00	23.74	1.150	13.600
3	0.00	0.00	1.006	-7.039	0.00	0.00	0.900	-8.450
4	0.00	27.79	1.045	-5.306	0.00	15.00	0.957	-6.180
5	0.00	0.00	0.977	-2.634	0.00	0.00	0.952	-3.760
6	0.00	35.63	1.050	-6.340	0.00	15.50	0.946	-7.480
Taps	Line No.		Tap setting		Line No.		Tap setting	
	4		0.9156		4		0.938	
	7		1.005		7		1.100	
System Generation(MW)			153.050		153.368			
System Losses (MW)			18.050		18.368			
Gen Cost ($ / Hr)			393.760		395.776			
No. of gradient adjustments			10		12			

Figures 4.2, 4.3, 4.4 and 4.5 below show the comparison of optimal system parameters obtained for the IEEE 6 bus system using both the steepest descent method and the linear programming method. These results are as presented in table 4.1(f) above.

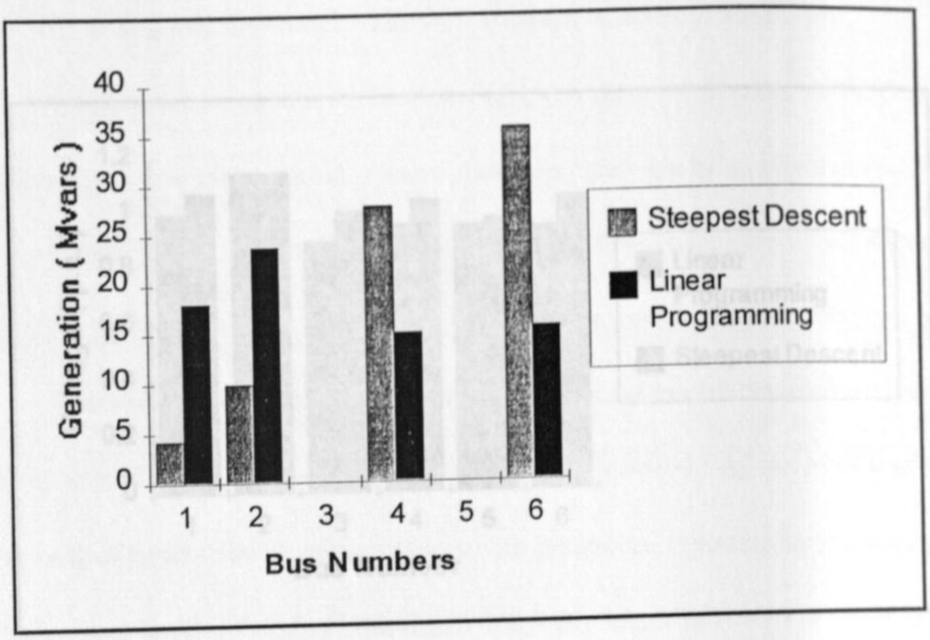


Fig 4.2: Optimal reactive power generation for the IEEE 6 bus system

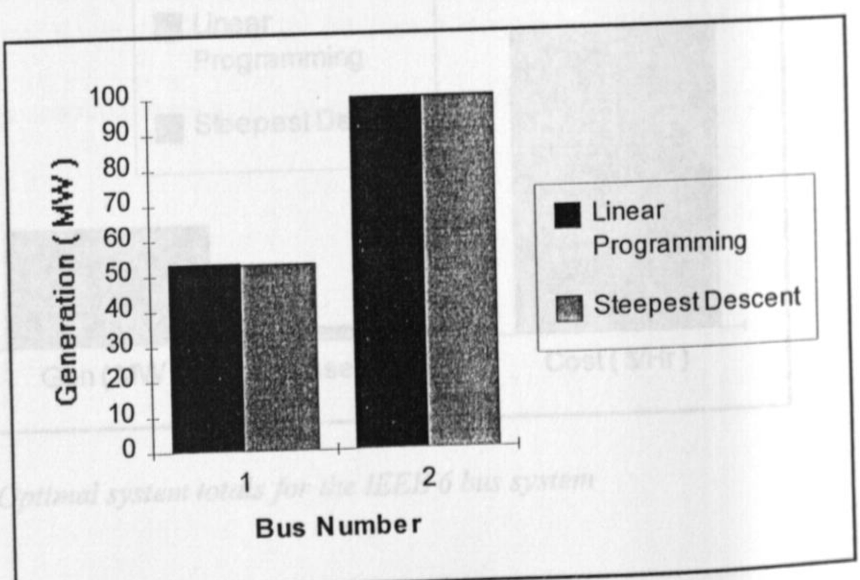
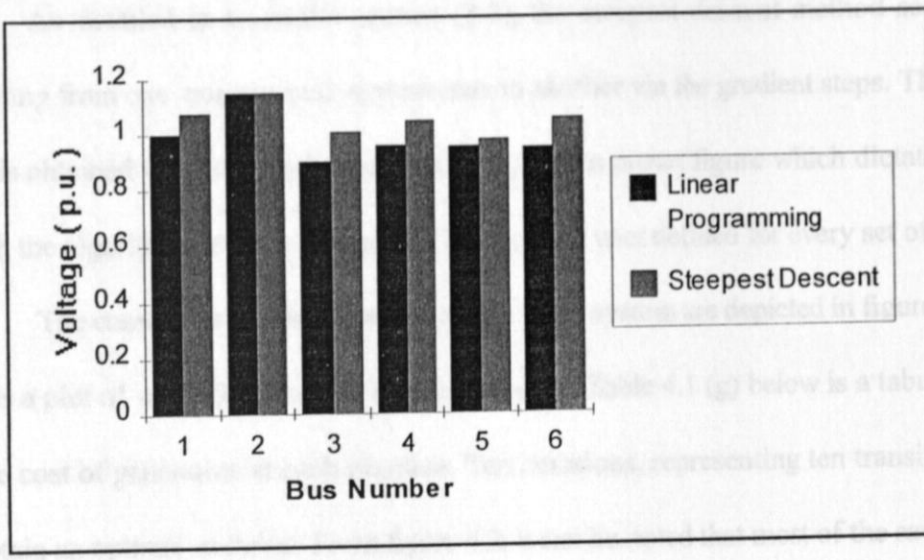
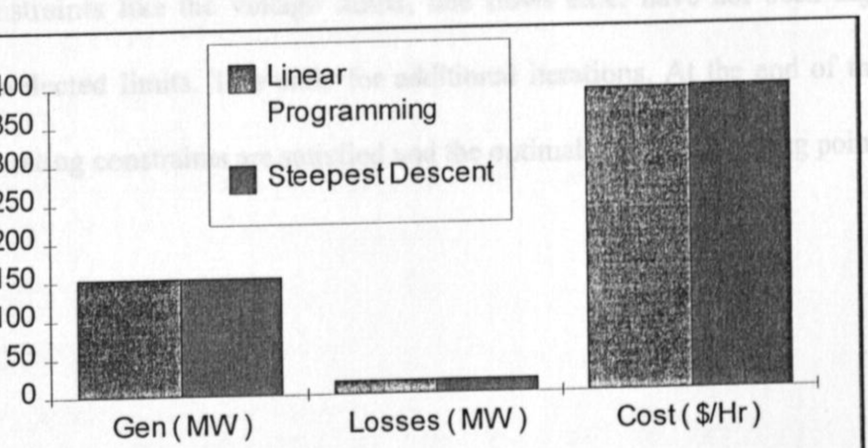


Fig 4.3 : Optimal real power generation at buses 1 and 2 for the IEEE 6 bus system



4.4 : *Optimal Bus voltage levels for the IEEE 6 bus system.*



4.5 : *Optimal system totals for the IEEE 6 bus system*

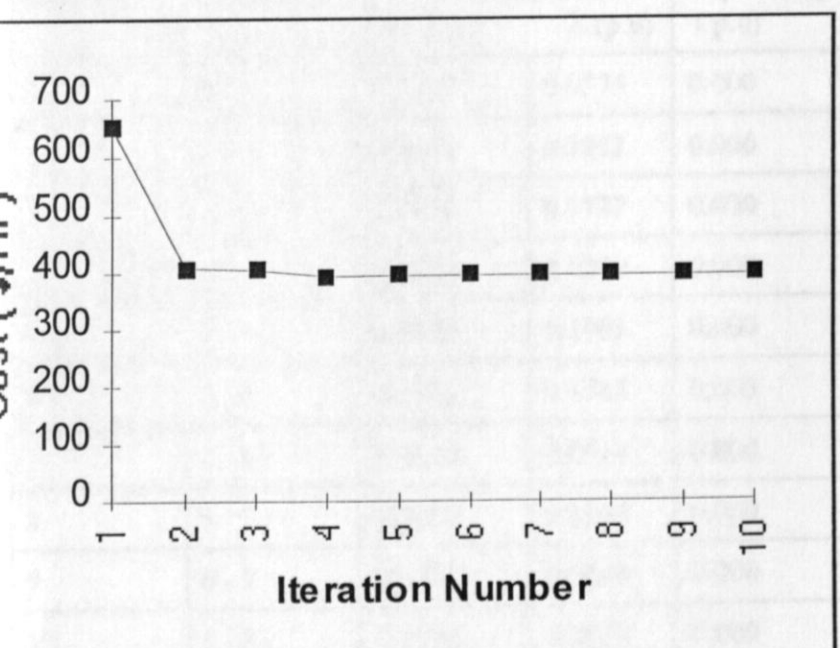
Convergence characteristics of the steepest descent algorithm.

As detailed in an earlier section (3.3), the steepest descent method seeks a solution by ...siting from one non-optimal system state to another via the gradient steps. The optimal system ... is obtained when the gradient is less or equal to a preset figure which dictates the accuracy to ...ch the algorithmn must converge to. This figure is user defined for every set of control variables.

The convergence characteristics for the 6 bus system are depicted in figure 4.2 below which ...ws a plot of cost ($/Hr) versus iteration number. Table 4.1 (g) below is a tabular representation ...e cost of generation at each iteration. Ten iterations, representing ten transitions, are required ...btain an optimal solution. From figure 4.2, it can be noted that most of the savings are obtained ...he first two iterations; from $ 651.69 per hour to $408.10 per hour, a 37.38 % saving in ...eration cost. But although the minimum cost has almost been obtained, other operating ...straints like the voltage limits, line flows e.t.c. have not been adjusted to fall within the selected limits. This calls for additional iterations. At the end of the tenth iteration, all the ...rating constraints are satisfied and the optimal system operating point obtained.

ble 4.1 (g) convergence results for IEEE 6 bus system

eration No.	Cost ($ / Hour)
	651.69
	408.10
	403.91
	392.39
	393.40
	393.79
	393.78
	393.71
	393.77
0	393.76



g 4.6: Convergence characteristics of the IEEE 6 bus system using the steepest descent method

4.2 TEST RESULTS FOR THE IEEE 30 BUS SYSTEM

4.2.1 Description of the IEEE 30 bus system.

There are 40 transmission lines with 4 under load tap changing trasformers (ULTC) to control voltages at load buses 9, 10, 12 and 27. In addition, there are 6 shunt capacitors at load buses 15, 17, 20, 21, 24 and 29 to be dispatched in order to improve the voltage profile should need arise. Line and transformer data is presented in Table 4.2 (a).

The test system consists of six generators with a total installed capacity of 425 MW of real power and 550 MVAR of reactive power. The generator data and cost coefficients are given in Table 4.2 (b). Load data is shown in Table 4.2 (c).

Table 4.2 (a) Line and transformer data for IEEE 30 bus system

Line No.	Bus No.s	Line Impedance R (p.u)	X (p.u)	Tap Setting (p.u)	Line Char B (p.u)	Rating (MVA)
1	1 - 2	0.0192	0.0575	0.000	0.000	130
2	1 - 3	0.0452	0.1852	0.000	0.000	130
3	2 - 4	0.0570	0.1737	0.000	0.000	65
4	3 - 4	0.0132	0.0379	0.000	0.000	130
5	2 - 5	0.0472	0.1983	0.000	0.000	130
6	2 - 6	0.0581	0.1763	0.000	0.000	65
7	4 - 6	0.0119	0.0414	0.000	0.000	90
8	5 - 7	0.0460	0.1160	0.000	0.000	70
9	6 - 7	0.0267	0.0820	0.000	0.000	130
10	6 - 8	0.0120	0.0420	0.000	0.000	32
11	6 - 9	0.0000	0.2080	0.000	0.000	65
12	6 - 10	0.0000	0.5560	0.000	0.000	32
13	9 - 11	0.0000	0.2080	0.000	0.000	65

4	9 - 10	0.0000	0.1100	0.000	0.000	65
5	4 - 12	0.0000	0.2560	0.000	0.000	65
6	12 - 13	0.0000	0.1400	0.000	0.000	65
7	12 - 14	0.1231	0.2559	0.000	0.000	32
8	12 - 15	0.0662	0.1304	0.000	0.000	32
9	12 - 16	0.0945	0.1987	0.000	0.000	32
0	14 - 15	0.2210	0.1997	0.000	0.000	16
21	16 - 17	0.0824	0.1932	0.000	0.000	16
22	15 - 18	0.1070	0.2185	0.000	0.000	16
23	18 - 19	0.0639	0.1292	0.000	0.000	16
24	19 - 20	0.0340	0.0680	0.000	0.000	32
25	10 - 20	0.0936	0.2090	0.000	0.000	32
26	10 - 17	0.0324	0.0845	0.000	0.000	32
27	10 - 21	0.0348	0.0749	0.000	0.000	32
28	10 - 22	0.0727	0.1499	0.000	0.000	32
29	21 - 22	0.0116	0.0236	0.000	0.000	32
30	15 - 23	0.1000	0.2020	0.000	0.000	16
31	22 - 24	0.1150	0.1790	0.000	0.000	16
32	23 - 24	0.1320	0.2700	0.000	0.000	16
33	24 - 25	0.1885	0.3292	0.000	0.000	16
34	25 - 26	0.2544	0.3800	0.000	0.000	16
35	25 - 27	0.1093	0.2087	0.000	0.000	16
36	28 - 27	0.0000	0.3960	0.000	0.000	65
37	27 -29	0.2198	0.4153	0.000	0.000	16
38	27 - 30	0.3202	0.6027	0.000	0.000	16
39	29 - 30	0.2399	0.4533	0.000	0.000	16

8 - 28	0.0636	0.2000	0.000	0.000	32

Table 4.2 (b) Generator data and cost coefficients for IEEE 30 Bus system

PGmin (MW)	PGmax (MW)	QGmin (MVar)	QGmax (MVar)	Cost Coefficients a	b	c
50.00	200.00	-20.00	200.00	0.0	2.00	0.0037
20.00	70.00	-20.00	100.00	0.0	1.75	0.0175
15.00	50.00	-15.00	80.00	0.0	1.00	0.0625
10.00	35.00	-15.00	60.00	0.0	3.25	0.0083
10.00	30.00	-10.00	50.00	0.0	3.00	0.025
12.00	40.00	-15.00	60.00	0.0	3.00	0.025

Generating cost, $F(P_{Gi}) = \sum(a_i + b_i P_{Gi} + c_i P_{Gi}^2)$

Table 4.2 (c) Load data and bus voltage limits for IEEE 30 bus system

Load (MW)	(MVar)	Voltage limits Vmin (p.u)	Vmax (p.u)
0.00	0.00	0.95	1.15
21.70	12.70	0.95	1.15
2.40	1.20	0.95	1.15
7.60	1.60	0.95	1.15
94.20	19.00	0.95	1.15
0.00	0.00	0.95	1.15
22.80	10.90	0.95	1.15
30.00	30.00	0.95	1.15
0.00	0.00	0.95	1.15
5.80	2.00	0.95	1.15
0.00	0.00	0.95	1.15

11.20	7.50	0.95	1.15
0.00	0.00	0.95	1.15
6.20	1.60	0.95	1.15
8.20	2.50	0.95	1.15
3.50	1.80	0.95	1.15
9.00	5.80	0.95	1.15
3.20	0.90	0.95	1.15
9.50	3.40	0.95	1.15
2.20	0.70	0.95	1.15
17.50	11.20	0.95	1.15
0.00	0.00	0.95	1.15
3.20	1.60	0.95	1.15
8.70	6.70	0.95	1.15
0.00	0.00	0.95	1.15
3.50	2.30	0.95	1.15
0.00	0.00	0.95	1.15
0.00	0.00	0.95	1.15
2.40	0.90	0.95	1.15
10.60	1.90	0.95	1.15

2 Initial Load Flow results for the IEEE 30 Bus system

Table 4.2 (d) below shows the initial load flow results obtained with no reactive power
pensation for load buses 15, 17, 20, 21, 24 and 29. The four regulating transformers on buses
0, 12 and 27 are set at their initial values of 1.0 p.u. It can be noted from Table 4.2 (d) that the

magnitudes at load buses 14 through 30 are all below the lower operating limit of 0.95 p.u.

licates that the reactive power generated from the six units at buses 1, 2, 5, 8, 11 and 13 is

quate. Additional reactive power is therefore required to solve this limit violation and

the voltage profile.

Table 4.2 (d) Initial Load Flow results for IEEE 30 bus system

	Generation		Voltage	Phase angle
o.	MW	MVar	|V| (p.u)	δ (°)
	99.5800	-22.9100	1.0000	0.0000
	80.0000	14.9600	1.0000	-2.1404
	0.0000	0.0000	0.9885	-4.4370
	0.0000	0.0000	0.9856	-5.3160
	50.0000	43.7400	1.0000	-7.6520
	0.0000	0.0000	0.9877	-6.2720
	0.0000	0.0000	0.9841	-7.3730
	20.0000	68.7900	1.0000	-6.7990
	0.0000	0.0000	0.9731	-7.8127
	0.0000	0.0000	0.9525	-10.0160
	20.0000	13.3700	1.0000	-5.3630
	0.0000	0.0000	0.9726	-9.1890
	20.0000	19.8400	1.0000	-7.5400
	0.0000	0.0000	0.9554	-10.2540
	0.0000	0.0000	0.9493	-10.3480
	0.0000	0.0000	0.9562	-9.8570
	0.0000	0.0000	0.9478	-10.2160

18	0.0000	0.0000	0.9371	-11.0600
19	0.0000	0.0000	0.9333	-11.2608
20	0.0000	0.0000	0.9372	-11.0170
21	0.0000	0.0000	0.9389	-10.5640
22	0.0000	0.0000	0.9395	-10.5510
23	0.0000	0.0000	0.9357	-10.8270
24	0.0000	0.0000	0.9267	-11.0570
25	0.0000	0.0000	0.9345	-11.2220
26	0.0000	0.0000	0.9152	-11.5180
27	0.0000	0.0000	0.9489	-10.9990
28	0.0000	0.0000	0.9844	-6.7950
29	0.0000	0.0000	0.9274	-12.4310
30	0.0000	0.0000	0.9149	-13.4620
Total System generation (MW)			289.5800	
Total System losses (MW)			5.8900	
Total Generation Cost ($ / Hr)			900.3000	
No. of Load flow iterations			3	

4.2.3 Optimal Power Flow results for the IEEE 30 bus system

Optimal power flow results using the steepest descent method are presented in Table 4.2 (e). They are compared with OPF results for the same system obtained using the linear programming method as documented in reference 20.

From the initial system state as shown in Table 4.2(d), the operating cost of $900.36 per hour is improved to an optimal operating cost of $802.65 per hour for the steepest descent method and

).315 per hour for the linear programming method. This represents a net saving of 13 % and 7.5 or the steepest descent and linear programming methods respectively. The shunt capacitors on pensated load buses 15, 17, 20, 21, 24 and 29 are set to optimally dispatch 3.72, 0.39, 1.61, 3.56, and 3.53 Mvar of reactive power respectively. The under load tap changing transformers on s 11, 12, 15 and 36 are adjusted to 1.05, 0.965, 1.047 and 1.00 p.u respectively. All the load bus ages are within the operating range of 0.95 - 1.05 p.u respectively.

The steepest descent method converges to a lower operating cost than the linear programming hod. This could be explained by the fact that the steepest descent formulation developed in this k was exact, with no approximations made in the solution approach. On the other hand, the ar programming method uses several assumptions in its optimisation approach with a view to roving solution time. Consequently, the steepest descent method takes more iterations (20) to ain a solution as compared to 12 iterations required in the linear programming method.

It can be deduced therefore, that the steepest descent method is more suitable than the linear gramming method if very accurate results are desired. On the other hand the fast converging ar programming method is more appropriate in real time applications, where fast OPF solutions neccessary.

.2 (e) Optimal Power Flow results for IEEE 30 bus system

F results using steepest descent				OPF results using Linear Programming			
eneration		Voltage	Angle	Generation		Voltage	Angle
	MVar	| V | (p.u)	δ (º)	MW	MVar	| V | (p.u)	δ (º)
5	-17.37	1.050	0.000	235.60	6.05	1.100	0.000
7	29.51	1.039	-3.740	20.00	100.00	1.083	-4.730
0	0.00	1.028	-5.816	0.00	0.00	1.041	-6.020
0	0.00	1.023	-6.996	0.00	0.00	1.029	-7.270
7	32.10	1.015	-10.600	10.00	35.61	1.032	-11.820
0	0.00	1.021	-8.151	0.00	0.00	1.015	-8.600
0	0.00	1.010	-9.672	0.00	0.00	1.013	-10.410
26	43.30	1.024	-8.504	10.00	33.94	1.014	-9.100
0	0.00	1.036	-10.251	0.00	0.00	1.018	-11.080
00	0.00	1.011	-12.133	0.00	0.00	0.998	-13.190
54	34.14	1.100	-8.929	10.00	-10.00	0.951	-10.640
00	0.00	1.031	-11.528	0.00	0.00	0.972	-11.680
00	29.46	1.069	-10.655	12.00	-13.70	0.951	-10.640
00	0.00	1.014	-12.455	0.00	0.00	0.965	-12.830
00	3.72	1.008	-12.508	0.00	5.79	0.968	-13.240
00	0.00	1.015	-12.067	0.00	0.00	0.980	-12.810
00	0.39	1.006	-12.333	0.00	15.75	0.996	-13.520
00	0.00	0.996	-13.115	0.00	0.00	0.970	-14.130
00	0.00	0.993	-13.272	0.00	5.00	0.975	-14.430
00	1.61	0.996	-13.048	0.00	5.99	0.983	-14.270

21	0.00	0.00	0.997	-12.608	0.00	16.63	0.992	-13.910
22	0.00	3.51	0.998	-12.592	0.00	0.00	0.991	-13.870
23	0.00	0.00	0.983	-12.970	0.00	0.49	0.967	-13.860
24	0.00	3.71	0.983	-12.866	0.00	6.28	0.973	-14.250
25	0.00	0.00	0.965	-13.316	0.00	0.00	0.968	-13.860
26	0.00	0.00	0.992	-12.520	0.00	0.00	0.950	-14.330
27	0.00	0.00	1.017	-8.630	0.00	0.00	0.974	-13.340
28	0.00	0.00	0.971	-13.830	0.00	0.00	1.011	-9.140
29	0.00	3.53	0.960	-14.772	0.00	2.70	0.961	-14.960
30	0.00	0.00	0.996	-14.780	0.00	0.00	0.946	-15.810
Taps	Line No.		Tap setting		Line No.		Tap setting	
	11		1.05		11		0.903	
	12		0.965		12		1.094	
	15		1.047		15		1.032	
	36		1.00		36		1.031	
System Generation(MW)			293.190		297.000			
System Losses (MW)			9.790		13.600			
Gen Cost ($ / Hr)			802.560		840.315			
No. of gradient adjustments			20		12			

3 RESULTS FOR THE KENYA POWER SYSTEM

3.1 Introduction

In sections (4.1) and (4.2) of this chapter, test results obtained for both the IEEE 6 bus and 0 bus sample systems are presented and analysed. They are compared with results obtained using he linear programming method as developed by Mang'oli [20]. The objective in both of these two udy systems is to minimise the cost of generating real power subject to satisfying both the system perating and equipment loading parameters as detailed in the solution approach developed in hapter 3.

As shall be described shortly, the Kenya power system is predominantly hydro, whose perating cost is negligible for all practical purposes because the water head, as a source of electrical nergy is " free " once a hydro station is constructed. In contrast, thermal stations are expensive to perate since they require a continuous fuel input (eg diesel, coal, gas, etc) for their normal peration. Of course the more the real power they are generating, the more the running cost (as letermined from the associated cost coefficients). On the Kenya power system, there is only one najor thermal station at Kipevu. In addition, the Kenya power system also has one geothermal station at Ol-Karia. Like hydro stations, geothermal stations have negligible running costs once the nitial infrastucture of sinking the steam wells and laying the piping to convey steam to the steam urbines is put in place.

Due to this unique nature of the Kenya power system, the best objective function to be minimised subject to meeting the system loading requirements is the real power loss. To achieve this, the generators are each set to dispatch a given amount of real power (apart of course from the swing bus) in accordance with an appropriate generation policy. The steepest descent algorithm d then be used to automatically adjust the available var sources to give the least real power loss g the transmission lines and a good voltage profile which assures reliable service.

The generation policy adopted in this thesis is to set both hydro and geothermal stations to erate power at their points of maximum efficiency and let the lone thermal unit to supply the cit, that is, the thermal unit is used for peaking up. Alternatively, the negligibly expensive hydro geothermal stations are set to generate at maximum capacity (depending on availability of urces) and the small deficit is supplied by the thermal station. In either case, the idea is to set mal generation to the lowest value possible so as to keep the cost of running the thermal station e minimum.

In the following sections, both the load flow and optimal power flow results for the Kenya ver system are presented and analysed. It should be mentioned that these results are considered ible since the steepest descent algorithm performed agreeably when applied to solve the IEEE us and 30 bus test systems. No comparison results are presented for the Kenya power system e to the author's knowledge, no similar study has been undertaken on the system before.

.2 Description of the Kenya Power system.

The Kenya power system is a medum sized network with 45 high voltage transmission lines erconnecting 42 main buses. The current peak demand is 630 MW. This load is served by nine in generating stations seven of which are hydro-electric stations. In addition, the system also has eothermal station at OL-Karia and a thermal station at Kipevu in Mombasa. However, these are all capacity generators and the bulk of the system demand is served by the hydro stations. These clude Gitaru, Kindaruma, Masinga, Kamburu, Turkwel, Kiambere and Owen Falls. Owen Falls s in Uganda but feeds power into the Kenyan grid via a tie line that joins the system at Musaga in western Kenya. Table 4.3(a) gives the line and transformer data. Generator data is presented in Table 4.3(b) and Table 4.3(c) shows the load data. This system data is obtained from reference 21.

Table 4.3 (a) Line and Transformer data for the Kenya Power System.

Line No.	Bus No.s	Line Impedance		Tap Setting	Line Char	Rating
		R (p.u)	X (p.u)	(p.u)	B (p.u)	(MVA)
1	5 - 15	0.0000	0.1000	0.0000	0.0000	33
2	2 - 9	0.0000	0.2222	1.0500	0.0000	75
3	3 - 10	0.0000	0.2400	0.0000	0.0000	50
4	4 - 11	0.0000	0.2776	0.0000	0.0000	47
5	1 - 12	0.0000	0.0765	0.0000	0.0000	170
6	6 - 13	0.0000	0.1171	0.0000	0.0000	110
7	7 - 19	0.0459	0.1878	0.0000	0.2856	210
8	8 - 35	0.0000	0.3064	0.0000	0.0000	65
9	9 - 25	0.0097	0.0559	0.0000	0.0108	150
10	10 -13	0.0187	0.0480	0.0000	0.0090	73
11	10 - 26	0.1209	0.2880	0.0000	0.0559	73
12	11 - 13	0.0770	0.0445	0.0000	0.0086	150
13	11 - 37	0.1042	0.2142	0.0000	0.0416	81
14	12 - 13	0.0039	0.0930	0.0000	0.0018	73
15	13 - 39	0.0000	0.0189	1.0000	0.0000	110
16	14 - 34	0.0100	0.0205	0.0000	0.0040	81
17	14 - 35	0.0000	0.1333	0.0000	0.0000	30
18	15 - 16	0.0660	0.1355	0.0000	0.0263	73

19	16 - 17	0.0415	0.0853	0.0000	0.0160	81
20	17 - 18	0.0212	0.0436	0.0000	0.0085	81
21	17 - 19	0.0094	0.0345	0.0000	0.0087	81
22	19 - 20	0.0378	0.0777	0.0000	0.0151	81
23	19 - 21	0.0668	0.1372	0.0000	0.0266	81
24	19 - 24	0.0674	0.1525	0.0000	0.0296	73
25	21 - 22	0.0361	0.0743	0.0000	0.0144	81
26	21 - 23	0.0571	0.1174	0.0000	0.0228	81
27	24 - 25	0.0395	0.0810	0.0000	0.0158	81
28	25 - 26	0.0432	0.0922	0.0000	0.0180	73
29	26 - 27	0.0000	0.0542	0.0000	0.0000	250
30	26 - 28	0.1270	0.3025	0.0000	0.0587	73
31	26 - 40	0.0000	0.0243	0.0000	0.0000	150
32	28 - 29	0.0437	0.1041	0.0000	0.0202	81
33	29 - 30	0.0874	0.2081	0.0000	0.0404	120
34	30 - 31	0.0914	0.2178	0.0000	0.0423	73
35	31 - 32	0.0284	0.0678	0.0000	0.0131	73
36	32 - 33	0.0874	0.2081	0.0000	0.0404	81
37	33 - 34	0.0217	0.0508	0.0000	0.0099	150
38	34 -36	0.0535	0.1099	0.0000	0.0213	230
39	34 - 41	0.0000	0.0539	0.0000	0.0000	61
40	37 - 38	0.0606	0.1246	0.0000	0.0242	83
41	39 - 40	0.0081	0.0468	0.0000	0.0705	83
42	39 - 41	0.0805	0.3572	0.0000	0.5480	150
43	39 - 42	0.0074	0.0302	0.0000	0.0459	200
44	40 - 42	0.0228	0.1361	0.0000	0.1935	150

5	41 - 42	0.0927	0.3791	0.0000	0.5764	250

Table 4.3 (b) Generator data for the Kenya Power System

Bus No.	PGmin (MW)	PGmax (MW)	QGmin (MVar)	QGmax (MVar)
	20.000	200.000	-90.000	100.000
2	20.000	150.000	-20.000	25.000
3	20.000	100.000	-20.000	20.000
4	20.000	100.000	-20.000	20.000
5	20.000	150.000	-25.000	25.000
6	20.000	100.000	-50.000	60.000
7	20.000	200.000	-50.000	60.000
8	10.000	60.000	-15.000	15.000
42	20.000	200.000	-90.000	100.000

Table 4.3 (c) Load data for Kenya Power System.

Bus	Load		Voltage limits	
No.	(MW)	(MVar)	Vmin (p.u)	Vmax (p.u)
1	0.00	0.00	0.95	1.15
2	0.00	0.00	0.95	1.15
3	0.00	0.00	0.95	1.15
4	0.00	0.00	0.95	1.15
5	0.00	0.00	0.95	1.15
6	0.00	0.00	0.95	1.15
7	0.00	0.00	0.95	1.15

8	0.00	0.00	0.95	1.15
9	0.00	0.00	0.95	1.15
10	0.00	0.00	0.95	1.15
11	0.00	0.00	0.95	1.15
12	0.00	0.00	0.95	1.15
13	0.00	0.00	0.95	1.15
14	0.00	0.00	0.95	1.15
15	0.00	0.00	0.95	1.15
16	11.00	7.00	0.95	1.15
17	7.00	4.00	0.95	1.15
18	7.00	2.00	0.95	1.15
19	3.00	2.00	0.95	1.15
20	9.00	2.00	0.95	1.15
21	0.00	0.00	0.95	1.15
22	7.00	3.00	0.95	1.15
23	11.00	5.00	0.95	1.15
24	14.00	7.00	0.95	1.15
25	6.00	3.00	0.95	1.15
26	0.00	0.00	0.95	1.15
27	197.00	84.00	0.95	1.15
28	1.00	0.00	0.95	1.15
29	0.00	0.00	0.95	1.15
30	0.00	0.00	0.95	1.15
31	1.00	0.00	0.95	1.15
32	0.00	0.00	0.95	1.15
33	0.00	0.00	0.95	1.05

	0.00	0.00	0.95	1.05
	69.00	40.00	0.95	1.05
	3.00	2.00	0.95	1.05
	6.00	3.00	0.95	1.05
	12.00	6.00	0.95	1.05
	0.00	0.00	0.95	1.05
	0.00	0.00	0.95	1.05
	0.00	0.00	0.95	1.05
	0.00	0.00	0.95	1.15

2 Initial Load flow results for the Kenya Power System

Table 4.3 (d) below shows the initial load flow results obtained with no compensation for oad buses apart from reactive generation at the eight generators. It can be observed that Nairobi, the highest demand of 197 MW, experiences an undervoltage of 0.9058 p.u. Load buses 14, nd 28 through to 36 experience high voltage levels ranging from 1.0377 p.u registered at Kipevu 1442 p.u at Rabai. These buses are geographically situated at the coastal region of the country are physically far from the main generating stations. In addition, there is inadequate reactive er generation in the region (only -11.39 Mvar obtained from Kipevu). It is worthwhile to note bus number 35 (Kipevu), though tending towards an overvoltage, registers a relatively lower age level than the other load buses in the region because there is a relatively high power sumption at this bus.

Load buses towards the western region of the country like Muhoroni, Chemosit, Kisumu exhibit lower voltage levels compared to load buses in the coastal region. A possible planation for this is that as opposed to the coastal region, the western region is served by two enerators (Owen Falls and Turkwel) dispatching a total of -51.48 Mvar of reactive power which nds to regulate the voltage levels at adjacent load buses. In addition, these buses have a more niform load distribution as compared to the coastern region whose load distribution is highly non-niform with Mombasa (Kipevu) consuming the bulk of the power (69 MW, 40 Mvar).

This state of non-uniformity of voltage levels in different parts of the system network is not esirable. A more uniform voltage profile is normally required to improve transmission efficiency nd reduce real power loss along the transmission lines. Efforts must therefore be made to arrest the ndency of under and overvoltages noted at load buses in the central, western and coastal regions nd maintain a more uniform voltage distribution throughout the system. This, ofcourse, means locating extra reactive power sources to compensate either for under or over voltages at specific oad buses, and is achieved by obtaining an optimal power flow solution, where reactive power ources available in the system are optimally set to dispatch a certain amount of reactive power.

Table 4.3(d) Initial Load flow results for the Kenya Power System

Bus No.	Bus Name	Generation MW	Generation MVar	Voltage | V | (p.u)	Phase angle δ (°)
1	GITARU	102.6800	2.3500	1.0000	0.0000
2	OLKARIA	45.0000	6.3900	1.0000	-5.6395
3	KNDRUMA	35.0000	-1.4400	1.0000	-5.1325
4	MASINGA	30.0000	-4.0400	1.0000	-4.8417
5	OWENFLS	42.0000	-11.1800	1.0000	-2.1584
6	KAMBURU	30.0000	-5.4300	1.0000	-7.9460

7	TURKWEL	40.0000	-40.3000	1.0000	-5.7816
8	KIPEVU	25.0000	-11.3900	1.0000	-18.3137
9	OLKARIA	0.0000	0.0000	0.9909	-11.4304
10	KDARUMA	0.0000	0.0000	1.0070	-9.9170
11	MASINGA	0.0000	0.0000	1.0146	-9.5491
12	GITARU	0.0000	0.0000	1.0013	-4.4988
13	KAMBURU	0.0000	0.0000	1.0070	-9.9450
14	KIPEVU	0.0000	0.0000	1.1081	-19.6232
15	JINJA	0.0000	0.0000	1.0120	-4.5366
16	TORORO	0.0000	0.0000	1.0005	-8.1460
17	MUSAGA	0.0000	0.0000	1.0020	-9.9688
18	WEBUYE	0.0000	0.0000	1.0000	-10.1294
19	LESSOS	0.0000	0.0000	1.0069	-10.3671
20	ELDORET	0.0000	0.0000	1.0024	-10.7323
21	MUHORON	0.0000	0.0000	0.9969	-11.8523
22	CHEMOSIT	0.0000	0.0000	0.9932	-12.1203
23	KISUMU	0.0000	0.0000	0.9872	-12.5117
24	LANET	0.0000	0.0000	0.9880	-12.3986
25	NAIVASHA	0.0000	0.0000	0.9852	-12.8906
26	JUJA RD	0.0000	0.0000	0.9633	-15.5450
27	NAIROBI	0.0000	0.0000	0.9058	-22.5723
28	SULHMUD	0.0000	0.0000	1.0569	-18.2846
29	KIBOKO	0.0000	0.0000	1.0812	-18.8790
30	MTITONDEI	0.0000	0.0000	1.1165	-19.7099
31	VOI	0.0000	0.0000	1.1343	-20.0133
32	MAUNGU	0.0000	0.0000	1.1358	-19.9858

33	MARIAKAN	0.0000	0.0000	1.1279	-19.6315
34	RABAI	0.0000	0.0000	1.1230	-19.4804
35	KIPEVU	0.0000	0.0000	1.0377	-22.5462
36	KILIFI	0.0000	0.0000	1.1223	-19.6468
37	KIGANJO	0.0000	0.0000	0.9941	-11.7298
38	NANYUKI	0.0000	0.0000	0.9820	-12.4769
39	KAMBURU	0.0000	0.0000	1.0116	-11.4821
40	DANDORA	0.0000	0.0000	0.9821	-13.7785
41	RABAI	0.0000	0.0000	1.1442	-18.2993
42	KIAMBERE	30.0000	-147.6500	1.0000	-11.7165
Total System generation (MW)			379.6800		
Total System losses (MW)			17.6800		
Total Generation Cost ($ / Hr)			-		
No. of Load flow iterations			3		

4.3.3. Optimal Power Flow results for the Kenya Power system.

From an initial operating point shown in Table 4.3(d), the optimal power flow program is executed taking into account compensation devices at load buses 19, 26, 27, 30 and 41. The optimal power flow results are shown in Table 4.3(e).

Examining the results, the following can be observed:

(i) All the load bus voltages are now within the permitted operating range of 0.95 and 1.05 p.u.

(ii) The voltage profile is now more uniform throughout the system than was the case at the tial operating point. The lowest voltage of 0.9549 p.u is registered at bus number 35 (Kipevu) d the highest of 1.0534 p.u. at bus 15 (Jinja). Bus 27 (Nairobi), which is the main consumption de , has a voltage of 0.9663 p.u.

(iii) The slack bus real power generation drops from a value of 102.68 MW at the initial erating point to a value of 96.59 MW at the optimum, representing a drop of 5.94 %. Likewise, e system real power losses drops by a similar amount, from an initial value of 17.68 MW to an timal value of 11.59 MW. The compensating devices at selected load buses 19, 26, 27, 30, and are set to optimally dispatch -24.88, 23.53, 31.28, -23.18 and -95.50 Mvar of reactive power spectively. All these figures are well within their operating limits.

(iv) On average, load buses with high power consumption display relatively lower voltage vels eg. Nairobi and Kipevu. This is however expected, and so long as the voltages do not violate e allowable limits, the operating point is practically acceptable.

It can be seen therefore that the Optimal Power Flow program has been efffectively used to tomatically obtain an optimal operating point that not only results in the least real power loss but so ensures that all the equipment and operating constraints are satisfied.

Table 4.3(e) Optimal power flow results for the Kenya Power System

Bus No.	Bus Name	Generation MW	Generation MVar	Voltage |V| (p.u)	Phase angle δ (º)
1	GITARU	96.5900	-5.3100	1.0053	0.0000
2	OLKARIA	45.0000	1.6900	1.0184	-5.0128
3	KNDRUMA	35.0000	-7.3900	1.0030	-4.4156
4	MASINGA	30.0000	-3.6900	1.0180	-4.2747
5	OWENFLS	42.0000	-4.4000	1.0484	-2.0888

6	KAMBURU	30.0000	-19.4600	1.0000	-7.1996
7	TURKWEL	40.0000	-32.7500	1.0412	-5.2081
8	KIPEVU	25.0000	18.5400	1.0082	-16.2091
9	OLKARIA	0.0000	0.0000	1.0194	-10.5391
10	KDARUMA	0.0000	0.0000	1.0241	-9.1060
11	MASINGA	0.0000	0.0000	1.0314	-8.8234
12	GITARU	0.0000	0.0000	1.0120	-4.1648
13	KAMBURU	0.0000	0.0000	1.0234	-9.1665
14	KIPEVU	0.0000	0.0000	0.9908	-17.2184
15	JINJA	0.0000	0.0000	1.0534	-4.2679
16	TORORO	0.0000	0.0000	1.0326	-7.3747
17	MUSAGA	0.0000	0.0000	1.0274	-8.9283
18	WEBUYE	0.0000	0.0000	1.0254	-9.0815
19	LESSOS	0.0000	-24.8800	1.0294	-9.2708
20	ELDORET	0.0000	0.0000	1.0249	-9.6215
21	MUHORON	0.0000	0.0000	1.0202	-10.7077
22	CHEMOSIT	0.0000	0.0000	1.0166	-10.9649
23	KISUMU	0.0000	0.0000	1.0109	-11.3404
24	LANET	0.0000	0.0000	1.0164	-11.3782
25	NAIVASHA	0.0000	0.0000	1.0163	-11.9384
26	JUJA RD	0.0000	23.5300	1.0021	-14.6193
27	NAIROBI	0.0000	31.2800	0.9663	-20.9502
28	SULHMUD	0.0000	0.0000	1.0174	-15.8921
29	KIBOKO	0.0000	0.0000	1.0148	-16.0688
30	MTITONDEI	0.0000	-23.1800	0.9970	-16.1202
31	VOI	0.0000	0.0000	1.0120	-16.8275

2	MAUNGU	0.0000	0.0000	1.0131	-16.9160
3	MARIAKAN	0.0000	0.0000	1.0054	-16.9198
4	RABAI	0.0000	0.0000	1.0010	-16.8573
5	KIPEVU	0.0000	0.0000	0.9549	-20.7722
6	KILIFI	0.0000	0.0000	0.9995	-17.0500
7	KIGANJO	0.0000	0.0000	1.0119	-10.9494
8	NANYUKI	0.0000	0.0000	1.0001	-12.4769
9	KAMBURU	0.0000	0.0000	1.0116	-11.6730
0	DANDORA	0.0000	0.0000	1.0141	-12.9417
1	RABAI	0.0000	-95.5000	1.0102	-15.4542
2	KIAMBERE	30.0000	-79.4600	1.0294	-10.9272
otal System generation (MW)			373.5900		
otal System losses (MW)			11.5900		
otal Generation Cost ($ / Hr)			-		
o. of gradient adjustments			20		

CHAPTER 5

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

.1 SUMMARY

The short term operational planning problem was solved. The solution gives the optimum ettings of generators, transformer taps and reactive power dispatch of the available var sources like ynchronous condensors, capacitors and reactors for every given load demand. In addition, perational system constraints like voltage levels are automatically corrected to ensure a reliable and ecure supply of power.

The OPF problem was solved using the steepest descent method. In the solution approach dopted, real and reactive power dispatches were simultaneously optimised by minimising a fuel cost function, subject to system operational constraints. In case of the Kenya power system, a suitable eal power generation policy is adopted and the OPF solution sought for the least real power loss long the transmission lines. This is accomplished by letting the hydro stations to serve the base oad, with the lone thermal unit providing peaking power.

Two sample systems were used for testing the developed OPF computer program. These two ystems, the IEEE 6 bus and 30 bus systems, are approved by the IEEE Power System Working Committee for power system studies. The optimal power flow results obtained for these two systems re compared with those obtained using a linear programming method developed earlier [20].

5.2 CONCLUSIONS

From the OPF results obtained for the three systems used in this stu conclusions can drawn.

(i) The steepest descent method converges to a lower value of operating c programming method. For the 6 bus system, the solution converges to an optimal $393.76 per hour for the steepest descent method as compared to $395.776 per programming method. For the 30 bus system, the figures are respectively $802.56 hour for the two methods, a 4.7 % difference in favour of the steepest descent m

It can therefore be concluded that in situations where resources are scarc strict utilization of available resources eg severe shortage of coal, oil etc), th method is more suitable than the linear programming method as a method of solv operational planning problem.

(ii) At the optimum, the steepest descent method locates an operating relatively more uniform load bus voltage distribution than the linear programmin uniform voltage profile improves transmission efficiency since line losses, which level differences in adjacent buses in the network, are less.

(iii) On average, the linear programming method requires fewer iteratio solution than the steepest descent method. For the 30 bus system, it requires 12 the steepest descent method requires 20 iterations. The linear programming meth suitable than the steepest descent method in cases where faster computations situations could include real time applications and contingency analysis. The to achieve faster computation times is of course a poorer voltage profile and an c her running costs.

RECOMMENDATIONS FOR FUTURE RESEARCH

For future research the following should be studied:

(i) Incorporating optimal ordering schemes in the load flow solution and determine the effect concerns improvement of computation speed and computer memory requirements, especially for ge power systems in excess of 300 buses.

(ii) Incorporating contingency analysis schemes in the short term planning problem to tomatically reflect for example generator outages.

(iii) Study methods of extending the algorithm to solve the long term hydro-thermal co-dination planning problem.

REFERENCES

[1] J. F. Dopazo, O. A. Klitin, G. W. Stagg and M. Watson, " An Optimization Technique for real and reactive power allocation ", *Proceedings of the IEEE*, 1967, pp 1877-1885

[2] J. Peschon, D. S. Piercy, W. F. Tinney and O. J. Tveit," Optimum Control of Reactive Power Flow ", *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-87, Jan 1968, pp 40 - 88.

[3] Dommel H.W. & W.F. Tinney, "Optimal power flow solutions", *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS 87, October 1968, pp 1866-1876.

[4] R. B. Gungor, N. F. Tsang and B. Webb, " A Technique for Optimizing Real and Reactive Power Schedules " , *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-90, July 1971, pp 1781- 1790

[5] Stagg G.W. & El-Abiad A.H., "Computer methods in Power system analysis", McGraw Hill, New York, 1968.

[6] William D. Stevenson, "Elements of Power System Analysis", McGraw Hill B.C., New York, Chapter 8, 4th Edition, 1988.

[7] Nagrath I. J. & D.P. Kothari, "Modern Power System Analysis," Tata McGraw Hill, 2nd edition, Chapter 6, 1989.

[8] Tinney W.F. & C.E. Hart, "Power flow Solution by Newton's method", *IEEE Transactions on Power Apparatus and Systems,* Vol. 86, PP 1449, Nov. 1967.

[9] Stott B., "Decoupled Newton load flow", *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS - 91, pp 1955, 1972.

[10] Stott B. & O. Olsacc, "Fast decouped load flow", *IEEE Transactions on Power Apparatus and Systems,* Vol PAS - 93 pp 859, 1974.

[11] Wood Allen J & B.F. Wollenburg, "Power generation, operation and control," John Wiley

and sons, 1984.

] Sato and W. F. Tinney, " Techniques of exploiting the sparsity of the Network admittance matrix ", *Transactions on Power Apparatus and Systems*, Vol. PAS-82,, pp 944, 1963.

] Alsacc O. & B. Scott, "Optimal load flow with steady state security", *IEEE Transactions on Power Apparatus and Systems*, Vol. 93, May/June 1974, pp 745-751.

] Hajdu P.L. et al, "Optimum load shedding Policy for power systems", *IEEE Transactions on Power Appartus and Systems*, Vol. 87, March 1968, pp 784-795.

] K.R.C. Mamandur & R.D. Chenoweth, "Optimal Control of reactive power flow for improvements in voltage profiles and for real power loss minimisation", *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-100, July 1981 pp 3185- 3194.

5] Donald A. Pierre, "Optimization Theory with Applications, Wiley, 1969.

7] Rao, S.S., "Optimization theory and Applications", Wiley Eastern Ltd., 2nd Edition, 1984.

3] Burchett R.C. et al, "Developments in optimal power flow", *IEEE Transactions on Power Apparatus and Systems*, Vol PAS-101, Feb 1982, pp 406 - 413.

9] R.R. Shoults & D.T. Sun, "Optimal power flow based on P-Q decomposition", *IEEE Transations on Power Apparatus and Systems*, Vol PAS - 101, Feb 1982, pp 397-405.

0] Mang'oli M. K. W., " Optimal Reactive Power Planning Using Decomposition Techniques", PhD Thesis, Jan. 1992.

1] Kenya Power and Lighting , " High Voltage System studies and operations Guide ", 1991

APPENDIX A

COMPUTATION OF THE GRADIENT VECTOR:

The sequence of computations is outlined in the simplifed flow chart of fig. 3.1. The part that essentially determines computer time and storage requirements is labeled "solve load flow". Here, the main task performed includes the computation and triangularization of the 2N x 2N Jacobian matrix, which can be quite large for a large power system. The Gaussian elimination and backward substitution technique employed in this thesis makes it easier to solve the system of linear equations either for the voltage magnitude and angle corrections in the power flow loop or for the Lagrangian multipliers, [λ]. This technique is faster since the inverse of the Jacobian matrix need not be explicitly determined. The following sections outline the main computations carried out in the optimization algorithm as effected in the developed computer program.

(A.1) Load flow solution:

The set of linear equations solved in the load flow loop is:

$$\begin{bmatrix} H & : & N \\ \cdots & \cdot & \cdots \\ J & : & L \end{bmatrix} \begin{bmatrix} \Delta\delta \\ \cdots \\ \dfrac{\Delta V}{V} \end{bmatrix} = \begin{bmatrix} \Delta P \\ \cdots \\ \Delta Q \end{bmatrix} \qquad \text{A1}$$

Where [$\Delta\delta$], [$\Delta V/V$] are vectors of voltage angle and relative magnitude corrections respectively and [ΔP] and [ΔQ] are the vectors of power residuals with components:

$$\Delta P_k = P_{k\,sched} - P_{kcalc} \qquad \text{A2}$$

$$\Delta Q_k = Q_{k\,sched} - Q_{kcalc} \qquad \text{A3}$$

nd [H], [N], [J] and [L] are submatrices of the Jacobian matrix with the elements

$$H_{km} = \partial P_k(V,\delta) / \partial\delta_m , \quad N_{km} = (\partial P_k(V,\delta) / \partial V_m).\, V_m \qquad \text{A4}$$

$$Q_{km} = \partial Q_k(V,\delta) / \partial\delta_m . \qquad L_{km} = (\partial Q_k(V,\delta) / \partial\delta_m).\, V_m$$

The expressions for calculating the elements of the submatrices [H], [N], [J] and [L] were iven earlier in equations (2.14 - 2.17).

.2 Lagrange multipliers:

Once the power flow is accurate enough (by satisfying the specified tolerance in power nismatch vector), the algorithm is switched to the solution of:

$$\begin{bmatrix} H & : & N \\ \dots & . & \dots \\ J & : & L \end{bmatrix}^T \begin{bmatrix} \lambda_P \\ \dots \\ \lambda_Q \end{bmatrix} = -Z \begin{bmatrix} H_1 \\ \\ N_1 \end{bmatrix} - \begin{bmatrix} \frac{\partial(\sum W_j(\delta))}{\partial \delta} \\ \\ \frac{\partial \sum W_j(V)}{\partial V} \end{bmatrix}$$

(A5)

Where $W_j(.)$: refers to the penalty terms and only enters if the objective function has been augmented with penalty functions which depend on the variables indicated by the partial derivative.

$[\lambda_P]$, $[\lambda_Q]$: Are subvectors of the Lagrangian multipliers associated with real and reactive power equality constraints respectively and :

$[H_1] = [\partial P_1(V,\delta) / \partial \delta]$ (A6)

$[N_1] = [\partial P_1(V,\delta) / \partial V). V]$ (A7)

$Z = \partial f_1 / \partial P_{G1}$ (A8)

With f_1 being the cost of generating active power at the swing bus.

A.3 The gradient vector

After A5 has been solved for Lagrangian multipliers, the gradient of the objective function with respect to all control parameters is computed. It's components are as follows:

) For voltage control:

$$\partial V_i = \{(1 / V_i)(Z.N_{1i} + \Sigma \lambda_{Pm} N_{mi} + \Sigma \lambda_{Qm} L_{mi})\} + \{\partial \Sigma W_j(V) / \partial V_i\} \qquad \text{(A9)}$$

2) For power source control:

$$/\partial P_{Gi} = \{\partial f_i / \partial P_{Gi}\} - \lambda_{Pi} + \{\partial \Sigma W_j(P_{Gi}) / \partial P_{Gi}\} \qquad \text{(A10)}$$

3) For tranformer tap control:

$$/\partial t_{ik} = \{(a_i.N_{ik} + b_i H_{ik} + a_k N_{ki} + b_k H_{ki}) / t_{ik}\} + 2V_k^2(b_k B_{ik} - a_k G_{ik}) + \{\partial \Sigma W_j(t_{ik}) / \partial t_{ik}\} \qquad \text{(A11)}$$

Where:

$$= \begin{cases} Z, & \text{if } i = 1, \text{ slack node} \\ \lambda_{Pi}, & \text{otherwise} \end{cases} \qquad \text{(A12)}$$

$$_i \quad = \begin{cases} \lambda_{Pi}, & \text{if } i = P, Q \text{ node} \\ 0, & \text{otherwise} \end{cases} \qquad \text{(A13)}$$

analogous for a_k, b_k

d $G_{ik} + jB_{ik} = -t_{ik} Y_{ik}$

t_{ik} = turns ratio for transformer connecting buses i and k.

Y_{ik} = element of bus admittance matrix.

Feasible direction of steepest descent:

With the gradient componets from (A9), (A10) and (A11) available, the feasible direction of pest descent [r], is formed with:

$$= \begin{cases} 0, & \text{if } \nabla F_i < 0 \text{ and } U_i = U_{i\,max} \\ 0, & \text{if } \nabla F_i > 0 \text{ and } U_i = U_{i\,min} \\ -\nabla F_i, & \text{otherwise} \end{cases} \qquad \text{(A14)}$$

en the adjustments of control variables in this direction follow from

$$\Delta U_i = \alpha . r_i \qquad \text{(A15)}$$

OPTIMUM VALUE OF α

The objective function, F, becomes a function of the scalar α only when the control variables moved in the direction of [r]. Fig.A1 below shows the variation of F with α.

igure A1. Variation of F with step length along steepest descent direction

The minimizing step length, α_{min}, can be found from three points at which the value of F

wn. One value, F_o is already known from the previous optimization cycle. A second and

value is found by exploratory moves along the direction of descent [r].

1: Compute $\Delta = \sum_{i=1}^{m} r_i^2$

hen divide each component of [r] by Δ.

normalization of the descent direction ensures that r_i is a reasonable change in U_i.

2: Evaluate $F(\alpha)$ at the points 0, 1, 2, 3, ---α, α_1, α_2 where α_2 is the first of these values

hich F has increased.

hen the optimum value of α must lie between α_1 and α_2 i.e,

$$\alpha < \alpha_{min} < \alpha_2 \qquad \text{(A16)}$$

3: A quadratic polynomial is then fitted to the three values $F(\alpha)$, $F(\alpha_1)$ and $F(\alpha_2)$ and it's

mum $F(\alpha_{min})$, is located at

$$= \frac{F(\alpha_1)(\alpha_2^2 - \alpha^2) + F(\alpha)(\alpha_1^2 - \alpha_2^2) + F(\alpha_2)(\alpha^2 - \alpha_1^2)}{F(\alpha_1)(\alpha_2 - \alpha) + F(\alpha)(\alpha_1 - \alpha_2) + F(\alpha_2)(\alpha - \alpha_1)} \qquad \text{(A17)}$$

f $F(\alpha_{min}) < F(\alpha)$. Then α_{min} is accepted as the estimate of the optimum value of the

otherwise α is taken as the true minimum, α_{opt}.

APPENDIX B

PROGRAMMING APPROACH

ORGANIZATION OF THE PROGRAM

The program is modularly designed where one main program calls a number of subroutines written to perform a specific task. Each subroutine is made as independent and self contained ossible, making it easy to test and modify each subroutine separately without effecting major ges in the calling program.

CHOICE OF COMPUTER LANGUAGE FOR PROGRAMMING.

A number of high level programming languages were initially studied to find the most opriate in terms of ease and efficiency of coding the OPF program. These languages included ascal and FORTRAN. FORTRAN was chosen for two main reasons:-

(i) Power system analysis involves a series of complicated computations involving complex ber arithmetic. Almost all the parameters used in power system analysis like line impedances, admittance matrices, nodal powers, line power flows, etc are complex quantities. In FORTRAN, aration of complex numbers is an in built feature unlike in C and Pascal where user defined ctions have to be written to implement complex number arithmetic. Availability of complex ables in FORTRAN makes it easy and efficient to solve problems in power system studies. This ure greatly influenced the choice of FORTRAN as the language of implementing the OPF gram.

(ii) Power system analysis also involves huge amounts of data which should be arranged in resentable way to facilitate easy result analysis. FORTRAN has a powerful inbuilt feature for atting data called FORMAT, which makes easy arrangement of huge amounts of data possible

DETAILED DESCRIPTION OF THE OPF PROGRAM.

Reference is made to figure 3.1 of chapter 3 which gives a flow chart of the main steps ved in solving an optimal power flow problem using the steepest descent method as developed s work.

An important point to note from figure 3.1 is that the steepest descent algorithm is basically es of transitions from one load flow solution (which represents a feasible but not yet optimal ating point) to another operating point with a lower value of objective function (cost of ration, real power loss etc.) via the gradient steps. Once a feasible load flow solution is ned, the parameters required to compute the vector of gradients (partial derivatives of objective tion with respect to control variables) are available.

The gradient vector is then used to adjust the control variables, in this case real power ration, voltage magnitude at PV buses and transformer tap settings, to a new operating point a lower value of objective function. A step length that ensures the maximum adjustment of the rol variables to an operating point with the least cost of generation (or real power loss) in the tion of steepest descent (negative) direction is required.

At the new operating point, a load flow is run and a vector of gradients again computed. If e gradients are less than a user specified tolerance (a default figure of $0.04 per unit of control able is assumed), the minimum value of the objective function has been obtained. This esents the operating point with the least cost of generation. Apart from achieving the minimum eration cost, all the system operating constraints like load bus voltages, line flows etc and pment loading constraints like generator outputs, transformer taps, reactive power from densors etc are all within their operating ranges.

In view of the foregoing, the following two main programs and eleven sub-programs were eloped and coded in FORTRAN 77. The modules are given names that give an insight into their nded functions.

.1 MAIN PROGRAM " OPFLOW "

This is the main calling program which controls the sequence of calling the various routines to perform specific tasks. It also declares all those variables and arrays that are global he subroutines serving under it and assigns suitable variable names that are consistent throughout subroutines.

.2. MAIN PROGRAM " DATINPUT "

As the name implies, this program facilitates the entry of data that completely defines a given ver system network. This information is stored in a user defined data file and contains the owing information:-

- Number of buses in the network
- Number of transmission lines
- Number of generators
- Number of synchronous condensors or capacitors
- Number of tap changing transformers and their position in the network
- The selected swing bus
- Number of inductors in the network
- Branch impedances and line charging admittances

- Scheduled real and power generation at buses with generators

- Scheduled real and reactive power demand at each bus in the network.

- Initial assumed voltages at each system bus .

3.-Allowable limits for both equipment loading (generators, condensors, transformer taps, [c]itor, inductors)and operational constraints (bus voltage limits and real and reactive line flows)

-Bus names, up to 20 characters long. This is optional but recommended, since it gives more [mean]ing than a mere bus number.

An appropriate data file for the IEEE 6bus system could be IEEE6B.DAT. Once all this [infor]mation is read and appropriately stored in the user defined data file, control is passed back to [the m]ain calling program OPFLOW, which calls the next subroutine. It should be emphasized that [subro]utine DATINPUT is excluded if a system data file is already in existence.

. SUBROUTINE " SYSREAD " :

This subroutine assumes a data file containing all the information describing a given power [syste]m is available and residing in the active directory of main calling program OPFLOW. Like the [nam]e implies, SYSREAD reads in all the system data from the data file and assigns values to the [varia]ble names already declared in the main program OPFLOW. SYSREAD makes it easy to study [vario]us systems whose data files have already been created using program DATINPUT.

4. SUBROUTINE "YBUILD"

Using the line impedances and line charging admittances as obtained from the system data [file], this subroutine computes the elements of the N by N bus admittance matrix. In building this [matr]ix, YBUILD only considers those buses that are connected. Logic is inserted to exclude those [buse]s in the system that are not linked. This matrix features in virtually all the subsequent utation processes.

Once the bus admittance matrix is built, control is passed back to OPFLOW which invokes utine LDFLOW.

5 SUBROUTINE " LDFLOW "

This is the central core around which the whole steepest descent formulation centers. outine LDFLOW uses bus voltages and the bus admittance matrix already built by YBUILD lve the load flow equations using the Newton Raphson method as detailed in section 2.3 of ter 2.

In this subroutine, the load flow equations (2.9) are solved and used to compute the 2N by wer mismatch vector in accordance with equation (2.12). If the power mismatches are all less a specified tolerance, then the solution has converged and control passed back to the main ng program, OPFLOW. Otherwise the solution has not converged and bus voltage magnitudes angles have to be adjusted to a new setting. This task is accomplished by two subroutines, the of which is subroutine JACOB.

6 SUBROUTINE "JACOB"

In this subroutine, the elements of the 2N by 2N Jacobian matrix are computed in accordance equation (2.10) and as detailed in section 2.1.2 of chapter 2. Once all the elements of the bian matrix have been computed, JACOB passes control back to subroutine LDFLOW which kes subroutine GAUSSE.

7 SUBROUTINE "GAUSSE"

LDFLOW passes to this subroutine the 2N by 1 power mismatch vector and the 2N by 2N obian matrix obtained from subroutine JACOB. GAUSSE uses these two matrices to compute voltage magnitude and angle corrections in accordance with equation (2.13) of chapter 2. In this e, GAUSSE uses the Gaussian forward elimination and backward substitution method to evaluate voltage magnitude and angle corrections. This is very efficient since the inverse of the Jacobian rix need not be explicitly determined. Once this is accomplished, control is passed back to routine LDFLOW which updates the bus voltage magnitudes and phase angles according to ation (2.25). Using this new set of bus voltages, the power mismatches are again computed and process repeated until all fall below a specified tolerance. At this stage a load flow solution has n obtained and control passed back to the main calling program OPFLOW.

If an optimal power flow solution is not needed, OPFLOW calls the output subroutine that nts out the load flow results. Otherwise it calls subroutine STEEPDNT that implements the epest descent algorithm.

.8 SUBROUTINE " STEEPDNT "

This subroutine assumes a feasible load flow solution exists and does the following upon ng invoked:-

(i) It evaluates the vector of Lagrange multipliers in accordance with equation (3.25) of apter 3 using the transpose of the latest Jacobian matrix computed by JACOB.

(ii)Using the vector of Lagrange Multipliers so determined, STEEPDNT computes the mponents of the gradient vector, automatically incorporating penalty functions to account for olated functional constraints like line flows, load bus voltage limits etc. This is in accordance with ation (3.37). If all the elements of the gradient vector are less than a specified tolerance [a default e of $0.04 per unit of control variable is used]a minimum is detected and no further reduction he objective function is possible. The output subroutine that prints out OPF results is therefore ed .

Otherwise, a new set of control variables is required to further lower the value of the ective function. To achieve this, STEEPDNT calls two subroutines in the order given below.

.9. SUBROUTINE " OPSTEP "

Using the gradient vector obtained in STEEPDNT, subroutine OPSTEP determines the imum step length that moves the control variables to a point with the least value of objective ction in the direction of steepest descent (negative). This is accomplished by conducting a one nensional search as detailed in appendix A. OPSTEP conducts the search of the optimum step gth by evaluating the value of the objective function at several step lengths and then picking out optimum. It uses two subroutines; subroutine ADJUST and COST.

3.10.SUBROUTINE "ADJUST "

Given the vector of gradient components and a step length, subroutine ADJUST moves the ctor of control variables to a new operatng point in the negative direction. The value of objective nction at the new operating point needs to be evaluated. This is achieved using subroutine COST.

3.11.SUBROUTINE " COST "

This calculates the cost of generating real power given the number of generators, cost efficients of each generator and the power generated by each. This value of cost is passed back to outne OPSTEP that determines whether the current step length is the optimum or not. If yes, this length is used to determine the new operating point by determining a new set of control ables. OPSTEP passes control back to STEEPDNT that proceeds until the minimum cost is d. At this point, an optimal operating point has been obtained and a subroutine to print out the results is invoked.

12. SUBROUTINE " OUTPUT "

As the name implies, this is the subroutine that prints out the output results for either the nal load flow solution or an OPF solution. These results include the following :

(i)Both Real and Reactive power generation at generator buses .

(ii) Bus voltage magnitudes and their phase angles relative to the swing bus .

(iii) Real and Reactive power flows on transmission lines .

(iv) Real and Reactive power losses on transmission lines .

(v) Tap settings of regulating transformers .

(vi) Cost of generation .

(vii) Reactive power out put of condensors and capacitors at compensated load buses.

s output data is stored in a data file and is arranged in a way so as to facilitate easy reading and lysis of the results.

A complete listing of the whole OPF program as written and coded in FORTRAN 77 is given appendix C.

APPENDIX C
LISTING OF FORTRAN 77 SOURCE PROGRAMS

```
 PROGRAM opflow
is is the main OPF calling program
-ordinates all the main subroutines

arts by declaring all variables
 complex Ybus(100,100),Zbus(100,100),Ychar(100,100),Y(100)
 COMPLEX Sgen(100),Sload(100),S(100),E(100),Spq,g,Yvars(100)
 Real Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
 REAL Qgmax(100),Qgmin(100),P(100),Pgmax(100),Pgmin(100)
 REAL delPQ(100),delAV(100),JACO(100,100),Esh(100),Tapset(100)
 real a(100),b(100),c(100),BJ(100,100),Tmin(100),Tmax(100)
 real Spqmax(100,100),Sgmax(100),jak(100,100)
 INTEGER Ltype(100),Itrans(100),Method(100)
 character*10 fname
 character*12 Bname(100)
 common/blk1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
 common/blk2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax,
6Pgmin,Esh
 common/blk3/delPQ,delAV,Jaco,BJ
 common/blk4/Ltype,Itrans,method
 common/blk5/tapset,Jak,Tmin,Tmax
 common/blk6/a,b,c,Spqmax,Sgmax
 common/blk7/Bname
 print*,'Type in the name of file containing SYSTEM DATA : '
 read'(a10)',fname
 OPEN(unit=9,file=fname,status='OLD')
 rewind(9)
 call sysread(n,Iswing,Ngen,Lines,Lcap,Treal,Timag)
urns with system data
ild the Ybus
 call Ybuild(n)
 order of jacobian matrix
 M=2*n
erform a load flow to establish system state
 call LDflow(n,Iswing,k,m,Treal,Timag)
ecide if optimal load flow is needed
 print*,'There are two options:'
 print*
 print*,'1. Normal load flow solution'
 print*,'2. Optimal load flow solution'
 print*
 print*,' Choose 1 or 2 '
 read*,Kchoice
 if(Kchoice.eq.2)then
```

```
      ksw=1
c if OPF needed invoke the steepest descent algorithm
      call steepdnt(n,iswing,conver,treal,timag)
      end if

      call output(n,Iswing,k,Lines,Lcap)
c outputs the loadflow results into a file
      end

      subroutine sysdata(n,Iswing,Ngen,lines,Lcap,treal,timag)
c start of subroutines

c READS IN ALL THE SYSTEM DATA FROM A STORED DATA FILE
      COMPLEX Ybus(100,100),Zbus(100,100),Ychar(100,100),Y(100)
      COMPLEX Sgen(100),Sload(100),S(100),E(100),Yvars(100)
      REAL Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
      REAL Qgmax(100),Qgmin(100),P(100),Pgmax(100),Pgmin(100)
      REAL delPQ(100),delAV(100),JACO(100,100),Esh(100),tapset(100)
      real magv,angle,a(100),b(100),c(100),BJ(100,100),Tmin(100)
      real Spqmax(100,100),Sgmax(100),Jak(100,100),Tmax(100)
      character*12 Bname(100)
      INTEGER Ltype(100),Itrans(100),Method(100)
      common/blk1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
      common/blk2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax,
     6Pgmin,Esh
      common/blk3/delPQ,delAV,Jaco,BJ
      common/blk4/Ltype,Itrans,method
      common/blk5/tapset,Jak,Tmin,Tmax
      common/blk6/a,b,c,Spqmax,Sgmax
      common/blk7/Bname
      character null*18
        read(9,40)null
40      format(a18)
        read(9,40)null
        read(9,40)null
        read(9,40)null
        read(9,40)null
        read(9,56)n,Iswing,Ngen,Lines,Lcap
56      format(2x,i3,7x,i3,10x,i3,11x,i3,8x,i3)
        read(9,40)null
        read(9,40)null
        read(9,40)null
c now read in the line impedances
        do 10 k=1,lines
        read(9,46)I,J,YBUS(I,J),YCHAR(I,J),tapset(i),Spqmax(i,j)
  46 FORMAT(2X,I4,1x,I4,4X,F7.4,2x,F7.4,4X,F7.4,2x,F7.4,4X,F7.4,
```

```
62x,f8.13)
   Ybus(j,i)=Ybus(i,j)
   Ychar(J,I)=Ychar(I,J)
   tapset(j)=tapset(i)
   Spqmax(j,i)=Spqmax(i,j)
10 continue
  read(9,40)null
  read(9,40)null
  read(9,40)null
  read(9,40)null
  print*
  do 745 k=1,n
ad(9,702)i,Pgmin(i),Pgmax(i),Qgmin(i),Qgmax(i),Emin(i),Emax(i),
  1Tmin(i),Tmax(i)
2  format(1x,i3,4x,f6.3,2x,f6.3,4x,f6.3,2x,f6.3,4x,f6.4,2x,f6.4,
  23x,f7.4,2x,f7.4)
45 continue
   read(9,40)null
   read(9,40)null
   read(9,40)null
   read(9,40)null
   read(9,40)null
   read(9,40)null
   read(9,40)null
   print*
   do 327 j=1,n
   read(9,905)i,Bname(i),sgen(i),sload(i),E(i),Ltype(i)
05  format(i3,1x,A12,2x,f8.4,2x,f8.4,3x,f8.4,2x,f8.4,3x,f6.4,2x,
   2f6.4,1x,i1)
   S(i)=Sgen(i)-Sload(i)
   Qvar=aimag(Sload(i))
   Qmax(i)=Qgmax(i)-Qvar
   Qmin(i)=Qgmin(i)-Qvar
   print 905,i,Bname(i),sgen(i),sload(i),e(i),ltype(i)
27 continue
   read(9,40)null
   read(9,40)null
   read(9,40)null
   print*
   read(9,127)treal,timag
27  format(9x,f8.6,10x,f8.6)
   print 127,treal,timag
   read(9,40)null
   read(9,40)null
   read(9,40)null
   read(9,40)null
   print*
```

```
   do 77 j=1,n
   read(9,78)i,method(i),Itrans(i),Esh(i),Yvars(i)
   format(2x,i3,4x,i1,4x,i3,4x,f6.4,4x,f9.4,2x,f9.4)
   continue
   read(9,40)null
   do 998 j=1,n
   read(9,999)i,a(i),b(i),c(i),Sgmax(i)
9  format(2x,i3,9x,f9.6,9x,f9.6,9x,f9.6,4x,f8.3)
8  continue
   close(9)
   end

  subroutine Ybuild(n)
his subroutine builds the Ybus matrix
  real a,b,c
  complex Ybus(100,100),Zbus(100,100),Ychar(100,100),Y(100)
  complex Yvars(100),Sgen(100),Sload(100),S(100),E(100)
  real Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
  real Qgmin(100),P(100),Pgmax(100),Pgmin(100),delPQ(100)
  real Jaco(100,100),Esh(100),tapset(100),BJ(100,100)
  real Tmin(100),Tmax(100),Qgmax(100),delAV(100),Jak(100,100)
  integer n,Ltype(100),Itrans(100),method(100)
  common/blk1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
  common/blk2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax
 1,Pgmin,Esh
  common/blk3/delPq,delAV,jaco,BJ
  common/blk4/Ltype,Itrans,method
  common/blk5/tapset,Jak,Tmin,Tmax
  open(unit=8,file='Ybus.dat',status='old')
  rewind(8)
valuate elements of bus admittance matrix from
ine charging admittances and branch impedances
   DO 30 I=1,N
   Ybus(i,i)=Yvars(i)
   DO 40 J=1,N
   IF(I.NE.J.and.Ybus(i,j).ne.(0.0,0.0)) THEN
   Ybus(i,i)=Ybus(i,i)+(1/Ybus(i,j))+Ychar(I,J)
   Ybus(i,j)=-1/Ybus(i,j)
   END IF
0  CONTINUE
30 CONTINUE
   return
   end

   SUBROUTINE Ldflow(n,Iswing,k,m,Treal,Timag)
This is the main load flow computational subroutine
```

```
      integer n,m,iswing
      COMPLEX Ybus(100,100),ZBUS(100,100),Ychar(100,100),Y(100)
      COMPLEX Sgen(100),Sload(100),S(100),E(100),Yvars(100)
      REAL Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
      REAL Qgmax(100),Qgmin(100),P(100),Pgmax(100),Pgmin(100)
      REAL delPQ(100),delAV(100),JACO(100,100),Esh(100)
      real a(100),b(100),c(100),Tmin(100),Tmax(100)
      real Spqmax(100,100),Sgmax(100),Jak(100,100),tapset(100)
      INTEGER Ltype(100),Itrans(100),Method(100)
      character*12 Bname(100)
      COMMON/BLK1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
      COMMON/BLK2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax
     2,Pgmin,Esh
      COMMON/BLK3/delPQ,delAV,JACO,BJ
      COMMON/BLK4/Ltype,Itrans,Method
      common/blk5/Tapset,Jak,Tmin,Tmax
      common/blk6/a,b,c,Spqmax,Sgmax
      common/blk7/Bname
      open(unit=10,file='change.dat',status='old')
start iterating
      k=0
55    Pchr=0.0
      Qchi=0.0
      Xch=0.0
      do 160 i=1,n
Ignore slack bus
Find the bus powers for all buses
      P(I)=0.0
      Q(I)=0.0
      DO 410 J=1,N
       Gpq=real(ybus(i,j))
       Bpq=-aimag(ybus(i,j))
       ep=real(E(i))
       fp=aimag(E(i))
       eq=real(E(j))
       fq=aimag(E(j))
       P(i)=P(i)+(ep*(eq*Gpq+fq*Bpq))+(fp*(fq*Gpq-eq*Bpq))
       Q(i)=Q(i)+(fp*(eq*Gpq+fq*Bpq))-(ep*(fq*Gpq-eq*Bpq))
10    continue
 a value for bus powers p(i) and q(i) for bus i...n
find change
Adjust the slack bus power
       if(i.eq.iswing)then
       Pl=real(Sload(iswing))
       Ql=aimag(Sload(iswing))
       Pt=Pl+P(iswing)
       Qt=Ql+Q(iswing)
       Sgen(iswing)=cmplx(Pt,Qt)
```

```
    else
    Pshed=real(S(I))
    Qshed=aimag(S(i))
    delPQ(I)=Pshed-P(I)
ind max changes
    if(abs(delPQ(I)).gt.Pchr) then
    Pchr=abs(delPQ(I))
    end if
    if(ltype(i).eq.2)then
    rel=real(S(i))
is a PV bus
    Ltype(i+n)=2
    s(i)=cmplx(REL,q(i))
    Sgen(i)=S(I)+Sload(i)
    delPQ(i+n)=0.0

    end if
heck for transformer control
    if(ltype(i).eq.3)then
    if(ltype(i+n).eq.0)then
    ltype(i+n)=3
    end if
    delPQ(i+n)=Qshed-Q(i)
    if(abs(delPQ(i+n)).gt.Qchi)then
    Qchi=abs(delPQ(i+n))
    end if
    end if
     end if
0   CONTINUE
est for convergence
    write(10,51)
    format(2x,'i    delP    delQ    P(i)    Q(i)    Vp    tap')
    do 49 i=1,n
    ep=real(E(i))
    fp=aimag(E(i))
    Vp=cabs(E(i))
    write(10,34)i,delPQ(i),delPQ(i+n),P(i),Q(i),Vp,tapset(i)
    continue
est for convergence using the mismatches
    if((Pchr.le.treal).and.(Qchi.le.timag))return
    call jacob(n,m,iswing)
acob returns with the elements of the jacobian
o far we have the elements of the jacobian and the power
match vector delPQ
olve the loadflow equations [delPQ]=[JACO][delAV] for delAV,

hanges in volt. mag. and angles
 *
```

```
c the order of the jacobian matrix is 2n, where n is the no. of
c buses
c call GAUSSE, a subroutine that solves for delAV, given delPQ
c and JACO
c
      ksw=0
      call GAUSSE(m,Iswing,ksw)
c returns with voltage magnitude and angle corrections
c adjust volt.and angles
      do 870 i=1,n
c ignore slack bus
       if(i.ne.iswing)then
c add changes of real and imaginary components of voltage
       ep=real(E(i))
       fp=aimag(E(i))
       Emag=sqrt((ep*ep)+(fp*fP))
       angle=atan2(fp,ep)
       if(abs(delAV(i)).gt.0.55)then
       if(delAV(i).gt.0.0)then
       delAV(i)=0.55
       else
       delAV(i)=-0.55
       end if
       end if
       if(abs(delAV(i+n)).gt.0.1)then
       if(delAV(i+n).gt.0.0)then
       delAV(i+n)=0.1
       else
       delAV(i+n)=-0.1
       end if
       end if
       IF(ltype(i).eq.2)then
       angle=angle+delAV(i)
       ep=Esh(i)*cos(angle)
       fp=Esh(i)*sin(angle)
       E(i)=cmplx(ep,fp)
       else
       angle=angle+delAV(i)
       Emag=Emag+(delAV(i+n)*Emag)
       ep=Emag*cos(angle)
       fp=Emag*sin(angle)
       E(i)=cmplx(ep,fp)
       end if
       END IF
 884  format(1x,i2,2x,f9.4,2x,f9.4,4x,f9.4,2x,f9.4,4x,f9.4,2x
     1,f9.4)
 870  continue
c adjust iteration count and continue
```

```
      k=k+1
      go to 165
iteration continues
      END

        SUBROUTINE jacob(n,m,iswing)
Does the calculation assuming polar co-ordinates are in use
        integer n,iswing,M
     COMPLEX Ybus(100,100),ZBUS(100,100),Ychar(100,100),Y(100)
     COMPLEX Sgen(100),Sload(100),S(100),E(100),Yvars(100)
     REAL Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
     REAL Qgmax(100),Qgmin(100),P(100),Pgmax(100),Pgmin(100)
     REAL delPQ(100),delAV(100),JACO(100,100),Esh(100)
    1,tapset(100)
     real Tmin(100),Tmax(100),jak(100,100),BJ(100,100)
      INTEGER Ltype(100),Itrans(100),Method(100)
      COMMON/BLK1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
      COMMON/BLK2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax
    1,Pgmin,Esh
      COMMON/BLK3/delPQ,delAV,JACO,BJ
      COMMON/BLK4/Ltype,Itrans,Method
      common/blk5/tapset,Jak,Tmin,Tmax
      open(unit=8,file='trial.dat',status='old')
 find elements of the jacobian
      do 390 i=1,n
      do 395 j=1,n
Ignore slack bus
Find the bus current
      ep=real(E(i))
      fp=aimag(E(i))
se these values to evaluate diagonal elements of Jacobian
       Gpp=real(ybus(i,i))
       Bpp=aimag(ybus(i,I))
       Vp=(ep*ep+fp*fp)
       Jak(i,i)=-Q(i)-(Bpp*Vp)
       Jak(i+n,i)=P(i)-(Gpp*Vp)
       Jak(I,I+n)=P(I)+(Gpp*Vp)
       Jak(i+n,i+n)=Q(i)-(Bpp*Vp)
       if(i.ne.iswing)then
for j1
       JACO(I,I)=-Q(i)-Bpp*Vp
       BJ(i,i)=Jaco(i,i)
       end if
heck for PQ bus or PV bus with violated limits
       If(Ltype(i).eq.3)then
for j3
       JACO(i+n,i)=P(i)-(Gpp*Vp)
```

```
  BJ(i+n,i)=jaco(i+n,i)
r J2
   Jaco(i,i+n)=P(i)+Gpp*Vp
   Bj(i,i+n)=Jaco(i,i+n)
r j4
   JACO(i+n,i+n)=Q(i)-(Bpp*Vp)
   BJ(i+n,i+n)=jaco(i+n,i+n)
  else
   jaco(i,i+n)=0.0
   BJ(i,I+n)=0.0
   jaco(i+n,i)=0.0
   BJ(i+n,i)=0.0
   jaco(i+n,i+n)=0.0
   BJ(i+n,i+n)=0.0
   end if
nd the diagonal elements
   am=real(E(j)*Ybus(i,j))
   bm=aimag(E(j)*Ybus(i,j))
   if(j.ne.i)then
   Jak(i,j+n)=(am*ep)+(bm*fp)
   Jak(i+n,j+n)=(am*fp)-(bm*ep)
   Jak(i,j)=(am*fp)-(bm*ep)
   Jak(i+n,j)=-Jak(i,j+n)
   end if
   if(i.ne.iswing.and.j.ne.i.and.j.ne.iswing)then
 J1
   Jaco(i,j)=(am*fp)-(bm*ep)
   BJ(i,j)=jaco(i,j)
     if((ltype(i).eq.2.and.ltype(j).eq.3).or.(ltype(i).eq.3.
 1and.ltype(j).eq.3))then
 J2
    JACO(I,j+n)=(am*ep+bm*fp)
    BJ(i,j+n)=jaco(i,j+n)
    else
    Jaco(i,j+n)=0.0
    BJ(i,j+n)=0.0
    end if
eck for PV bus
    if(Ltype(i).eq.3)then
or J3
    Jaco(i+n,j)=-(am*ep+bm*fp)
    BJ(i+n,j)=jaco(I+n,j)
    else
    jaco(i+n,j)=0.0
    BJ(i+n,j)=0.0
    end if
    if(ltype(i).eq.3.and.ltype(j).eq.3)then
4
```

```
      JACO(I+n,j+n)=(am*fp-bm*ep)
      BJ(i+n,j+n)=jaco(i+n,j+n)
      else
      Jaco(i+n,j+n)=0.0
      BJ(i+n,j+n)=0.0
      end if
      end if
5     continue
0     continue
        return
        end

     SUBROUTINE KAUSSE(N,L,ksw)
erforms gaussian forward ellimination and backward
ubstitution
     INTEGER N,MXPOS,M
     REAL A(100,100),B(100),X(100),MAX,SUM,D(100,100)
     COMMON/BLK3/B,X,A,D
     if(ksw.eq.1)then
     do 35 i=1,n
     do 35 j=1,n
     A(i,j)=D(i,j)
     continue
     end if
     m=n/2
IVOTAL ORDERING AND FORWARD ELLIMINATION
     DO 70 I=1,N-1
ELECTION OF LARGEST ELEMENT
       MAX=ABS(A(I,I))
       MXPOS=I
       DO 30 J=I+1,N
       IF(ABS(A(J,I)).GT.MAX)THEN
       MAX=ABS(A(J,I))
       MXPOS=J
       END IF
       CONTINUE
NTERCHANGE columns
       IF(MAX.NE.0.0)THEN
       IF(MXPOS.NE.I)THEN
       DO 40 J=1,N
       CALL SWAP(A(MXPOS,J),A(I,J))
       CONTINUE
       CALL SWAP(B(I),B(MXPOS))
       END IF
ORWARD ELLIMINATION
       DO 60 J=I+1,N
       RATIO=A(J,I)/A(I,I)
       DO 50 K=I+1,N
```

```
      A(J,K)=A(J,K)-RATIO*A(I,K)
      CONTINUE
      B(J)=B(J)-RATIO*B(I)
      CONTINUE
      END IF
      CONTINUE
ACK SUBSTITUTION
      goto 123
      n=n-1
      if(A(n,n).eq.0.0)goto 122
      X(N)=B(N)/A(N,N)
      DO 90 I=N-1,1,-1
      SUM =0.0
      DO 80 J=I+1,N
      SUM=SUM+A(I,J)*X(J)
      CONTINUE
      if(A(I,I).eq.0.0)then
      X(i)=0.0
      else
      X(I)=(B(I)-SUM)/A(I,I)
      END IF
      CONTINUE
      RETURN
      END

      SUBROUTINE SWAP(VAR1,VAR2)
waps two variables
      REAL VAR1,VAR2,TEMP
      TEMP=VAR1
      VAR1=VAR2
      VAR2=TEMP
      RETURN
      END

      subroutine steepdnt(n,iswing,conver,treal,timag)
alculates lagrange multipliers from loadflow results
lso calculates the gradient vector
      complex Sgen(100),S(100),Sload(100),E(100),Yvars(100)
      complex Ybus(100,100),Zbus(100,100),Ychar(100,100),Y(100)
      complex Pgen,Pdem
      real
PQ(100),delAV(100),Jaco(100,100),BJ(100,100),Tmax(100)
      real a(100),b(100),c(100),lambda,tapset(100),Tmin(100)
      real Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
      real Qgmin(100),P(100),Pgmax(100),Pgmin(100),Esh(100)
      real gamma(50),alpha(50),theta(50),magv,Pregrad
      real Vtemp(50),Ptemp(50),Temp(50),Dvolt(50),Dtran(50)
```

```
      real Spqmax(100,100),Sgmax(100),Jak(100,100),Dpowr(50)
      integer ltype(100),itrans(100),method(100),Qgmax(100)
      character*12 Bname(100)
      common/blk1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
      common/blk2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax
     1,Pgmin,Esh
      common/blk3/delPQ,delAV,Jaco,BJ
      common/blk4/ltype,itrans,method
      common/blk5/tapset,Jak,Tmin,Tmax
      common/blk6/a,b,c,Spqmax,Sgmax
      common/blk7/Bname
      common/blk8/Vtemp,Ptemp,Temp,Dvolt,Dtran,Dpowr
      common/Blk9/gamma,alpha,delta
      integer n,m,conver
      open(unit=12,file='multi.dat',status='old')
      open(unit=19,file='systate.dat',status='old')
      rewind(12)
      rewind(19)
      m=2*n
      mk=0
      Imax=1
c fill in the vector of cost changes
      print*,'initial penalty weightings: volt,QG,PG1 : '
      read*,wvolt,wQgen,delta
      print*,'multipliers: gamma,alpha,delta:    '
      read*,vfac,Qfac,pfac
      print*
      Print*,'max no. of gradient adjustments:    '
      read*,maxiter
      do 96 i=1,n
      gamma(i)=wvolt
      alpha(i)=wQgen
96    continue
      Pgj=real(Sgen(iswing))
      const=b(iswing)+(2.0*c(iswing)*100*Pgj)
      do 20 i=1,n
      if(i.ne.iswing)then
      Vp=Cabs(E(i))
      delPQ(i)=-const*Jak(iswing,i)
c for voltage limits, penalise for any violation
      if(ltype(i).eq.3)then
      delPQ(i+n)=-const*Jak(iswing,i+n)
c check whether voltage limits violated
      if(Vp.gt.Emax(i))then
c penalty for upper limit violation
      delPQ(i+n)=delpq(i+n)-(2.0*gamma(i)*Vp*(Vp-Emax(i)))
      else if(Vp.lt.Emin(i))then
c penalise for lower limit violation
```

```
  delpq(i+n)=delpq(i+n)-(2.0*gamma(i)*Vp*(Vp-Emin(i)))
lse no penalty
  end if
  end if
dd other terms for violated functional constraints
or slack bus power
  Pg1=real(Sgen(iswing))
  if(Pg1.gt.Pgmax(iswing))then
PQ(i)=delPQ(i)-(2*delta*(Pg1-Pgmax(iswing))*jak(iswing,i))
  if(ltype(i).eq.3)then
PQ(i+n)=delPQ(i+n)-(2*delta*(Pg1-Pgmax(iswing))*jak(1,i+n))
  end if
  else if(Pg1.lt.Pgmin(iswing))then
PQ(i)=delPQ(i)-(2*delta*(Pg1-Pgmin(iswing))*jak(iswing,i))
  if(ltype(i).eq.3)then
lPQ(i+n)=delPQ(i+n)-(2*delta*(Pg1-Pgmin(iswing))*jak(1,i+n))
  end if
  end if
for reactive penalties

  do 25 j=1,n
  if(Ltype(j).eq.2.or.j.eq.iswing)then
check for reactive power violations
  if(Q(j).gt.Qmax(j))then
  delPQ(i)=delPQ(i)-(2.0*alpha(j)*(Q(j)-Qmax(j))*jak(j+n,i))
  if(Ltype(i).eq.3)then
lPQ(i+n)=delPQ(i+n)-(2.0*alpha(j)*(Q(j)-Qmax(j))*jak(j+n,i+n))
  end if
  else if(Q(j).lt.Qmin(j))then
  delPQ(i)=delPQ(i)-(2.0*alpha(j)*(Q(j)-Qmin(j))*jak(j+n,i))
  if(Ltype(i).eq.3)then
lPQ(i+n)=delPQ(i+n)-(2.0*alpha(j)*(Q(j)-Qmin(j))*jak(j+n,i+n))
  end if
  end if
  end if
5 continue

  end if
0 continue
Find the transpose of the jacobian
  do 30 i=1,m
```

```
  do 30 j=i+1,m
  tem=BJ(i,j)
  BJ(i,j)=BJ(j,i)
  BJ(j,i)=tem
  continue
 transposed jacobian plus the rhs vector to find multipliers
voke Gausse to find lagrange multipliers
  print*,'Iam there.........Ureka'
  ksw=1
  call Gausse(m,iswing,ksw)

turns with multipliers in delAV
e the lagrangian multipliers so obtained to compute the
ients
lPQ is again used as a work area
   gradv=0.0
   gradP=0.0
   gradt=0.0
   do 55 i=1,n
eck for a PV bus
   Dvolt(i)=0.0
   if(ltype(i).eq.2.or.i.eq.iswing)then
 is a pv bus
r voltage control
   Vp=cabs(E(i))
itialise gradient
   Dvolt(i)=const*Jak(iswing,i+n)
op to add other terms
   do 45 j=1,n
ditional terms of  voltage gradient
   if(j.ne.iswing)then
   Dvolt(i)=Dvolt(i)+(delAV(j)*Jak(j,i+n))
   end if
   if(ltype(j).eq.3)then
   Dvolt(i)=Dvolt(i)+(delAV(j+n)*Jak(j+n,i+n))
   end if
   continue
heck for violated functional constraints eg. Qvars
dd other terms for violated functional constraints
   do 39 j=1,n
   if(j.eq.iswing)then
   PGj=real(Sgen(iswing))
   if(Pgj.gt.Pgmax(iswing))then
   Dvolt(i)=Dvolt(i)+(2.0*delta*(PGj-Pgmax(j))*jak(j,i+n))
   else if(PGj.lt.Pgmin(j))then
   Dvolt(i)=Dvolt(i)+(2.0*delta*(PGj-Pgmin(j))*jak(j,i+n))
   end if
   end if
```

```
   if(j.eq.iswing.or.Ltype(j).eq.2)then
   if(Q(j).gt.Qmax(j))then
lt(i)=Dvolt(i)+(2.0*alpha(j)*(Q(j)-Qmax(j))*jak(j+n,i+n))
   else if(Q(j).lt.Qmin(j))then
lt(i)=Dvolt(i)+(2.0*alpha(j)*(Q(j)-Qmin(j))*jak(j+n,i+n))
   end if
   end if
   continue

   Dvolt(i)=Dvolt(i)/Vp
   if(Esh(i).eq.Emax(i).and.Dvolt(i).lt.0.0)then
   Dvolt(i)=0.0
   else if(Esh(i).eq.Emin(i).and.Dvolt(i).gt.0.0)then
   Dvolt(i)=0.0
   end if
    if(abs(Dvolt(i)).gt.gradv)gradv=abs(Dvolt(i))
   Dvolt(i)=-Dvolt(i)
   end if
or power source control
se P(I) as a work area
   Pgi=real(Sgen(i))
   Qgi=aimag(Sgen(i))
   SGi=sqrt(Pgi**2+Qgi**2)
   if(Pgi.ne.0.0.and.i.ne.iswing)then
ontrollable power source available
   Dpowr(i)=b(i)+(2.0*c(i)*100*Pgi)-delAV(i)
   if(Pgi.eq.Pgmax(i).and.Dpowr(i).lt.0.0)then
   Dpowr(i)=0.0
   else if(Pgi.eq.Pgmin(i).and.Dpowr(i).gt.0.0)then
   Dpowr(i)=0.0
   end if
    if(abs(Dpowr(i)).gt.gradp)gradp=abs(Dpowr(i))
   Dpowr(i)=-Dpowr(i)
   end if
   if(Ltype(i).eq.3.and.method(i).eq.1)then
   Dtran(i)=0.0
ransformer used to control voltage
ecall the connecting bus
   k=itrans(i)
   tik=tapset(i)
   Vk=cabs(E(i))
   Gpq=real(Ybus(i,k))
   Bpq=aimag(Ybus(i,k))
```

```
      ai=delAV(i)
      bi=delAV(i+n)
      if(k.eq.iswing)then
      ak=const
      else
      ak=delAV(k)
      end if
      if(ltype(k).eq.3)then
      bk=delAV(k+n)
      else
      bk=0.0
      end if
se delPQ as a work area

1=(ai*jak(i,k+n))+(bi*jak(i,k))+(ak*jak(k,i+n))+(bk*jak(k,i))
      tap2=2.0*Vk*Vk*((bk*Bpq)-(ak*Gpq))
      Dtran(i)=(tap1/tik)+tap2
      if(tik.eq.Tmax(i).and.Dtran(i).lt.0.0)then
      Dtran(i)=0.0
      else if(tik.eq.Tmin(i).and.Dtran(i).gt.0.0)then
      Dtran(i)=0.0
      end if
      if(abs(Dtran(i)).gt.gradt) gradt=abs(Dtran(i))
      Dtran(i)=-Dtran(i)
      end if
5     continue
      call cost(n,Ft)
      write(12,49)
9     format(2x,'i  dvolt.  dPGi    dtrans  tapset  Esh  Pgi
ax ')
      do 80 i=1,n
      if(ltype(i).eq.2.or.i.eq.iswing.or.method(i).eq.1)then
      Pgi=real(sgen(i))

ite(12,51)i,Dvolt(i),Dpowr(i),Dtran(i),tapset(i),Esh(i),Pgi,Im
     1ax
      end if
1     format(1x,i3,2x,f8.4,1x,f8.4,2x,f8.4,1x,f8.4,2x,f8.4,1x
      1,f8.4,2x,i5)
0     continue
test for convergence of gradients
       write(12,66)Imax,mk,gradv,gradp,gradt,lambda,Ft
       do 27 i=1,n
       write(12,61)Imax,delta,gamma(i),alpha(i)
1      format(1x,i3,1x,3(1x,f10.2))
7      continue
store partial results in a file
       write(19,81)imax
```

```
  format(1x,'gradient cycle: ',i5)
  write(19,43)
  write(19,82)gradv,gradp,gradt
  format(1x,'Max grad Residuals:  ',2x,'Volt: ',f10.3,2x
1,'PG :',f10.3,2x,'Taps:  ',f10.3)
  write(19,43)
  write(19,83)Ft
  format(1x,'Generating Cost:  ',f10.4)
  format(3x,'Bus code                Generation            Loads
4      Volt.')
  format(' ')
  write(19,76)
  format(1x,'          C U R R E N T  S Y S T E M  S T A T E ')
  write(19,43)
  write(19,122)
  DO 900 I=1,N
  ep=real(E(i))
  fp=aimag(E(i))
  magv=cabs(E(i))
  angle=atan2(fp,ep)
  angle=(angle/3.142)*180.0
  Pgen=100.0*sgen(i)
  Pdem=100.0*sload(i)
  write(19,905)i,Bname(i),Pgen,Pdem,magv,angle,tapset(i)
AT(1X,I3,1x,A12,4X,2(F8.2,1X),3X,f6.2,3x,f6.2,3x,F6.4,2x,F8.4
 12x,f6.4)
  continue
  write(19,43)
  write(19,139)mk
  format(1x,'Number of LF iterations =  ',2x,i4)
radv.le.0.05.and.gradp.le.0.05.and.gradt.le.0.05)return
herwise if gradient not small enough,adjust control
meters
ll subroutine step to find optimum step length
urns with optimum step length, lambda
  prod=0.0
  ncont=0
  do 12 i=1,n
  if(ltype(i).eq.2.or.i.eq.iswing)then
  prod=prod+(Dvolt(i)**2)
e delAV(i) to store the initial value of Esh(i)
  Vtemp(i)=Esh(i)
  ncont=ncont+1
  end if
  if(real(sgen(i)).ne.0.0.and.i.ne.iswing)then
```

```
   prod=prod+(Dpowr(i)*Dpowr(i))
ore the initial value of power gen at bus i
   Ptemp(i)=real(Sgen(i))
   ncont=ncont+1
   end if
   if(ltype(i).eq.3.and.method(i).eq.1)then
   temp(i)=tapset(i)
   prod=prod+(Dtran(i)**2)
   end if
   continue
   prod=sqrt(prod)
se it to adjust the control parameters
uroutine adjust does this. call it
loop around to scale down the gradient vector
    if(prod.ne.0.0)then
    do 14 i=1,n
    if(ltype(i).eq.2.or.i.eq.iswing)then
    Dvolt(i)=Dvolt(i)/prod
    end if
    if(real(Sgen(i)).ne.0.0.and.i.ne.iswing)then
    Dpowr(i)=Dpowr(i)/prod
    end if
    if(Ltype(i).eq.3.and.method(i).eq.1)then
    Dtran(i)=Dtran(i)/prod
    end if
   continue
   end if
    call step(n,iswing,treal,timag,stp)
    call adjust(n,iswing,stp)
    write(12,70)
    format(1x,' imax  Iter  gradv    gradp   gradt  lambda
t')
5   format(1x,i3,1x,i2,1x,f10.3,2x,f10.3,2x,f10.3,2x,f10.3
   1,2x,f10.3)
    use adjusted value of control parameters to find new
erating point
    m=2*n
    if(imax.ge.maxiter)return
    call loadfl(n,iswing,mk,m,treal,timag)
8   Pg1=real(Sgen(iswing))
(Pg1.gt.Pgmax(iswing).and.(Pg1-Pgmax(iswing)).gt.0.001)then
    delta=delta*pfac
    else if
g1.lt.Pgmin(iswing).and.(Pgmin(1)-Pg1).gt.0.001)then
    delta=delta*pfac
    end if
    do 11 i=1,n
```

```
  Vp=cabs(E(i))
  if(Vp.gt.Emax(i).and.(Vp-Emax(i)).gt.0.001)then
  gamma(i)=gamma(i)*Vfac
  else if(Vp.lt.Emin(i).and.(Emin(i)-Vp).gt.0.001)then
  gamma(i)=gamma(i)*vfac
  end if
  if(Ltype(i).eq.2.or.i.eq.iswing)then
  if(Q(i).lt.Qmin(i).and.(Qmin(i)-Q(i)).gt.0.001)then
  alpha(i)=alpha(i)*Qfac
  else if(Q(i).gt.Qmax(i).and.(Q(i)-Qmax(i)).gt.0.001)then
  alpha(i)=alpha(i)*Qfac
  end if
  end if
  continue
op to evaluate gradients
   Imax=Imax+1
   go to 69
   end

   subroutine opstep(n,iswing,treal,timag,stp)
   complex Sgen(100),S(100),Sload(100),E(100),Yvars(100)
   complex Ybus(100,100),Zbus(100,100),Ychar(100,100),YY(100)
   real delPQ(100),delAV(100),jaco(100,100),Bj(100,100)
   real Aa(100),Bb(100),Cc(100),jak(100,100),Tmin(100)
  1,Ptemp(50)
   real Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
   real Qgmax(100),P(100),Pgmax(100),Pgmin(100),Esh(100)
   integer n,m,conver,ltype(100),itrans(100),method(100)
   real,Tmax(100),Qgmin(100),Dtran(50)lambda,h,Vtemp(50)
   real F(20),gamma(50),alpha(50),Dpowr(50),temp(50),Dvolt(50)
   common/blk1/Ybus,Zbus,Ychar,YY,Sgen,Sload,S,E,Yvars
   common/blk2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax
  1,Pgmin,Esh
   common/blk3/delPQ,delAV,jaco,Bj
   common/blk4/ltype,itrans,method
   common/blk5/tapset,jak,Tmin,Tmax
   common/blk6/Aa,Bb,Cc
   common/blk8/Vtemp,Ptemp,temp,Dvolt,Dtran,Dpowr
   common/blk9/gamma,alpha,delta
   open(unit=8,file='opstep.dat',status='old')
   m=2*n
he first thing is to scale down gradients
inds the optimum step length
nitialise step length, i,s
    i=0
    t0=0.01
    call cost(n,Ft)
```

```
    FA=Ft
    write(19,3)Ft
    format(2x,' Initial cost at t0=0.0  is  ',F8.3)
    write(19,41)
    format(2x,'t0    iteration          Cost')
    format(2x,f6.4,3x,i2,3x,f8.3)
djust control parameters at lambda=t0
    t0=t0/2
    t0=t0*2
    i=i+1
    if(i.gt.20)then
    print*,'Too many iterations. Local minimum not found'
    return
    end if
    call adjust(n,iswing,t0)
olve loadflow with adjusted values of control parameters
    call loadfl(n,iswing,jk,m,treal,timag)
eturns with a feasible solution
aluate cost
    call cost(n,Ft)
    write(19,42)t0,i,Ft
    F(i)=Ft
    if(i.eq.1)go to 43
    if(F(i).lt.F(i-1)) go to 43
lse local minimum detected along descent direction
it a quadratic from three known points to find the minimum
    Fb=F(i-1)
    Fc=F(i)
    c=t0
    b=t0/2
    a=0.0
    if(i.gt.2)then
    a=t0/4
    Fa=F(i-2)
    end if
rom these values,find the optimum step length along this
ection
    step1=Fa*(b**2-c**2)+Fb*(c**2-a**2)+Fc*(a**2-b**2)
    step2=Fa*(b-c)+Fb*(c-a)+Fc*(a-b)
    stp=step1/(2*step2)
valuate the cost at this step
    call adjust(n,iswing,stp)
    call loadfl(n,iswing,jk,m,treal,timag)
    call cost(n,Ft)
    Fopt=Ft
    if(Fb.lt.Fopt)then
ake optimum step as b
    stp=b
```

```
      end if
      write(19,86)b,Fb,stp,Fopt
 86   format(2x,'b= ',f6.2,2x,'Fb=   ',F8.3,2x,'step =
',f6.2,2x,'Fopt= ',f8.3)
      return
      end
      subroutine adjust(n,iswing,t)
      complex Sgen(100),Sload(100),E(100),s(100),yvars(1
      complex Ybus(100,100),Zbus(100,100),Y(100),Ychar(1
      real Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
     1,Qgmax(100)
      real Qgmin(100),P(100),Pgmax(100),Pgmin(100),Esh(1
      real delPQ(100),delAV(100),jaco(100,100),bj(100,10
      real Jak(100,100),tapset(100),Tmin(100),Tmax(100)
      real Vtemp(50),Ptemp(50),temp(50),Dvolt(50),Dtran(
     2,Dpowr(50)
      integer ltype(100),itrans(100),method(100)
      common/blk1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars

common/blk2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax,Pg
      common/blk3/delPQ,delAV,jaco,bj
      common/blk4/ltype,itrans,method
      common/blk5/tapset,Jak,Tmin,Tmax
      common/blk8/Vtemp,Ptemp,temp,Dvolt,Dtran,Dpowr
* adjust Voltage and power as per t0
      do 30 i=1,n
      if(ltype(i).eq.2.or.i.eq.iswing)then
* recall the initial value of Esh(i)
      Esh(i)=Vtemp(i)
      Esh(i)=Esh(i)+(t*Dvolt(i))
      ep=real(E(i))
      fp=aimag(E(i))
      ang=atan2(fp,ep)
      if(Esh(i).gt.Emax(i))then
      Esh(i)=Emax(i)
      else if(Esh(i).lt.Emin(i))then
      Esh(i)=Emin(i)
      end if
      ep=Esh(i)*cos(ang)
      fp=Esh(i)*sin(ang)
      E(i)=cmplx(ep,fp)
      Ltype(i)=2
      end if
      Ltype(iswing)=1
      Pgi=Ptemp(i)
      if(Pgi.ne.0.0)then
      Pgi=Ptemp(i)
```

```
  Qgi=aimag(Sgen(i))
  Pgi=Pgi+(t*Dpowr(i))
  if(Pgi.gt.Pgmax(i))then
  Pgi=Pgmax(i)
  else if(Pgi.lt.Pgmin(i))then
  Pgi=Pgmin(i)
  end if
  Sgen(i)=cmplx(pgi,Qgi)
  S(i)=Sgen(i)-Sload(i)
  end if
or taps
  if(Ltype(i).eq.3.and.method(i).eq.1)then
  k=itrans(i)
  tapset(i)=temp(i)
  tapset(i)=tapset(i)+(t*Dtran(i))
  if(tapset(i).gt.Tmax(i))then
  tapset(i)=Tmax(i)
  else if(tapset(i).lt.Tmin(i))then
  tapset(i)=Tmin(i)
  end if
  tik=tapset(i)
  Ybus(i,i)=Ybus(i,i)+Ybus(i,k)
  Ybus(i,i)=Ybus(i,i)+(tik*tik*(-Ybus(i,k)))
  Ybus(i,k)=tik*(Ybus(i,k))
  Ybus(k,i)=Ybus(i,k)
  end if
)  continue
  return
  end
  subroutine cost(n,Ft)
  complex Ybus(100,100),Zbus(100,100),Ychar(100,100),Y(100)
  COMPLEX Sgen(100),Sload(100),S(100),E(100),Spq,g,Yvars(100)
  real Emax(100),Emin(100),Q(100),Qmax(100),Qmin(100)
 1,jak(100,100)
  REAL Qgmax(100),Qgmin(100),P(100),Pgmax(100),Pgmin(100)
  REAL delPQ(100),delAV(100),JACO(100,100),Esh(100)
 2,Tapset(100)
  real a(100),b(100),c(100),BJ(100,100),Tmin(100),Tmax(100)
  real Spqmax(100,100),Sgmax(100),gamma(50),alpha(50)
  INTEGER Ltype(100),Itrans(100),Method(100)
  character*10 fname
  character*12 Bname(100)
  common/blk1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
  common/blk2/Emax,Emin,Q,Qmax,Qmin,Qgmax,Qgmin,P,Pgmax
 1,Pgmin,Esh
```

```
common/blk3/delPQ,delAV,Jaco,BJ
common/blk4/Ltype,Itrans,method
common/blk5/tapset,Jak,Tmin,Tmax
common/blk6/a,b,c,Spqmax,Sgmax
common/blk7/Bname
common/blk9/gamma,alpha,delta
Ft=0.0
do 30 i=1,n
Pgi=real(Sgen(i))
if(Pgi.ne.0.0)then
Ft=Ft+a(i)+(b(i)*100*pgi)+(c(i)*10000*Pgi*pgi)
end if
if(ltype(i).eq.2)then
if(Q(i).gt.Qmax(i))then
Ft=Ft+alpha(i)*(Q(i)-Qmax(i))**2
else if(Q(i).lt.Qmin(i))then
Ft=Ft+alpha(i)*(Q(i)-Qmin(i))**2
end if
end if
if(Ltype(i).eq.3)then
Vmag=cabs(E(i))
if(Vmag.gt.Emax(i))then
Ft=Ft+gamma(i)*(Vmag-Emax(i))**2
else if(Vmag.lt.Emin(i))then
Ft=Ft+gamma(i)*(Vmag-Emin(i))**2
end if
end if
continue
pg1=real(Sgen(iswing))
if(Pg1.gt.Pgmax(iswing))then
Ft=Ft+delta*(Pg1-Pgmax(iswing))**2
else if(Pg1.lt.Pgmin(iswing))then
Ft=Ft+delta*(Pg1-Pgmin(iswing))**2
end if
return
end

subroutine output(n,iswing,k,Lines,Lcap)
complex Spqi,Spqj,C,g,Vi,Vj,current
real ANGLE,MAGV
COMPLEX Ybus(100,100),Zbus(100,100),Ychar(100,100),Y(100)
COMPLEX Sgen(100),Sload(100),S(100),E(100),Yvars(100)
INTEGER Ltype(100),Itrans(100),Method(100)
real tapset(100),Jak(100,100),Tmin(100),Tmax(100)
character*12 Bname(100)
COMMON/BLK1/Ybus,Zbus,Ychar,Y,Sgen,Sload,S,E,Yvars
COMMON/BLK4/Ltype,Itrans,Method
```

```
  common/blk5/tapset,Jak
  common/blk7/Bname
  open(unit=19,file='results.dat',status='old')
  rewind(19)
2   format(3x,'Bus code                Generation          Loads
  3Volt.')
3   format(1x,' p                 Pgen          Qgen      Pload
  Qload  |V|  Ang.')
   format(1x,'____________________________________________________')
   format('')
nt*,'----------------------------------------------------------'
   print 132
   write(19,132)
2   format(1x,' L O A D  F L O W    S O L U T I O N       ')
NT*,'______________________________________________________'
   write(19,41)
   WRITE(19,43)
   WRITE(19,43)
   WRITE(19,83)
   FORMAT(1X,'  N U M B E R   O F   C O M P O N E N T S ')
   WRITE(19,43)
   WRITE(19,72) N
   FORMAT(1X,'            Buses          :          ',3x,i4)
   write(19,73)iswing
   format(1x,'            Swing bus      :          ',3x,i4)
   write(19,74)lines
   format(1x,'            Lines          :          ',3x,i4)
   write(19,75)lcap
   format(1x,'            Cap/reactors   :          ',3x,i4)
   write(19,43)
   write(19,43)
   write(19,39)
   format(1x,' BaseMVA                   :          100 MVA ')
   write(19,43)
   write(19,43)
   write(19,78)
78  format(1x,' Method of solution : Newton Raphson')
   write(19,43)
   write(19,43)
   write(19,41)
   write(19,76)
6   format(1x,'           S T A T I O N   C O N D I T I O N S ')
   WRITE(19,77)
7   FORMAT(1X,' ( Generation,loads and calculated bus
ltages)')
   write(19,43)
```

```
     write(19,43)
     write(19,41)
     write(19,122)
     write(19,123)
     PRINT*
     PRINT 122
     PRINT 123
     DO 900 I=1,N
     ep=real(E(i))
     fp=aimag(E(i))
     magv=cabs(E(i))
     angle=atan2(fp,ep)
     angle=(angle/3.142)*180.0
     sgen(i)=100.0*sgen(i)
     sload(i)=100.0*sload(i)
     PRINT 905,I,Bname(i),Sgen(I),Sload(i),magv,angle
     write(19,905)i,Bname(i),sgen(i),sload(i),magv,angle
5
MAT(1X,I3,1x,A12,4X,2(F9.2,2X),4X,f6.2,3x,f6.2,3x,F6.4,2x,F8.4
0     continue
      print 133
      print 134
      write(19,43)
      write(19,43)
      write(19,41)
      write(19,133)
      write(19,134)
                                                            '
nt*,'________________________________________________________
                                                Line Flows
33    format(1x,'            Bus           :
   1 Losses ')
34    format(1x,' name        To    Name           MW     MVAR     MVA
).u
    2)  MW  MVAR ')
      DO 300 I=1,N
      write(19,41)
      write(19,100)i,Bname(i)
00    format(1x,i3,1x,A12)
      DO 310 J=1,N
      IF(I.NE.J.AND.YBUS(I,J).NE.(0.0,0.0))THEN
      ep=real(E(i))
      fp=aimag(E(i))
      eq=real(E(j))
      fq=aimag(E(j))
qi=(cmplx(ep,-fp))*((-Ybus(I,J))*(E(i)-E(J))+E(i)*Ychar(I,J))
```

```
=(cmplx(eq,-fq))*((-Ybus(i,j))*(E(j)-E(i))+E(j)*Ychar(i,j))
    Ploss=100*(real(Spqi)+real(Spqj))
    Qloss=-100*(aimag(Spqi)+aimag(Spqj))
    Smva=100*cabs(Spqi)
    current=(E(i)-E(j))/(-Ybus(i,j))
    cur=cabs(current)
    POWER=100.0*REAL(Spqi)
    VARS=-100.0*AIMAG(Spqi)
RINT THE LINE FLOW Sp_q
    PRINT 315,J,Bname(j),POWER,VARS,Smva,cur,Ploss,Qloss
5
MAT(8X,I3,1x,A12,2x,F7.2,1X,F7.2,1x,f7.2,1x,f6.2,3x,f7.2,1x,f7
  1.2)
    write(19,315)j,Bname(j),power,vars,Smva,cur,Ploss,Qloss
    END IF
0   CONTINUE
    write(19,41)
0   CONTINUE
    write(19,706)
6   format(2x,'Bus   To   Bus        Controlled Bus     Final Tap

    do 709 i=1,n
    if(method(i).eq.1)then
    Nt=Itrans(i)
    write(19,707)Nt,Bname(nt),i,Bname(i),i,tapset(i)
7   format(1x,i3,1x,A12,3x,i3,1x,A12,7x,i3,7x,f6.4)
    end if
9   continue
NT*,'______________________________________________________________'
     print 139,k
     WRITE(19,43)
     WRITE(19,41)
     write(19,139)k
     WRITE(19,41)
9    format(1x,'Total number of iterations =  ',2x,i4)
nt*,'______________________________________________________________'
     print*,' System data and results contained in file
SULTS.DAT'
     PRINT*,'Type : RESULTS.DAT to view'
     close(19)
     return
     end
```