ANALYSIS AND MANAGEMENT OF OPEN ACCESS POWER TRANSACTION BASED ON AI TECHNIQUES

A DISSERTATION

Submitted in partial fulfilment of the
requirements for the award of the degree
of
MASTER OF ENGINEERING
in
ELECTRICAL ENGINEERING
(With Specialization in Power System Engineering)

By

SUNIL VERMA

DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITY OF ROORKEE
ROORKEE–247 667 (INDIA)

MARCH, 2000

CANDIDATE'S DECLARATION

I hereby declare that the work which is being presented in this dissertation entitled **"ANALYSIS AND MANAGEMENT OF OPEN ACCESS POWER TRANSACTION BASED ON AI TECHNIQUES"** in the partial fulfillment of the requirements for the award of the Degree of **Master of Engineering in Electrical Engineering** with specialization in **Power System Engineering** submitted in the Department of Electrical Engineering, University of Roorkee, Roorkee, is an authentic record of my own work carried out under the kind guidance of **Dr. N.P. Padhy,** Department of Electrical Engineering, University of Roorkee, Roorkee.

The matter embodied in this dissertation has not been submitted by me for the award of any other degree.

Dated : 1.3.2000

Place : Roorkee

(SUNIL VERMA)

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

01/03/2000

(Dr.N.P. Padhy)

Department of Electrical Engineering
University of Roorkee
Roorkee -247667 (U.P.)
(INDIA)

ACKNOWLEDGMENT

It is my pleasure to express my deep sense of gratitude and thanks to my guide Dr. N.P. Padhy, Lecturer, Electrical engineering Department, University of Roorkee for helping me in guidance and towards constant support, whenever I faced problems during this course of my dissertation inspite of his busy academic schedule.

I thank to the Mr. Bharat Gupt O.C., Power System Simulation Lab, Electrical Engineering Department, University of Roorkee for providing computer facilities for simulation of this dissertation work.

I also thank the Almighty, faculties of power systems group and friends for being with me whenever I was in need.

Dated : 1. 3. 2000

Place : Roorkee

(SUNIL VERMA)

ABSTRACT

The present scenario of power transmission in a power system is more or less flexible. Each alternative of open access to power transfer offers different solutions with both economic and secure operation. This thesis presents the above study using Evolutionary Programming based Optimal Power Flow (EP-OPF) algorithm. The proposed algorithm is useful to optimize the generation cost satisfying the network constraints and provide options for open access. The algorithm of open access power transaction considers non-utility generators (NUG), which is agreed to supply the power to the network to meet the increased load demand. The proposed algorithm considers load increment at each and every load point and comment on the best transaction. The proposed analysis offers different transaction options. The selection of any option is based on the optimal cost and security constraints. Validity of the proposed algorithm is tested on IEEE-30 bus test system with and without NUG.

NOMENCLATURE

N_{bus}	=	No. of buses in the network
N_{gen}	=	No. of generator connected in the network
$F_i(Pg_i)$	=	Generation cost of i^{th} generator for Pg_i generation
Pg_i	=	Real Power generation of generator i
Pg_{slack}	=	Real power generation of slack generator
P_i^{net}	=	Net real power injected at bus i
Q_i^{net}	=	Net reactive power injected at bus i
P_k^{net}	=	Real power generation of generator k
Q_k^{net}	=	Reactive power generation of generator k
Pg_i^{min}	=	Minimum value of generation by generator i
Pg_i^{max}	=	Maximum value of generation by generator i
θ_i	=	Phase angle at bus i
$\|E_i\|$	=	Voltage magnitude at bus i
$\|E_k\|^{sp}$	=	specified voltage at generator bus k
V_i	=	Voltage at bus i
V_i^{min}	=	Minimum voltage limit at bus i
V_i^{max}	=	Maximum voltage limit at bus i
MVA_{ij}	=	Power flow in the line connected between bus i and j
MVA_{ij}^{min}	=	Minimum limit of power flow in the line connected between bus i and j

MVA_{ij}^{max}	=	Maximum limit of power flow in the line connected between bus i and j
Pd_i	=	Load demand at bus i
P_D	=	Total load demand of the network
Pl	=	Transmission losses when utility supplying the load
bit_reqd_i	=	No. of bit required for generator i
$Cost_i$	=	Generation cost of i^{th} chromosome
p_c	=	Probability of crossover
p_m	=	Probability of mutation
p_i	=	Probability of selection of i^{th} chromosome
q_i	=	Cumulative probability of i^{th} chromosome
Pd_j'	=	Increased load demand at load point j
Pg_{ij}'	=	Real power generation of i^{th} generator when load increases at jth load point and load supplied by existing utility only
Pg_{ij}''	=	Power generation by the i^{th} generator when load increment takes place at load point j, NUG supplying constant load Pd'
Pg_{ij}'''	=	Power generation by i^{th} generator when the load increment takes place at load point j and increased load supplied by both existing utility and NUG.
Pg_{NUGj}^{allow}	=	The allowable generation for the NUG when load increment takes place at load point j
$Pd_j^{required}$	=	Required load demand at load point j
$Pg_j^{high\_cost}$	=	Generation of pseudo generator connected at load point j

Pg_{NUGj}''	=	Power generation by NUG when supplying load with existing utility and load increment at load point j
Pl_j'	=	Transmission losses when load demand increases at j^{th} load point and load supplied by existing utility only
Pl_j''	=	Transmission losses when load demand increases at j^{th} load point and increased load supplied by NUG only while existing utility set at previous optimal point
Pl_j'''	=	Transmission losses when NUG supplying constant load Pd' for load increment at j^{th} load point
Pl'^{v}	=	Transmission losses when load demand increases at j^{th} load point and the increased load supplied by both existing utility and NUG.
C	=	Optimal cost of generation when existing utility supplying load
C_j'	=	Optimal cost of generation when load demand increases at j^{th} load point and only existing utility supplying load
C_j''	=	Cost of generation of NUG only, when existing utility set at its previous optimal point.
C_j'''	=	Optimal cost of generation of utility when NUG providing fixed generation Pd_j'
$C_j'^{v}$	=	Optimal cost of generation for combined operation of utility and NUG, when load increment takes place at j^{th} load point

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION TO OPTIMAL POWER FLOW (OPF)

OPF is a generic term that describes a broad class of problems in which we seek to optimize a specific objective function which satisfying constants dictated by operational and physical particulars of the electric network. Conventional OPF formulation aims to minimize the operating cost of thermal resources subjected to constraints such as active and reactive power balances at each and every bus, real and reactive power limits, voltage limits, transmission power flow limits etc.

Optimal power flows attempt to find the best possible setting for a list of control variables such that a desired objective is met, sometimes a weighted composite objective function may be formed to minimize losses and at the same time minimize VAR additions.

Some optimal power flows also model system security constraints, which set the optimal control settings such that the system can "survive" a specified list of contingencies. A contingency is defined as a set of system component outage (e.g. live, bus or generator outage combinations). "Survival" means that emergency limits (e.g., voltage and line flow limits) are not exceeded in any of the contingency cases. It also refers to maintain steady-state stability, which is implied by a solved ac load flow.

The objective function of the OPF is to minimize the cost of generation and network losses, satisfying the equality and inequality constraints.

Mathematically,

$$Min.f(x,y)$$

Subjected to

$$g(x,y)=0$$

$$h(x,y)\leq 0$$

Where x is the unknown state vector defined as: -

$$x = \begin{vmatrix} \theta_i & \}\text{on each PQ bus} \\ |E_i| & \\ \theta_i & \text{on each PV bus} \end{vmatrix}$$

Another vector, y is defined as: -

$$y = \begin{vmatrix} \theta_i & \\ |E_i| & \}\text{on each reference bus} \\ P_k^{net} & \\ Q_k^{net} & \}\text{on each PQ bus} \\ P_k^{net} & \\ |E_k|^{sp} & \}\text{on each QV bus} \end{vmatrix}$$

The vector y made up of all the parameters that must be specified. Some of these parameters are adjustable (for example generator output P_k^{net} and the generator bus voltage). Some of the parameters are fixed, as far as the OPF calculation is concerned, such as P and Q at each and every load bus. To make the distinction, y vector can be divided into two parts u and p:

$$y=\begin{bmatrix} u \\ p \end{bmatrix}$$

Where u represents the vector of adjustable variable, and p represents the fixed or constant variables.

The function f (x,y) includes the total cost of generation and network losses,

$$f(x,y) = \sum_{i=1}^{all\_gen} F_i(Pg_i)$$

The equality constraint g (x,y) = 0 provides a set of equations that governs the power flow:

$$g(x,y) = \begin{vmatrix} P_i(|E|,\theta_i) - P_i^{net} \\ Q_i(|E|,\theta_i) - Q_i^{net} \\ P_k(|E|,\theta_i) - P_k^{net} \end{vmatrix} \begin{matrix} \}\textit{for}\text{ each PQ (load) bus i} \\ \textit{for}\text{ each PV (generator) bus k, not including the reference bus} \end{matrix}$$

The h (x,y) are the inequality constraints on dependent and independent variables:

1.2 INTRODUCTION TO OPEN ACCESS POWER TRANSACTION:

The impact and new challenges posed by deregulation have received extensive attention in recent years. In this new environment, one common problem has been encouraged, namely the market activities in electricity trading can exert unprecedented and serve pressure on the existing transmission system. Such networks were originally designed to accommodate certain generation / load patterns (e.g. favoring larger and more economic units). Under deregulation the generation patterns resulting from market activities can be quite different from the traditional one, possibly worsening flow congestion and security margins. Further more, since any new generator in the system can sell all or part of its output to single or multiple buyers located anywhere within the point of view, the aim is to optimize power transactions in order to maximize the benefit of pool operation. These benefits are maximized for a particular market based on the electricity prices, of generation offered to the system and by minimizing the total operation cost of the system.

Normally the generation cost characteristics curve for the utility can be expressed as a second order polynomial $C = a + b\,P + c\,P^2$ £/hr The cost curve for a utility in the example system is shown in fig. 1.1.

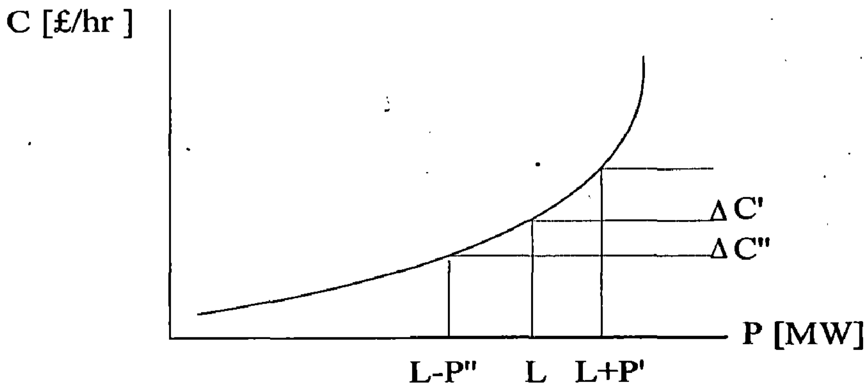

Fig . 1.1 Total operation cost for utility

At the local load level utility is operating at the marginal cost level:

$$MC = \lambda_0 = \frac{dC}{dP} = b + 2cL \quad [\text{£/hr}]$$

This cost represents the minimum increment in the total generation cost of utility to supply an additional unit of generation over the utility's load level L. The marginal cost is often used to define the level of import / export power systems, for a load level of L, the utility is willing to add an incremental P' to its generation and sell it to interconnected system if the benefit associated with the transaction is greater than or at least equal to:

$$\Delta C' = C(P'+L) - C(L)$$

The ratio of the incremental cost of utility to the incremental power is denoted as the incremental price of electricity and computed as

$$price\pi = \frac{\Delta C'}{\Delta P} = \frac{a + b(P'+L) + c(P'+L)^2 - (a + bL + cL^2)}{(P'+L) - L}$$

$$\pi = \lambda_0 + CP' \quad [\text{£/hr}]$$

The utility will sell P′ to the interconnected system by increasing the local generation, at a price at least equal to π, and receiving P′π from the interconnected system for the export. In the same way, the utility may buy P″ from the interconnected system of the cost associated with the transaction is lower or at most equal to ΔC″. The maximum price to pay for the proposed transaction is computed as: -

$$\pi = \lambda_0 - CP'' \quad [\text{£/hr}]$$

In Figure 1.2, the marginal cost curve (MC) and the incremental price curve (π) as well as a possible price curve (λ) in utility are plotted as function of generation level in the utility. The incremental price curve will provide the utility with the maximum (minimum) price to pay for (receive from) a purchase (sell) of an amount of power T.

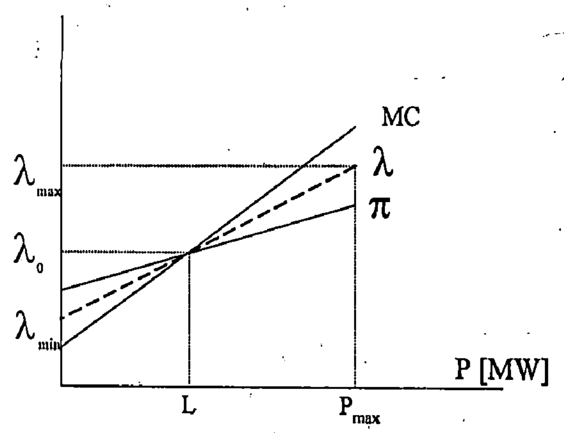

Fig. 1.2 Price curve in utility

The utility will increase its generation level from the local load L if the market price of each unit of extra power is greater than λ_0. If the price is lower than λ_0, the utility will import power from the network to supply its local load.

After transactions are defined, the utility will generate P (i) = L (i) + T (i). If T (i) is positive the utility is selling power and receiving λ(i) T (i) as a transaction payment, conversely if T (i) is negative the utility is buying power and the transaction payment - λ(i) T (i).

CHAPTER 2

LITERATURE SURVEY

2.1 LITERATURE REVIEW ON OPF PROBLEM

A wide variety of optimization techniques have been applied to solve OPF problems. The techniques can be classified as: -

1. Non linear programming (NLP)
2. Quadratic programming (QP)
3. Newton based solution of optimality conditions
4. Linear programming (LP)
5. Hybrid version of linear programming and integer programming
6. Interior point method
7. Evolutionary programming based method (EP)

NONLINEAR PROGRAMMING (NLP)

This is the earliest formulation category since it nicely within the framework presented by physical models of the electric network.

Carpentier [1] first introduced a generalized, nonlinear programming formulation of the economic dispatch problem, including voltage and other operating constraints.

Dommel and Trinney [3] developed a NLP method to minimize fuel cost and active power losses using the penalty function optimization approach. This NLP method checks the boundary on using a lagrange multiplier approach, and is capable of solving large size power system problems up to 500 buses. The method's limitation is in the modeling of components such as transformer taps which are accounted for in the load flow and not in the optimization routine.

Shen and Laughton [1] presented a method which implements an iterative indirect approach based on Lagrange -Kuhn -Tucker conditions of optimally for solving power systems problems. This method was validated on a sample 135 kV British system of 270 buses and was applied to solve economic dispatch objective function with constraints, which satisfy necessary condition using the Kuhn -Tucker conditions. The constraints include voltage levels, generator loading, reactive source loading and transformer loading. This method requires less computational time when a tolerance of 0.001 was chosen.

El. Abaid and Jaimes [1] presented a general formulation of the economic dispatch OPF problem and used a non-linear programming technique, which employed the Lagrange multiplier approach for handling inequality constraints. The method was developed for real power and voltage magnitude dispatch optimization.

Sasson [1] extended Dommel and Tinney work where he tried to improve convergence of Newton based approach. This work minimizes the cost of fuel and transmission line losses by implementing a non-linear programming technique, which employs the Powell and Fletcher -Powell algorithms. This work checks the convergence at every stage of optimization process. However, because the method is similar to the Lagrange and Kuhn -Tucker methods, the issue of wondering phenomena can mar the efficiency of the method. This method was tested on the synthetic IEEE 30 bus system and is limited by being incapable of handling more than two constraints per node.

Alsac and Stott [1] presented a non-linear programming approach, which was tested on a reduced gradient method utilizing the Lagrange multiplier and penalty function techniques. This method minimizes the cost of total active power generation problems and incorporates steady state security and insecurity constraints. This method was validated on the 30 bus IEEE test system and the solution was found within 14.3 seconds. The correct choice of gradient step sizes was crucial to make algorithm successful.

Billinton and Sachdeva [1] presented a non-linear programming approach using the Powell and Fletcher -Powell algorithm and included the penalty factor approach. The objectives considered by this method were calculation of real and reactive power losses and cost of real and reactive power dispatch. This method was illustrated on a synthetic system based on a reduced model of the Saskatchewan Power Corporation System. This algorithm handles mixed hydro -thermal cost function and non-linear thermal input cost functions.

Baralo [1] presented a non-linear optimization approach using complex Hassian matrix approximation (Diflex Hessian) for exact real -time optimal dispatching with security constraints. This method was designed for on line operation and demonstrated on 1200-bus system.

Housos and Irisarri [1] presented a method which employs a variable matrix technique and the algorithm employed a sparsity coding to improve the Hessian matrix instead of using full Hassian, and was also based on the Broydn-Fletcher-Goldford-Shanno (BFGS) and Darrion-Fletcher-Powell (DFP) methods. This method was used to solve power flow which makes it optimum and validated on 14 and 118 bus system. The algorithm perform well for small systems such as 14 bus system failed to perform with the same accuracy for large scale problems such as 118 bus system.

Shoults and Sun [1] presented an optimal power flow problem, which was based on the real and reactive (P/Q) decomposition algorithm. The P- problem involves the minimization of hourly production costs by controlling generator real power outputs and taps setting on phase shifter. The Q- problem involved the minimization of real transmission of real transmission losses by controlling generator terminal voltages, transformer tap setting and shunt capacitor/reactors. Both P and Q problems include static security constraints such as voltage limits, line flow capacity rating and generator reactive power limits. A non linear optimization strategy based upon the gradient method employing the sequential strategy based upon the gradient method employing the sequential unconstrained minimization technique (SUMT) was developed. An outside -in penalty function is defined, to force the dependent function to be feasible at the optimal point. A 5 -bus system was used to demonstrate the P- and Q- sub problems. This method has the capability of solving large systems as 1500 bus and 2500 transmission lines. It was actually tested on a practical 962 bus system was solved in 46 seconds and 5 bus system took 2.3 seconds.

Divi and Kesavan [1] presented a shifted penalty function approach which overcomes the ill conditioning of Hessian of the Penalty function method for solving constrained non linear programming problems. The method exploited a reduced gradient concept and adapted Fletcher's Quasi- Newton technique for optimization of shifted penalty functions which further improves the convergence and accuracy. In this method, the OPF variables were decomposed into an independent set "x" which consists of the generator bus variables and a dependent set "y" which consists of the load bus voltage variables and the equality constraints are not included in the shifted penalty function but are used to obtain the reduced gradient of the penalty function. The objective function was a scalar valued function and can be either cost of fuel generation or transmission line losses, and the constraints used in the formulation were equality constraints which are the load demand equations and the inequality constraints are real and reactive power for both load and generation and voltage magnitude. The choice of the swing bus is critical to the solution of the problem. This method was validated on three synthetic systems with an 11-bus system being the largest. The method saves 30 percent of the computational time over standard penalty function methods.

Talukdar [1] presented a Quasi -Newton (variable matrix) method for solving general NLP optimal power flow problems. The method is attractive due to the following reasons: (1) it can accommodate OPF constraints in a straight forward manner, (2) it is robust and will attain a feasible solution from infeasible initial starting points and (3) it appears to be very fast. This method finds an optimal solution by using the Berna, Locke, and Westerberg (BLW) decompositions technique. A large 1000 bus system is partitioned into small systems and is capable of being handled on relatively small machines of 2MB and limited to the 25-bus system.

Mamoh [1] presented a non-linear programming technique, which satisfies the extended Kuhn -Tucker conditions (EKT) using simplex -like algorithm, a generalized sensitivity method using differentiation and a generalized sensitivity method using eigen values. This method was validated on 118 bus systems.

Lin [1] presented real time economic dispatch method based on penalty factor obtained from the base case solution .The method was validated on a 14 bus system to demonstrate the feature of the algorithm. A classical economic dispatch Lagrange multiplier approach (by calculating penalty factors) was employed using a two-phase solution strategy. Phase 1 solved the initial problem and phase 2 used the solution from phase 1 as input.

Rehan [1] presented a voltage optimization algorithm using Quasi Newton method with the same convergence properties as the Han-Powell method. This method decides which constraints are active and which are not. This was limited in practice to a synthetic 14-bus system. This approach contains a scheme for detecting unfeasibility and the priority listing depends on the heuristics, which depends upon the operator's experience. This algorithm did not include transformer taps in its formulation, and the optimization scheme was not include with power flow as a constraint.

Hobiballahzadeh [1] presented an algorithm, which exploits zoutendijk's method for solving non-linear optimization problems. The sparisity and the embedded network structure of the constraints are utilized to speed up the solution technique, and the method of parallel tangent is used to speed up the convergence of non-linear technique. The method was tested on 5, 39 and 118 bus systems. The method is capable of improving convergence from one stage to the other stage and CPU time of this method is reduced because of sparsity coding, however CPU time requirements increase with system size, and deviation from the operating points may also cause problems because optimization is performed around the operating point.

Ponarajah and Galina [1] presented a continuation method (homotopy method) to solve non-linear programming optimization problems. This method was used to solve a minimization of fuel cost function problem, which has a quadratic objective function and linear constraints. The method was tested on 6, 10, 30 and 116 bus systems. It has been claimed that the method was found to be faster then the methods that rely on heuristics and methods that takes unfeasibility.

QUADRATIC PROGRAMMING

Quadratic programming is a special form of non-linear programming whose objective function is quadratic with linear constraints. Several QP methods in this category have been used to solve OPF (loss, voltage and economic dispatch) type of problems.

Reid and Hasdorf [1] presented a quadratic programming method specialized to solve the economic dispatch problem, which does not require penalty functions or the determination of gradient step size. The method was developed purely for research purposes, therefore, the model used is limited and employs the classical economic dispatch with voltage, real and reactive power limit and it was tested on 5, 14, 30, 57 and 118 bus systems. The CPU time required was very reasonable, however the time increases with increased system size.

Wallenberg and Stadin [1] presented another significant contribution where two optimization processes for solving the economic dispatch objective were compared. The two methods are based on the Dantzig-Wolfe algorithm and quadratic formulations. The proposed method was capable of handling practical components of a power system and the optimization routine was attached to the power flow with no area interchange. This method was tested on a practical 247 -bus system. The model solves the contingency constrained economic dispatch objectives and serves as one of the pioneering works of the decomposition algorithm for economic dispatch.

Giras [1] presented a quadratic programming approach, which employs a Quasi Newton technique based on Han-Powell algorithm. This algorithm provides a solution even from an infeasible initial starting point. Hence it was tested on small synthetic systems. The method appears to be fast because of its power flow super linear convergence qualities. The method converges rapidly for small-scale problems, however convergence criteria do not seem to be practical.

Burchett [1] presented a quadratic programming which solves four objective functions including fuel cost, active and reactive losses, and new shunt capacitors. The algorithm and the accompanying software were claimed to be a technology breakthrough, since the method is capable of solving up to 2000 buses on large mainframe computers with a computational time of five minutes. The economics dispatch method OPF problem in the method is much more complex than the classical economic dispatch problem.

Aoki and Satoh [1] presented an efficient method to solve the economic dispatch problem with DC load flow type network security constraints. This method employs a simplex approach parametric quadratic programming (PQP) method to overcome the problem of dealing with transmission losses as a quadratic form of generator outputs. The constrained employed are generator limits, branch flow limits, and transmission line losses. However because of many large bound variables, a pointer is employed to reduce the number of variables to the number of generators. The method was validated on a non-practical 10-bus system and CPU time of 0.2 -0.4 seconds was obtained for all cases studied.

Contaxis [1] presented a method, which solves the optimal power flow problem by decomposing the problem into two sub- problems, a real and a reactive sub problem. The OPF solution is formulated as a non-linear constrained optimization problem, recognizing system losses, operating limits on the generation units, and line security limits. This method employs an optimization technique called Beale's method, which is used for solving quadratic programming with linear constraints. The efficiency of this method is guaranteed by solving the real sub problem and using this result is used as input to the other subproblem until the full problem is solved. The result of the method was tested on a 27-bus system.

Talukdar [1] presented a quadratic programming method, which employs the Han Powell algorithm. This technique uses Berna, Locke and Westerberg (BLW) technique, which is as field under decomposition and parallel programming. This method was validated on a practical size hypothetical system of 550 and 1110 buses, and can be used to solve systems of 2000 buses or greater. The method's formulation reduces the problem to a quadratic programming form and however the process of step size selection is not fully accomplished in the method. Constrained economic dispatch is not handled by the method but the algorithm can be easily extended to do so.

Burchett [1] presented a modification of his work, which solves four objectives, functions including fuel cost, active-reactive losses and new shunt capacitors. The method was capable of obtaining a feasible solution from an infeasible starting point by creating a sequence of quadratic programming which converge to the optimal solution of the original non-linear problem. This method is capable of obtaining a feasible solution even if power flow divergence is obtained.

El-Kady [1] presented a proposal to solve the OPF problem for voltage control based on Quadratic programming algorithm. The method was applied to the Ontario hydropower System and was based on the variation of the total system load over a 24-hour period. The constraints included tap changer real and reactive generation, transformer taps. This method was also tested on 1079 bus system on an IBM 3081 mainframe computer within a solution time of approximately 7 minutes.

Aoki [1] presented a method, which was an efficient, practical and definitive algorithm for dealing constrained load flow (CLF) problems. This method has a procedure for control variable adjustment with help of Quasi-Quadratic programming formulation. The method has a step size approach to ensure convergence and also maintain priority among the constraints such as power, voltage and techniques.

Papalexopoulos [1] illustrates that proper implementation of second order OPF solution method maintain robustness with respect to different starting points. It was concluded that the decoupled problem is good for large problems and the method improves computation times by three or four folds. The method was tested on a practical 1549 bus system.

NEWTON RAPHSON BASED CATEGORY

Rashed [2] presented a method, which has employed non linear programming approach based on homotopy continuation algorithm for minimizing loss and cost objective functions. The method introduced an acceleration factor to calculate the update controls. The method was validated on an actual 179-bus system.

In addition Happ [2] presented a method which has used Lagrange multipliers in economic dispatch objective function. The method is based on Newton Raphson load flow and used Jacobian matrix to solve for incremental losses. The method was tested on 118 bus taken and the results were compared with other approaches. The algorithm is good for both on line and off line operations, and contingency studies were performed using this method.

Sun [2] presented a Newton based optimization technique for solving reactive power optimization. This method solved a quadratic approximation of the Lagrangian at each iteration and it has been tested for 912-bus system. The zigzagging phenomenon used in this method is comparable to the other conventional OPF techniques.

Pererira [2] presented a method, which solves an economic dispatch problem with security constraints using a decomposition approach. This method solved the following types of dispatch problems: the pure economic dispatch problem. The security constrained dispatch problem, and the security-constrained dispatch with rescheduling problem. The method linearized AC/DC power flows and performed sensitivity analysis of load variations. Practical testing of the method was performed on the Southern Brazilian system with encouraging results.

Sanders and Manroe [2] presented an algorithm for security constrained dispatch calculations. The method is good for real time on line use and was tested on a 1200-bus 1500 line practical power system. The method is referred to as a constrained economic dispatch calculation (CEDC) and was designed to meet the following objectives such as provide economic base points to load frequency control (LFC) promote reliability of service by respecting network transmission limitations, provide constrained participation factors, and be usable in present control computer systems. The security constraints were linearized and this requires the calculation of constraint -sensitivity factors. The load flow was not used as a constraint but it was used to simulate the periodic incremental system losses.

Monticelli [2] presented a framework for solving the economic dispatch problem with security constraints. The algorithm was based on mathematical programming decomposition. This technique allows the iterative solution of a base case economic dispatch and separate and contingency analysis with generation rescheduling to estimate constraint violation. Monticelli's method was tested on the IEEE 118 bus test system, and the specific dispatch problem solved were the pure economic dispatch problems, the security constrained dispatch problem, and the security constrained dispatch problem with rescheduling. These methods include preventive control actions and indicated an automatic way of adjusting the controls. The Bender's decomposition algorithm employed in this paper can only guaranteed under some convexity assumptions and required that each decomposition can be feasible for a feasible solution.

LINEAR PROGRAMMING CATEGORY

Wells [2] developed a linear programming approach to determine an economic schedule which is consistent with network security requirements. The cost objective and its constraints were linearized and solved using simplex method. The limitation of this method are the final results for an infeasible situation obtained may be optimum and rounding errors caused by optimal digital computers may cause constraints to appear overloaded.

Shen and Laughton [2] presented a dual linear programming technique. This method was tested on a 23-bus non-practical power system. Single line outage was considered in the problem formulation and comparative studies using non-linear programming were done. This method has been done well tested and has shown promising results.

Stott and Hobson [2] presented a series of paper applying linear programming to solve practical power systems problems. This method handled the calculation of control actions for relieving the network overloads dealing emergency state. This method includes six objective functions, which were prioritized and the method was extended to handle load shedding. The method is capable of handling high voltage taps, large sized systems and the method also handles unfeasibility-using heuristics. This method appears very efficient for the systems tested. It is limited to linear objective functions and it is recommended for the problems with quadratic objectives.

Stott and Marinho [2] presented a linear programming approach using a modified revised simplex technique for security dispatch and emergency control calculations on large power systems. The method accommodate multi segment generator cost curves and employed sparse matrix techniques, It was developed on an IBM 370 -158 and tested on 30 and 126 bus systems. The method included practical components such as transformer tap settings; the result were obtained within a reasonable time frame and appeared to be efficient.

Stadlin and Fletcher [2] presented a research paper, which has employed a model for voltage/reactive dispatch and control. The method provided a network modeling technique that shows the effect of reactive control of voltage. The OPF problem was solved using a linear programming technique. An advantage of the method is that the typical load flow equation can be decomposed into reactive power and voltage magnitude. The method provides the modeling of other devices such as current models, transformer taps, incremental losses and sensitivity of different models. The method was demonstrated on a 30 bus IEEE test system and the efficiency of the voltage/VAR model is dependent on how well the load characteristics can be estimated and how well the external network can be modeled.

Irving and Sterling [2] presented a linear programming approach to solve the economic dispatch of active power with constraints. The size of the system, which may be simulated on computer used, was restricted by the analog /hybrid nature of computer; however the method was capable of solving up to 50 generation and 300 node systems.

Houses and Irisarri [2] presented a Quasi Newton linear programming approach, which employed a variable weight technique and incorporated multi objective functions. This method employs sparsity coding to improve Hessian matrix instead of full Hessian matrix and linearized constraints are treated as a set of penalty function with variable weight coefficients. It was validated on 14 and 118 bus system and appears to perform well for small size system as compared to well-known methods.

Farghal [2] presented an approach for real time control of power system in an emergency state. A set of control actions based on the optimal re -dispatch function was applied to correct the insecure system operating conditions using sensitivity parameters.

This method is good for solving transmission line overloaded problems. Ramp rate constraints were used in the method's formulation and a classical dispatch along with fast decoupled load flow was employed. The method was tested on a 30-bus system for different load models and is found to be suitable for on-line operation.

Mota -Palomino and Quintana [2] presented a non-conventional linear programming technique involving a piece wise differentiable penalty function approach. This method solved contingency constrained economic dispatch (CED) objectives with linear constraints. The method was validated on a 10, 23 and 118 bus system. The descent direction depends on whether, at a certain point, the pseudo-gradient of the penalty function is a linear combination of the columns of the active set matrix or not, and the method optimal size was determined by selecting so that the active constraints remain active or feasible, and hence, only inactive constraints were considered to determine a step size. In all the cases studied, the method takes less iteration to get optimal solution than standard primal simplex techniques.

Moto-Palomamino and Quintana [2] presented a penalty function linear programming based algorithm to solve reactive power dispatch problems. The method used a criterion to form a sparse reactive power sensitivity matrix, which was modeled as a bipartite graph and its efficient constraint relaxation strategy is used for linearized reactive dispatch problem. This method allowed several constraint violations and can handle unfeasibility by finding the closet point to a feasible point. The method is capable of handling large system sizes based on sensitivity matrix (bipartite graph). The reactive power dispatch problem was made up of various function which include: (1) a vector of costs associated with changes in generated voltages at voltages controlled nodes (2) a vector of cost associated with changes in shunt susceptance connected to the nodes of the system, and (3) a vector of cost associated changes in transformer turn ratios. This method was tested on a 256 node 58 voltage controlled node interconnected Mexcian system. The sensitivity matrix was used to decide which constraints are binding.

Santos-Nieto and Quintana [2] presented a linear programming technique for solving linear reactive power flow problems. The main objectives are real power losses, load voltage deviation, and feasibility enforcement of violated constraints. A penalty function linear programming algorithm was implemented to handle unfeasibility. This method was validated on 253-bus Mexican test system.

MIXTURE OF LINEAR PROGRAMMING AND QUADRATIC PROGRAMMING CATEGORY

Nabona and Ferris [2] presented a method, which involved quadratic and linear programming for optimizing the economic dispatch objective. The minimum loss problem was solved using a linear programming approach and the minimum cost and reactive power problems were solved using either a quadratic or a linear programming approach. The technique is embedded in a Newton Raphson power flow program with limit on line flows and other constraints, which can be implemented easily in the formulation. This algorithm does not need to start at a feasible non-optimal point and validated on 14,30,57 bus systems. This approach may be feasible for online application and avoids the difficulties associated with the gradient method optimization approach.

Contaxis [2] presented a method to solve the optimal power flow problem by decomposing it into two sub problems: the real and reactive sub problems. The method employed both linear and quadratic programming, depending upon the type of problem solved. A quadratic programming was used to solve the two sub problems at each iteration, and a linear programming approach was used if the valve point loading was to be considered. The method solved fuel cost and system losses and linearized the non-linear constraints using z-matrix technique and sensitivity analysis. Line flows were expressed as a function of generator outputs by utilizing generalized generation distribution factor (GGDF)

INTERIOR POINT CATEGORY

Even though the interior point method was devised in early to mid 1980 s, its application to power system optimization problems begins slightly later. Clements [2] presented one of the first interior point research studies applied to power systems. Clements presented a non-linear programming interior point technique for solving power system state estimation problems. The method used a logarithmic barrier function interior point method to accommodate inequality constraints, and Newton's method to solve Karush-Kuhn-Tucker (KKT) equations. The method has the advantage of solving the problem in considerably fewer iterations as compared to linear programming techniques, were the no. of iteration becomes system dependent. This method was tested for a 118-bus system including 6, 30, 40 and 53 bus systems with favorable results. The choice of starting points was limitation of the method.

Ponnambalon [2] presented a newly developed dual affine (DA) algorithm (a variant of Karmakar's interior point method) to solve hydro-scheduling problem. The hydro scheduling problem was formulated as a linear programming problem with equality and inequality constraints. The number of iterations required to solve large-scale problems is relatively small and is generally between 20-60 iterations irrespective of the size of the problem. This algorithm is suitable for large numbers of constraints and is applicable to linear and non-linear optimization problem. This largest problem solved comprised of 880 variables and 3680 constraints and the algorithm has been implemented considering sparsity of the constraint matrix. This method was tested on up to 118-buses with 3680 constraints, and it was discovered that the dual affine algorithm is only appropriate for a problem with inequality constraints.

Vergas [2] presented an interior point (IP) method to solve power system economic dispatch problem. Vergas employed a successive linear programming (SLP) approach for security constrained economic dispatch (SCED) problem. The method employed a new dual affin interior point algorithm for solving LP problems and solved the classical OPF problem with power flow constraints, flows, real and reactive generations, transformer tap ratios and voltage magnitudes. This method was tested on IEEE 30-bus and 118-bus systems. The interior point approach gave the optimal solution in a less number of iterations.

Momoh [2] presented an implementation of a Quadratic Interior point (QIP) method for optimal power flow problem, economic dispatch and VAR planning. This method solves linear or quadratic functions with linear constraints. The method solves the economic dispatch in two process: (1) the Interior Point algorithm obtains the optimal generations and (2) the generation obtained from the IP method are implemented into the load flow to determine the violations. This method was tested on IEEE 14-bus test; however, security constrained economic dispatch or VAR planning objectives was not handled. The CPU time of QIP when compared to MINOS 5.0 was 8:1, and the result obtained were promising.

Mamoh [2] presented an approach that employed Karmakar's interior point method for solving linear programming problems. The method presented is an extended quadratic interior point (EQIP) method based on improvement of initial condition for solving both linear and quadratic programming problems. The method is an extension of the dual affine algorithm and solves power system optimization problem such as economic dispatch and VAR planning problem. The method is capable of accommodating the non-linearity in objectives and constraints. Discrete control variables and contingency constrained problems were not handled in the formulation of this aspect of work. The efficiency of this method is based on the ability to start with a good initial starting point. The EQIP approach was tested on 118-bus system and compared to MINOS 5.0 and it was found to be faster by a factor 5:1

Lu and Unum [2] presented an IP method for solving various sizes of network constrained security control linear programming problems. The method solved to relief the network overloads by active power controls and employed controls such as the generation shifting, phase shifter control HVDC link control, and load shedding. The method employed the linear programming technique to obtain an initial feasible solution before applying the interior point algorithm. The method appears to be efficient in terms of speed and accuracy. This method was applied to the IEEE 6, 30 and 118 bus test cases. The test case results were reliable, and the method uses less CPU time when compared to MINOS 5.0, however convergence may slow in the last few iterations of the process.

Granville [2] presented an IP method for solving the VAR planning objective function of installation cost and losses. The problem solved was a non-convex, non-linear programming problem with non-linear constraints, and the primal-dual variant of interior of interior point was discussed in this paper. This method was tested on very large practical (1862 and 3462) bus systems and the method handles unfeasibility by routinely adjusting the limits to handle load flow limits. However, proper weight must be assign in order to reach a solution satisfactory for both loss minimization and reactive power injection costs.

EVOLUTIONARY PROGRAMMING BASED METHOD

Jason Yuryevich and Kit Po Wong [4] developed an Evolutionary Programming based algorithm for the solution of OPF problem. This method was demonstrated for different classes of cost characteristics (quadratic, piecewise quadratic and sine). The method has been tested on standard IEEE-30 bus system. The basic algorithm is classified as follows: -

1. Representation of population
2. Initialization
3. Fitness of Candidate solution
4. Producing new solution by mutation

It can be concluded that NLP has high accuracy, but poor convergence for large systems. Quasi -Newton methods are inferior in performance to sparse Newton (and other Hessian) methods. They are inefficient on large sparse systems and now completely superceded. SUMT methods are known to exhibits numerical difficulties when the penalty factors they generate become inordinately large. They are now completely superceded. The LP has fast speed and reasonable accuracy and suitable for large systems. Interior point features good starting point and fast convergence. The advantages of the Evolutionary Programming based method are that, it can handle the generating plant with non-convex cost function generating plants, where the other classical method fails.

2.2 LITERATURE RIVEW ON OPEN ACCESS POWER TRANSACTION

Several models have been considered by the power industries for competition towards low cost power under open access transmission. Details of these are given below:

Francis D. Galina and Mariza Ilic [8] presented a general mathematical framework for the analysis and management of the power transaction under open access subjected to system security constraints. The framework introduces the notions of a virtual network of transactions and the transaction matrix, both describing virtual power flows among financial entities. The mathematical framework presented emphasizes on the power transaction between trading financial entities as the basic independent variable under open access. Financial entities can represent individual or groups of generators, retail or groups of loads or pure trading entities. The proposed framework can model utilities, purchasing pools, independent power producers and marketers. A minimum distance algorithm is presented as a means to allocate limited transmission capacity under congested conditions to a set of transactions proposed by the market forces. This algorithm serves to reschedule proposed transactions as well as to trade reserved transaction rights and to allocate transmission losses. This framework and these algorithms could be useful to assist on Independent System Operator (ISO) in its monitoring and emergency duties, both in an operations- planning and a real– time mode.

Roberto W. Ferroro and S.M. Shahidehpour [9] presented a methodology to compute the optimal inter – utility power interchange in deregulated power systems. Each utility defines a price curves for the import and export interchange. The problem is formulated as an optimization approach with a non-linear objective function and linear constraints. Electric losses are considered in the solution using penalty factors. The values of the penalty factors are calculated from a DC power flow. The price offered and required by the utilities are affected by the inclusion of the losses. It has been observed that the utilities with greater contributions to electric losses may have to lower their generation level even when they offer more reasonable prices for their exports. The method implemented on a three-area test system.

Rana Mukerji, et. al. [10], presented an application of OPF for the evaluation of wheeling and non-utility generation (NUG) related options. The method used OPF to determine the best control settings to accommodate wheeling or NUG options so as to maintain system security while minimizing losses or production cost. The above model uses the OPF for calculating the short-term marginal wheeling costs. Case studies involving the Northeast utilities and IEEE-30 bus test system was presented. It has been found that OPF can be used effectively to address a broad range of wheeling or NUG related planning issues.

Les Pereira et. al. [11] presented a case study of connecting a 50 MW combustion plant within a 2500 MW local load area. The study shows an illustration of how a small

50 MW generating plant can make a dramatic improvement in a generation deficient load area of 2500 MW that "imports" 80% of its power, if the plant location is internally within the load area. Furthermore, there is a significant improvement, if its (extra generation) connection is directly into the load area.

Paul R. Gribik, et. al. [12] presented a study on California's congestion management protocols, which provides power exchange and bilateral contract parties. The proposed method is that the ISO's goal is to efficiently auction transmission capacity instead of operating an energy market. This interpretation was more consistent with the observation that the proposed method may allow pairs of trade between scheduling coordinators (SC's). It has suggested that the Independent System Operator (ISO) is to adjust the preferred schedule submitted by the SC's only, if there is transmission congestion. If there is no congestion, ISO accept the preferred schedules submitted by the SC's.

John W.M. Cheng, et. al. [13] presented a method to evaluate an electricity transaction on the basis of system security, especially when numerous transactions have to be processed simultaneously. Monto Carlo simulations are used to construct a large population of random Bilateral Transaction Matrices (BTM) simulating the market activities. The random transaction indices are classified as either "secure" or "insecure" based on load flow studies. Quantitative measures, termed the probabilities of Secure Transactions (POST) are derived from the simulation results to analyze the feasibility of transactions in terms of security. The impact of firm contracts on system security as measured by POST is also studied under different operating and planning scenarios.

R.S. Fang et. al., [14] considered an open transmission dispatch environment in which pool and bilateral/multilateral dispatch coexist and proceeds to develop a congestion management strategy for different scenarios. The method described privatization of electricity transactions and related curtailment strategies and a mechanism for coordination between market participants to achieve additional economic advantages. A five-bus system has been used to demonstrate the proposed method.

Mesut E. Baran, et. al. [15] presented a power flow based method for an accurate assessment of the impact of a transaction on an area/utility. The method determines the following: the flow path of the transaction (both real and reactive power components), generator reactive power support from each area/utility. The assessment method was extensively tested on a real life system. The results indicate that the proposed aggregation method provides very accurate assessment of a transaction's impact on a system whereas marginal approaches, which make use of major components that, do not perform well. The method can also be used for determining transmission capacity reservations, and for addressing congestion problems.

CHAPTER 3

PROBLEM FORMULATION

3.1 OPTIMAL POWER FLOW ALGORITHM USING EVOLUTIONARY PROGRAMMING

The Optimal generation of the generating units satisfying transmission constraints can be solved with the help of Evolutionary Programming (EP). The implementation of EP in the OPF problem has been carried out using the following steps:

1. Prepare the database for the line data, bus data and generator data. Line data includes the information of the lines such as MVA limits, resistances and reactances of lines, Bus data includes the information of generators, loads at each and every bus. The generator data includes the cost coefficient of the generators including real and reactive generation limits.

2. Formation of Y bus using line resistance, reactance, shunt elements and tap changer ratio.

3. Calculate the number of bit required for generators by using the condition-

$$2^{bit\_reqd_i} < (Pg_i^{\max} - Pg_i^{\min}) \times precision < 2^{bit\_reqd_i}$$

Where precision $= 10^{\text{no of decimal place accuracy required}}$

4. Calculate the total no. of bit required for chromosomes generation using the formula

$$Total\ bit\ required = \sum_{\substack{i=1 \\ i \neq slack}}^{Ngen} bit\_reqd_i$$

5. Assume max population and population size

6. Generate chromosomes randomly consisting bits (0 or 1)

Ci = [010........001]

7. Separate out the bits of each chromosome for generations, and convert it into its equivalent decimal values. The values of generation corresponding to the i^{th} generation may be expressed by the relation –

$$Pg_i = Pg_i^{min} + deci(b_1b_2......)_2 \times ((Pg_i^{max} - Pg_i^{min})/(2^{bit\_reqd_i} - 1))$$

Where $deci(b_1b_2......)_2$ represents the decimal value of bits corresponding to i^{th} generation.

8. Assume

$$Pl = 0.03 \times P_D$$

Where

$$P_D = \sum_{i=1}^{Nbus} Pd_i$$

9. The generation of slack bus generator has been calculated using the following equality constraint-

$$Pg_{slack} = -\sum_{\substack{i=1 \\ i \neq slack}}^{Ngen} Pg_i + P_D + Pl$$

10. Check the Pg_{slack}, it should be within Pg_{slack}^{min}. & $Pg_{slack}^{max.}$ Otherwise go to step 6.

11. Perform the loadflow using the Newton Raphson method and hence determine bus voltage magnitudes and phase angles.

12. Calculate the line flows and line losses and also find the total line losses in the network. If the difference between calculated losses and assumed losses (or losses of previous iteration) violated the tolerable limit, go to step 9., otherwise go to step 13.

13. Check the bus voltage violation, i.e.

$$V_i^{min} \le V_i \le V_i^{max} \qquad ; i = 1, N_{bus} \text{ except generator bus}$$

If there is any bus that violated the limit then provided reactive power support optimally or go to step 6.

14. Check the MVA flows violation, i.e.

$$MVA_{ij}^{min} \le MVA_{ij} \le MVA_{ij}^{max} \qquad ; \text{for all the lines connected between bus i and j}$$

If the limit violates then provide reactive power support optimally or go to step 6.

15. Calculate the cost of generation using the relation

$$Cost = \sum_{i=1}^{Ngen} (a_i \times Pg_i^2 + b_i \times Pg_i + c_i)$$

(The cost function my be of any other nature)

16. Check the cost for k^{th} chromosomes with $(k\text{-}1)^{th}$ chromosomes, store the optimal cost in these two alongwith the corresponding generation.

17. Check the no. of chromosomes generated, if it is less then pop_size (say 20) go to step 6 for next random generation of chromosome.

18. Check the total no. of population, if it equal to maximum population then go to step 26.

19. The probability of selection will be higher for the chromosomes which has low value of cost, therefore fitness of each chromosomes will be reciprocal of cost.

$fitness_i = 1/Cost_i \qquad ;i = 1, \text{pop\_size}$

Where $fitness_i$ = fitness value of function for i^{th} chromosome

20. Total fitness value

$$F = \sum_{i=1}^{pop\_size} fitness_i$$

21. The probability of selection can be calculated by using the formula

$p_i = fitness_i / F \qquad ;i = 1, \text{pop\_size}$

22. Cumulative probabilities for each chromosomes are: -

$q_i = \sum_{j=1}^{i} p_j \qquad ;i = 1, \text{pop\_size}$

23. Generate the pop_size (say 20) random number in the range [0, 1]. Let the number represented by

Random number = r_i ; i = 1, pop_size

Select for each r_i the just smaller and just greater value of q_i

$q^m \le r_i \le q^M \qquad ;i = 1, \text{pop\_size}$

Then i^{th} chromosomes will be selected for new population.

Assume the probability of crossover p_c (say 0.9) therefore $p_c \times$ pop_size chromosomes undergo crossover.

Generate random number between [0, 1],

If $r_i < p_c$; i = 1, pop_size

CHAPTER 4

RESULTS AND DISCUSSSION

The analysis and management of open access performed on the IEEE-30 bus test system is shown in fig. 4.1. The parameters and data of IEEE-30 bus test system are given in appendix A1.1, A1.2, A1.3, A1.4.

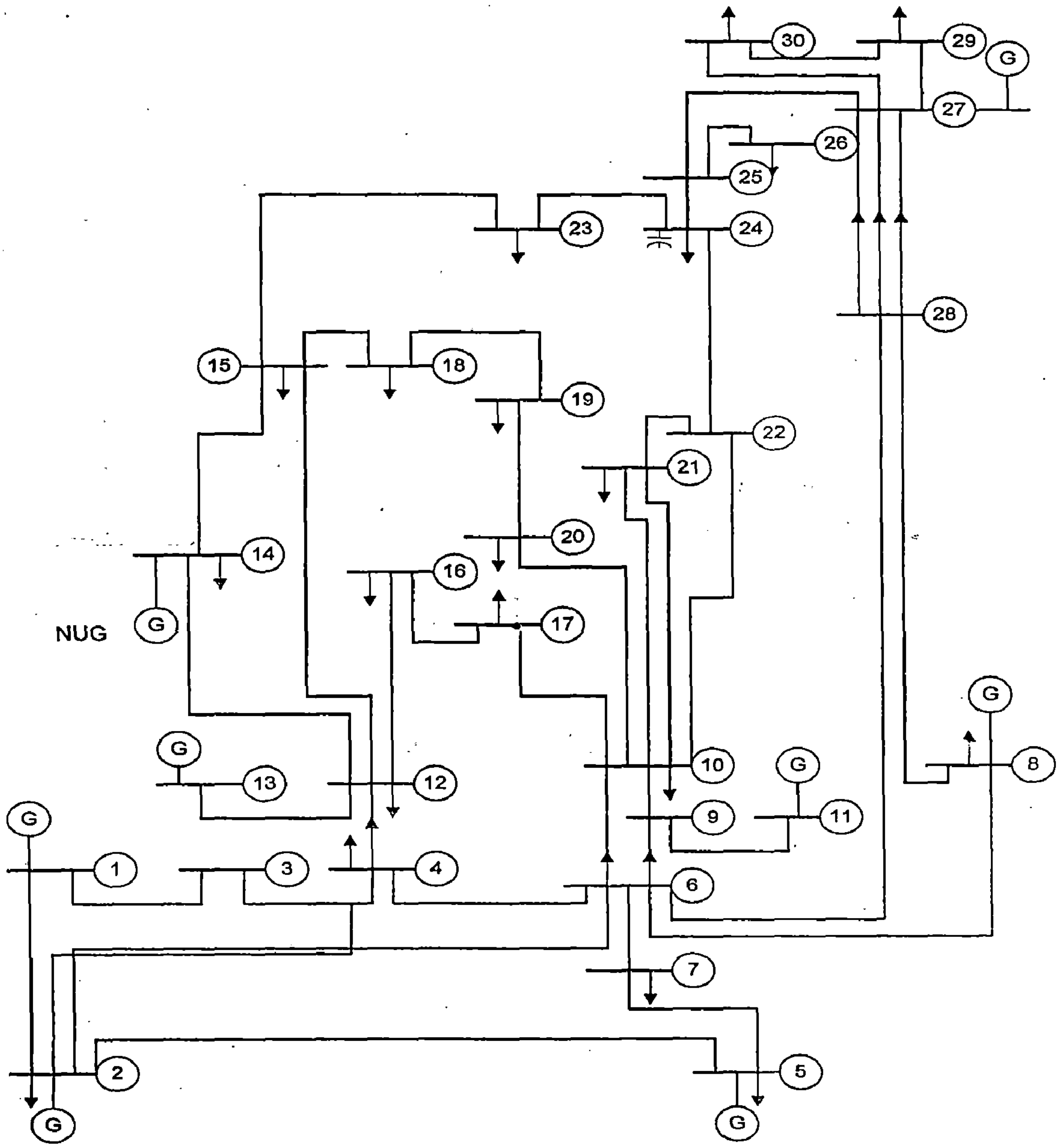

Fig. 4.1 IEEE-30 bus test system

The evolutionary programming based OPF (EP-OPF) has been used for the analysis and management of open access power transactions. The analysis performed for different case studies.

CASE 1: Optimal generation and total cost of IEEE-30 bus system has been calculated using EP-OPF. The results are shown in Table 4.1

TABLE 4.1 Optimal generation of 6-generating plant

Bus No.	Voltage (p.u.)	Generation (MW)
1	1.05	175.168
2	1.036	48.035
5	1.005	23.014
8	1.016	23.039
11	1.069	11.020
13	1.055	12.767

Total cost of generation C = 803.732 £/hr

CASE 2: The shift in optimal generations and hence the total costs have been observed for a unit MW increase in load at all the buses. Table 4.2 gives the generation, total cost and the increase in cost to meet the unit MW load.

CASE 3: The new generator has a maximum capacity of 50 MW, but the question is whether it can sell full 50 MW to the buyer through the system or not. The limitation is due to the line limit violation. To perform the case 3 a generator of very high cost characteristics is connected at the bus through which the power can be sold or transferred.

TABLE 4.2 Optimal generation of 6-generators for increment of load by 1 MW*

Bus No.	Pg1 MW	Pg2 MW	Pg5 MW	Pg8 MW	Pg11 MW	Pg13 MW	Total Cost of gen. (£/hr)	Increase in cost (£/hr)
1	177.39	47.273	20.685	22.745	12.745	13.315	806.977	3.245
2	173.55	50.205	21.575	21.471	12.902	14.301	807.219	3.487
3	173.57	50.205	21.575	21.471	12.902	14.301	807.274	3.542
4	173:59	50.205	21.575	21.471	12.902	14.301	807.446	3.714
5	173.62	50.205	21.575	21.471	12.902	14.301	807.418	3.686
6	171.01	49.326	22.329	26.373	12.510	12.219	807.418	3.686
7	171.02	49.326	22.329	26.373	12.510	12.219	807.474	3.742
8	179.68	46.452	22.671	22.941	10.314	12.274	807.432	3.700
9	171.01	49.326	22.329	26.373	12.510	12.219	807.420	3.688
10	171.01	49.326	22.329	26.373	12.510	12.219	807.422	3.690
11	171.01	49.326	22.329	26.373	12.510	12.219	807.420	3.688
12	173.58	50.205	21.575	21.471	12.902	14.301	807.321	3.589
13	173.58	50.205	21.575	21.471	12.902	14.301	807.321	3.589
14	173.60	50.205	21.575	21.471	12.902	14.301	807.388	3.656
15	173.61	50.205	21.575	21.471	12.902	14.301	807.408	3.676
16	171.01	49.326	22.329	26.373	12.510	12.219	807.420	3.688
17	171.01	49.326	22.329	26.373	12.510	12.219	807.437	3.705
18	171.03	49.326	22.329	26.373	12.510	12.219	807.503	3.771
19	171.04	49.326	22.329	26.373	12.510	12.219	807.514	3.782
20	171.03	49.326	22.329	26.373	12.510	12.219	807.497	3.765
21	171.02	49.326	22.329	26.373	12.510	12.219	807.462	3.730
22	171.02	49.326	22.329	26.373	12.510	12.219	807.400	3.668
23	171.03	49.326	22.329	26.373	12.510	12.219	807.500	3.768
24	171.04	49.326	22.329	26.373	12.510	12.219	807.515	3.783
25	179.70	46.452	22.671	22.941	10.314	12.274	807.500	3.768
26	171.06	49.326	22.329	26.373	12.510	12.219	807.586	3.854
27	179.69	46.452	22.671	22.941	10.314	12.274	807.450	3.718
28	171.01	49.326	22.329	26.373	12.510	12.219	807.436	3.704
29	179.43	46.393	23.151	20.588	12.510	12.274	807.585	3.853
30	176.71	49.443	21.027	21.373	13.059	12.712	807.398	3.666

*The above result gives an idea about the difference in extra generation cost due to different loading locations.

Based on the power supplied by the costly unit, the power transfer capability of the NUG is known. The Table 4.3 presents the maximum possible generation that can be transferred to different buses.

TABLE 4.3 Maximum possible generation by NUG without line violation**

Bus No.	Maximum Load which can be supplied by Pg14 (MW)
1	19.327
2	22.315
3	28.705
4	28.379
5	26.535
6	28.006
7	27.440
8	25.345
9	24.173
10	24.070
11	26.035
12	30.092
13	36.375
14	50.000
15	20.157
16	26.908
17	26.614
18	14.090
19	17.025
20	21.135
21	25.734
22	25.734
23	15.362
24	21.429
25	15.705
26	11.057
27	26.908
28	22.896
29	12.427
30	13.601

**The above analysis gives information about the maximum generation allowed to the NUG.

CASE 4: The increment in load (1 MW) may be supplied by the NUG, which is installed at bus no. 14 (it may be installed at any other bus). The NUG treated as a private supplier, which is ready to supply the power to utility. If the NUG do not want to change the optimal point of utility, then the generation required to meet the demand is shown in Table 4.4.

TABLE 4.4 Generation of NUG to meet the extra loads (1 MW)***

Bus No.	Pg14 MW	Cost of generation (£/hr)
1	0.919	3.278
2	0.956	3.413
3	0.972	3.472
4	0.987	3.525
5	1.020	3.644
6	0.998	3.566
7	1.014	3.623
8	1.001	3.577
9	0.999	3.568
10	0.999	3.568
11	0.998	3.567
12	0.985	3.518
13	0.985	3.516
14	1.000	3.574
15	1.009	3.604
16	0.998	3.567
17	1.003	3.584
18	1.021	3.651
19	1.024	3.661
20	1.019	3.642
21	1.011	3.612
22	1.010	3.610
23	1.020	3.646
24	1.025	3.664
25	1.019	3.644
26	1.045	3.737
27	1.006	3.593
28	1.004	3.587
29	1.042	3.724
30	1.066	3.813

***This result shows that at bus no. 1,2,3,4,6,9,10,11,12,13,16 the power required to meet the increased load is less than 1 MW, this is due to reduction in losses of the network. Therefore at these points the NUG will be interested to supply power.

CASE 5: If the utility is interested to make an agreement with the NUG in such a way, so the NUG has to supply increased load demand whatever it is. In this case the NUG supplying constant 1 MW while the utility have to adjust the reaming load. The optimal generation calculated is shown in Table 4.5

CASE 6: If the utility makes an agreement with the NUG and all seven are supplying the increased load demand. The result of EP-OPF are shown in Table 4.6

The optimal generation cost for existing load is 803.732 £/hr. For the extra load the optimal generation cost charges, which varies according to the load point variation. The maximum allowable generation by the NUG also varies accordingly to the load point. The incremental load at the bus is 1 MW, which is less than the maximum allowable loading. Therefore for the management of open access power transaction all the condition will be considerable.

The transaction options for different buses are shown in Table 4.7

TABLE 4.5 Generation of 6-existing Generator (NUG supplies 1 MW to meet the increased load)**

Bus No.	Pg1 MW	Pg2 MW	Pg5 MW	Pg11 MW	Pg13 MW	Pg13 MW	Total cost of gen. (£/hr	Increase in Cost (£/hr)
1	180.961	46.276	20.546	20.588	12.588	12.438	803.473	-0.259
2	180.873	46.276	20.616	20.588	11.882	13.205	803.667	-0.065
3	180.890	46.276	20.616	20.588	11.882	13.205	803.724	-0.008
4	180.906	46.276	20.616	20.588	11.882	13.205	803.776	0.044
5	180.943	46.276	20.616	20.588	11.882	13.205	803.899	0.167
6	180.919	46.276	20.616	20.588	11.882	13.205	803.820	0.088
7	180.935	46.276	20.616	20.588	11.882	13.205	803.875	0.143
8	180.921	46.276	20.616	20.588	11.882	13.205	803.828	0.096
9	180.920	46.276	20.616	20.588	11.882	13.205	803.823	0.091
10	180.920	46.276	20.616	20.588	11.882	13.205	803.823	0.091
11	180.920	46.276	20.616	20.588	11.882	13.205	803.823	0.091
12	180.904	46.276	20.616	20.588	11.882	13.205	803.770	0.038
13	180.904	46.276	20.616	20.588	11.882	13.205	803.770	0.038
14	180.920	46.276	20.616	20.588	11.882	13.205	803.824	0.092
15	180.921	46.276	20.616	20.588	11.882	13.205	803.855	0.123
16	180.918	46.276	20.616	20.588	11.882	13.205	803.818	0.086
17	180.924	46.276	20.616	20.588	11.882	13.205	803.837	0.105
18	180.944	46.276	20.616	20.588	11.882	13.205	803.903	0.171
19	180.948	46.276	20.616	20.588	11.882	13.205	803.917	0.185
20	180.842	46.276	20.616	20.588	11.882	13.205	803.897	0.165
21	180.933	46.276	20.616	20.588	11.882	13.205	803.867	0.135
22	180.932	46.276	20.616	20.588	11.882	13.205	803.865	0.133
23	180.942	46.276	20.616	20.588	11.882	13.205	803.898	0.166
24	180.948	46.276	20.616	20.588	11.882	13.205	803.918	0.186
25	180.942	46.276	20.616	20.588	11.882	13.205	803.899	0.167
26	180.970	46.276	20.616	20.588	11.882	13.205	803.992	0.260
27	180.928	46.276	20.616	20.588	11.882	13.205	803.850	0.118
28	180.925	46.276	20.616	20.588	11.882	13.205	803.840	0.108
29	180.996	46.276	20.616	20.588	11.882	13.205	803.980	0.248
30	180.993	46.276	20.616	20.588	11.882	13.205	804.069	0.337

****The result shows negative value of incremental cost, this is due to reduce in losses of network only. This type of reduction is allowed at the bus where the cost is negative. In this case the existing utility will have profit.

TABLE 4.6 Optimal generation of &-generator to meet the increased load demand (1 MW)***

Bus No.	Pg1 MW	Pg2 MW	Pg5 MW	Pg8 MW	Pg11 MW	Pg13 MW	Pg14 MW	Total Cost (£/hr)	Increase in Cost (£/hr)
1	171.57	47.33	20.41	26.47	15.02	12.61	0.783	807.351	3.619
2	179.85	49.85	21.64	17.74	12.03	12.76	0.587	807.170	3.438
3	177.71	48.09	20.06	20.68	14.70	12.21	0.783	807.258	3.526
4	179.49	45.16	20.20	22.05	13.05	13.26	1.076	807.482	3.750
5	179.92	49.85	21.64	17.74	12.03	12.76	0.587	807.404	3.672
6	179.51	45.16	21.20	22.05	13.05	13.26	1.076	807.526	3.794
7	179.12	49.85	21.64	17.74	12.03	12.21	1.859	807.409	3.677
8	179.51	45.16	21.20	22.05	13.05	13.26	1.076	807.533	3.801
9	179.51	45.16	20.20	22.05	13.05	13.26	1.076	807.529	3.797
10	179.51	45.16	20.20	22.05	13.05	13.26	1.076	807.529	3.797
11	179.51	45.16	20.20	22.05	13.05	13.26	1.076	807.526	3.794
12	179.49	45.16	20.20	22.05	13.05	13.26	1.076	807.476	3.744
13	179.49	45.16	20.20	22.05	13.05	13.26	1.076	807.476	3.744
14	175.41	48.21	20.54	20.09	12.58	14.84	2.446	807.654	3.922
15	175.08	51.32	21.30	20.00	13.92	12.00	0.587	807.293	3.561
16	179.51	45.16	20.20	22.05	13.05	13.26	1.076	807.526	3.794
17	179.51	45.16	20.20	22.05	13.05	13.26	1.076	807.543	3.811
18	179.53	45.16	20.20	22.05	13.05	13.26	1.076	807.610	3.878
19	179.53	45.16	20.20	22.05	13.05	13.26	1.076	807.620	3.888
20	179.53	45.16	20.20	22.05	13.05	13.26	1.076	807.600	3.868
21	179.52	45.16	20.20	22.05	13.05	13.26	1.076	807.572	3.840
22	179.52	45.16	20.20	22.05	13.05	13.26	1.076	807.570	3.838
23	179.53	45.16	20.20	22.05	13.05	13.26	1.076	807.604	3.872
24	179.54	45.16	20.20	22.05	13.05	13.26	1.076	807.623	3.891
25	179.53	45.16	20.20	22.05	13.05	13.26	1.076	807.602	3.870
26	179.56	45.16	20.20	22.05	13.05	13.26	1.076	807.695	3.963
27	179.52	45.16	20.20	22.05	13.05	13.26	1.076	807.554	3.822
28	179.51	45.16	20.20	22.05	13.05	13.26	1.076	807.545	3.813
29	179.55	45.16	20.20	22.05	13.05	13.26	1.076	807.686	3.954
30	179.18	49.85	21.64	17.74	12.03	12.21	1.859	807.601	3.869

*****At some buses if the load increases by 1 MW then the total cost shown in Table 4.6 (6-generator providing the optimal power) is less than the total cost shown in Table 4.2 (6-generator providing optimal power). This means that the utility will be in position to supply the power along with the NUG for load increment at that bus.

TABLE 4.7 Transaction options for the corresponding to bus

Bus No.	Transaction options				Transaction options selected
	C′-C (£/hr)	C″ (£/hr)	C‴-C (£/hr)	C′ᵛ-C (£/hr)	
1	3.245	3.278	-0.259	3.619	1&3
2	3.487	3.413	-0.065	3.438	2&3
3	3.542	3.472	-0.008	3.526	2&3
4	3.714	3.525	0.044	3.750	2
5	3.686	3.644	0.167	3.672	2
6	3.686	3.566	0.088	3.794	2
7	3.742	3.623	0.143	3.677	2
8	3.700	3.577	0.096	3.801	2
9	3.688	3.568	0.091	3.797	2
10	3.690	3.568	0.091	3.797	2
11	3.688	3.567	0.091	3.794	2
12	3.589	3.518	0.038	3.744	2
13	3.589	3.516	0.038	3.744	2
14	3.656	3.574	0.092	3.922	2
15	3.676	3.604	0.123	3.561	4
16	3.688	3.567	0.086	3.794	2
17	3.705	3.584	0.105	3.811	2
18	3.771	3.651	0.171	3.878	2
19	3.782	3.661	0.185	3.888	2
20	3.765	3.642	0.165	3.868	2
21	3.730	3.612	0.135	3.840	2
22	3.668	3.610	0.133	3.838	2
23	3.768	3.646	0.166	3.872	2
24	3.783	3.664	0.186	3.891	2
25	3.768	3.644	0.167	3.870	2
26	3.854	3.737	0.260	3.963	2
27	3.718	3.593	0.118	3.822	2
28	3.704	3.587	0.108	3.813	2
29	3.853	3.724	0.248	3.954	2
30	3.666	3.813	0.337	3.869	1

The details of transaction options is given in step 9 of algorithm of open access power transaction (sec. 3.2)

CHAPTER 5

CONCLUSIONS AND SCOPE OF FUTURE WORK

CONCLUSION: In the present era it is very important to economize the generation cost satisfying operational constraint. Optimal power flow (OPF) is a very important tool to solve this problem. There are many algorithms available and they are capable of solving the OPF problem. But these algorithms have limitation over the cost characteristics. The proposed EP-OPF can handle the different cost curves for different plants and even non-convex characteristics also.

In the view of increased load demand it is necessary to invite the private power producers. To handle the condition with new load demand and induction of private parties, it is important to analyze the economic operation with constraints. There are many ways to provide the power with or without private parties. The feasible option for the transaction is found based on the minimum cost of generation without violating constraints.

SCOPE FOR FUTURE WORK: The algorithm does not consider the contingency condition, so it may be included in the algorithm. Incorporating the FACTs devices maximum power transfer limit can be increased. The optimization study with FACTs can be studied to analyze the best possible transaction options.

REFERENCES

[1] James A. Momoh, M.E. El-Hawary and Rambabu Adapa, "A Review of selected Optimal Power Flow Literature to 1993, Part I: Non Linear & Quadratic Programming Approaches," IEEE Transactions on Power Systems, Vol. 14, No. 1, February 1999, pp 96-104.

[2] James A. Momoh, M.E. El-Hawary and Rambabu Adapa, "A Review of selected Optimal Power Flow Literature to 1993, Part II: Newton, Linear Programming & Interior Point Methods," IEEE Transactions on Power Systems, Vol. 14, No. 1, February 1999, pp 105-111.

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

[4] Jason Yuryevich, and Kit Po Wong, "Evolutionary Programming Based Optimal Power Flow Algorithm," IEEE Transaction on Power Systems, Vol. 14, No. 4, November 1999, pp 1245-1250.

[5] Goldberg, D.E., "Genetic Algorithm in Search, Optimization, and Machine Learning," International Student Edition, Addision-Weseley, 1999

[6] Zbigniew Michalewich, "Genetic Algorithm + Data Structures = Evolutionary Programs," Springer-Verlag, 1992.

[7] Allen J. Wood and Druce F. Wollenberg, "Power Operation and Control," Jhon Wiley and Sons Inc., 1996.

[8] Francis D. Galina and Mariza Ilic, "A mathematical Framework for the Analysis and Management of Power Transaction under Open Access," IEEE Transactions on Power Systems, Vol. 13, No. 2, May 1998, pp 681-687.

[9] Roberto W. Ferro and S.M. Shahidehpour, "Optimal Power Transactions with Network Constraints in Deregulated Systems," Electric Machine and Power Systems, 1997, pp 661-675.

[10] Rana Mukerji, Wendell Neugebauer, Richard P. Ludorf and Armand Catelli, "Evolution of Wheeling and Non-Utility Generation (NUG) Options using Optimal Power Flows," IEEE Transactions on Power Systems, Vol. 7, No.1, February 1992,pp 201-207.

[11] Les Pereira and Don DeBerry, "The Role of Less-Economic Generating Units in the Open Access Transmission Era," IEEE Transactions on Power Systems, Vol. 14, No. 3, August 1999, pp 1133-1136.

[12] Paul R. Gribik, George A. Angelidis and Ross R. Kovacs, "Transmission Access and Pricing with Multiple Separate Energy Forwards Markets," IEEE Transactions on Power Systems, Vol. 14, No. 3, August 1999, pp 865-876.

[13] John W.M. Cheng, Donald T. McGillis and Francisco D. Galina, "Probabilistic Security Analysis of Bilateral Transactions in a Deregulated Environment," IEEE Transactions on Power Systems, Vol. 14, No. 3, August 1999, pp 1153-1159.

[14] R.S. Fang and A.K. David, "Transmission Congestion Management in an Electricity Market," IEEE Transactions on Power Systems, Vol. 14, No. 3, August 1999, pp 877-883.

[15] Mesut E. Baron, Venket Banunarayanan and Kenneth E. Garren, "A transaction Assessment Method for Allocation of Transmission Services," IEEE Transactions on Power Systems, Vol. 14, No. 3, August 1999, pp 920-928.

[16] D.L. Post, S.S. Coppinger, and G.B. Sheble, "Application of Auctions as a pricing Mechanism for the interchange of Electrical Power," IEEE Transaction on Power Systems, Vol. 10, No. 3, August 1995, pp 1580-1584.

[17] Einar Stale Huse, Ivar Wangensteen and Hans H. Fannes, "Thermal Power Generation Scheduling by Simulated Competition," IEEE Transactions on Power Systems, Vol. 14, No. 2, May 1999, pp 472-477.

[18] Fernando Alvarado, "The Stability of Power System Markets," IEEE Transactions on Power Systems, Vol. 14, No. 3, May 1999, pp 505-511.

APPENDIX

A1.1 GENERATOR DATA

Bus No.	Pg^{min} MW	Pg^{max} MW	Qg^{min} MVAr	Qg^{max} MVAr	Cost Coefficients		
					a (£/hr)	b (£/MWhr)	c (£/MW^{2}hr)
1	50	200	-20	250	0.0	2.0	0.00375
2	20	80	-20	100	0.0	1.75	0.0175
5	15	50	-15	80	0.0	1.0	0.0625
8	10	35	-15	60	0.0	3.25	0.00834
11	10	30	-10	50	0.0	3.0	0.025
13	12	40	-15	60	0.0	3.0	0.025
14	0	50	-20	80	0.0	3.50	0.0725

Generating cost $f_i = a_i + b_i Pg_i + c_i Pg_i^2$

A1.2 BRANCH DATA

Branch No.	Bus No.	R p.u.	X p.u.	B (total) p.u	Rating MVA
1	1-2	0.0192	0.0575	0.0264	130
2	1-3	0.0452	0.1852	0.0204	130
3	2-4	0.0570	0.1737	0.0184	65
4	3-4	0.0132	0.0379	0.0042	130
5	2-5	0.0472	0.1983	0.0209	130
6	2-6	0.0581	0.1763	0.0187	65
7	4-6	0.0119	0.0414	0.0045	90
8	5-7	0.0460	0.1160	0.0102	70
9	6-7	0.0267	0.0820	0.0085	130
10	6-8	0.0120	0.0420	0.0045	32
11	6-9	0.0	0.2080	0.0	65
12	6-10	0.0	0.5560	0.0	32
13	9-11	0.0	0.2080	0.0	65
14	9-10	0.0	0.1100	0.0	65
15	4-12	0.0	0.2560	0.0	65
16	12-13	0.0	0.1400	0.0	65
17	12-14	0.1231	0.2559	0.0	32
18	12-15	0.0662	0.1304	0.0	32
19	12-16	0.0945	0.1987	0.0	32
20	14-15	0.2210	0.1997	0.0	16
21	16-17	0.0824	0.1932	0.0	16
22	15-18	0.1070	0.2185	0.0	16
23	18-19	0.0639	0.1292	0.0	16
24	19-20	0.0340	0.0680	0.0	32
25	10-20	0.0936	0.2090	0.0	32
26	10-17	0.0324	0.0845	0.0	32
27	10-21	0.0348	0.0749	0.0	32
28	10-22	0.0727	0.1499	0.0	32
29	21-22	0.0116	0.0236	0.0	32
30	15-23	0.1000	0.2020	0.0	16
31	22-24	0.1150	0.1790	0.0	16
32	23-24	0.1320	0.2700	0.0	16
33	24-25	0.1885	0.3292	0.0	16
34	25-26	0.2544	0.3800	0.0	16
35	25-27	0.1093	0.2087	0.0	16
36	28-27	0.0	0.3960	0.0	65
37	27-29	0.2198	0.4153	0.0	16
38	27-30	0.3202	0.6027	0.0	16
39	29-30	0.2399	0.4533	0.0	16
40	8-28	0.0636	0.2000	0.0214	32
41	6-28	0.0169	0.0599	0.0065	32
42	10-10	0.0	-5.2600		
43	24-24	0.0	-25.0000		

Base MVA = 100

A1.3 LOAD DATA

Bus No.	Load	
	MW	MVAR
1	0.0	0.0
2	21.7	12.7
3	2.4	1.2
4	7.6	1.6
5	94.2	19.0
6	0.0	0.0
7	22.8	10.9
8	30.0	30.0
9	0.0	0.0
10	5.8	2.0
11	0.0	0.0
12	11.2	7.5
13	0.0	0.0
14	6.2	1.6
15	8.2	2.5
16	3.5	1.8
17	9.0	5.8
18	3.2	0.9
19	9.5	3.4
20	2.2	0.7
21	17.5	11.2
22	0.0	0.0
23	3.2	1.6
24	8.7	6.7
25	0.0	0.0
26	3.5	2.3
27	0.0	0.0
28	0.0	0.0
29	2.4	0.9
30	10.6	1.9

Total load = 283.4 MW, 126.2 MVAR

A1.4 TRANSFORMER DATA

Bus	% tap
From to	
6-9	1.020
6-10	0.900
4-12	0.950
28-27	0.940

A2 CODING OF EP-OPF FORMULATION

```
#include<stdio.h>
#include<math.h>
#include<stdlib.h>
#include<time.h>
#include"inverse.c"
#define pop_size 20
int n,ncap,n1,ln,true=1,false=0,ind[44],line[44][3],typ[3],gen_conn[32];
int i,ii,index,j,jj,k,kk,rl,l,ll,maxit,nj,nm1,npu,npq,ik,m,again,tolerance;
float matx[32][32],b[32][32],g[32][32],jacob[60][60],jacobinv[60][60];
float pc[32],pg[32],qc[32],qg[32],ua[32],um[32],uad[32],da[32];
float pe[32],qe[32],gij,bij,sn,cs,qpmin[32],qpmax[32],yp[5],vsp[32];
float du[60],dp[32],dq[60],mvajk[44],mvakj[44],MVAM[44],z1[44][5];
float angd,angi,angj,bjj,bkk,bjk,bkj,conv,csd,epsp,epsq,gjj,gkk;
float gjk,gkj,ppg,ppq,ppc,pjk,pkj,ploss,qqg,qqc,qjk,qkj,qloss,r,rsq,sq,st;
float ssd,tap,uij,uua,uum,ui,uui,uj,usqk,usqj,uskj,usc,ucs,x,yc,xsq,err;
float qpmn,qpmx,vtolerance,vspv,linePlosses,lineQlosses;
void main()
{
FILE*f;
int tt,tt1,p,kk1,ge,bit[25][100],temp,temp3,temp2[100],bus[44],count[25];
int ngen,total_bit_reqd,bit_start,bit_stop,bit_reqd[10],slack,ra3,ra4,N;
float ttt,ra1[25],ra2,OPT,probc,probm,OPTPG[100],OPTCO;
float co[21],optco[100],fitness,prob[25],q[25],optPg[100][10];
float mvajk[44],mvakj[44],MVAM[44],Pload[44],Qload[44];
float Pgmin[10],Pgmax[10],Qgmin[10],Qgmax[10],A[10],B[10],C[10];
float Pd,Qd,Pg[25][10],Pl,ppjk[44],qqjk[44],ppkj[44],qqkj[44],temp1;
clrscr();
f1=fopen("loadflow.res","w");
printf("GAME NO. ==");
scanf("%d",&N);
for(i=1;i<=N;i++) ra3=random(100);
f=fopen("gen30.dat","r");
fscanf(f,"%d",&ngen);
for(i=1;i<=ngen;i++)
fscanf(f,"%d%f%f%f%f%f%f%f",&bus[i],&Pgmin[i],&Pgmax[i],&Qgmin[i],&Qgmax[i],&A[i],&B[i],&C
[i]);
fclose(f);
f=fopen("load30.dat","r");
fscanf(f,"%d",&n);
for(i=1;i<=n;i++)fscanf(f,"%f%f",&Pload[i],&Qload[i]);
fclose(f);
f=fopen("mva30.dat","r");
fscanf(f,"%d",&n1);
for(i=1;i<=n1;i++)fscanf(f,"%f",&MVAM[i]);
fclose(f);
Pd=0;Qd=0;
for(i=1;i<=n;i++){
Pd+=Pload[i];
Qd+=Qload[i];
}
slack=1;
/*total_bit_reqd*/
total_bit_reqd=0;
```

```
for(i=1;i<=ngen;i++){
        if(i==slack)bit_reqd[i]=0;
        else{
        bit_reqd[i]=(int)(1+log(1000*(Pgmax[i]-Pgmin[i]))/log(2.0));
        total_bit_reqd+=bit_reqd[i];
        }
}
OPTCO=9999999;
/*random number generation*/
for(ge=1;ge<=pop_size;ge++)
        for(j=1;j<=total_bit_reqd;j++)
        bit[ge][j]=random(2);
/*decimal conversion*/
/*seperation of bits*/
Pl=0.03*Pd;
optco[0]=9999999;
for(tt1=1;tt1<=50;tt1++)/*super loop start*/{
fitness=0.0;
optco[tt1]=9999999;
for(ge=1;ge<=pop_size;ge++){  /*ge loop start*/
top:
bit_start=1;
bit_stop=bit_reqd[1];
for(i=1;i<=ngen;i++){
k=1;Pg[ge][i]=0.0;
        for(j=bit_start;j<=bit_stop;j++){
        if(bit_reqd[i]==0)break;
        Pg[ge][i]+=bit[ge][j]*pow(2,bit_reqd[i]-k);
        k+=1;
        }
if(bit_reqd[i]==0)Pg[ge][i]=0.0;
else Pg[ge][i]=Pgmin[i]+Pg[ge][i]*((Pgmax[i]-Pgmin[i])/(pow(2,bit_reqd[i])-1));
bit_start+=bit_reqd[i];
bit_stop+=bit_reqd[i+1];
}
for(p=1;p<=10;p++)/*loss loop start*/{
Pg[ge][slack]=Pd+Pl;
        for(i=1;i<=ngen;i++){
        if(i==slack)Pg[ge][slack]-=0.0;
        else Pg[ge][slack]-=Pg[ge][i];
        }
if((Pg[ge][slack]<Pgmin[slack])||(Pg[ge][slack]>Pgmax[slack])){
        for(j=1;j<=total_bit_reqd;j++)
        bit[ge][j]=random(2);
goto top;
}
/*cost calculation*/
co[ge]=0;
for(i=1;i<=ngen;i++)
co[ge]+=A[i]+B[i]*Pg[ge][i]*100+C[i]*Pg[ge][i]*Pg[ge][i]*10000;
if(co[ge]>10+optco[tt1-1]){
for(j=1;j<=total_bit_reqd;j++)
bit[ge][j]=random(2);
goto top;
}
/*LOADFLOW PROGRAM*/
```

```
/*************************************************************
f=fopen("ieee30.dat","r");
fscanf(f,"%d%d%d",&n,&n1,&ncap);
for(i=1;i<=n;i++){
        qpmin[i]=0.0;
        qpmax[i]=0.0;
        vsp[i]=0.0;
        }
for(i=1;i<=2*n;i++)
for(j=1;j<=2*n;j++){
jacob[i][j]=0.0;jacobinv[i][j]=0.0;
}
conv=3.14159/180.0;
fscanf(f,"%d%f%f",&maxit,&epsp,&epsq);
npu=0;npq=0;k=1;
for(j=1;j<=n;j++){
        fscanf(f,"%d%d%f%f%f%f%f%f%d",&i,&typ[j],&uum,&uua,&ppg,&qqg,&ppc,&qqc,&gen_con
n[j]);
if(gen_conn[j]==1){
ppg=Pg[ge][k];
k+=1;
}
        switch(typ[j]){
        case 0:{
                index=n;
                break;
                }
        case 1:{
                npu=npu+1;
                index=n-npu;
                break;
                }
        case 2:{
                npq=npq+1;
                index=npq;
                break;
                }
        }
ind[i]=index;
pg[index]=ppg; qg[index]=qqg;
pc[index]=ppc; qc[index]=qqc;
um[index]=uum; ua[index]=uua*conv;
}
/*BEFORE READING THE ADMITTANCE MAKE B AND G MATRICES ALL ZERO*/
for(i=1;i<=n;i++)
for(j=1;j<=n;j++){
g[i][j]=0.0;
b[i][j]=0.0;
}
/*DIMENSION OF JACOBIAN IS npq+n-1 */
nm1=n-1;
nj=nm1+npq;
for(i=1;i<=n1;i++){
        fscanf(f,"%d%d%d%f%f%f%f",&ln,&rl,&ll,&r,&x,&yc,&tap);
        sq=(r*r+x*x)*tap;
        line[i][1]=ll;
```

```
            line[i][2]=rl;
            j=ind[ll];
            k=ind[rl];
st=sq*tap;
xsq=x/sq;
rsq=r/sq;
z1[i][1]=r/st;
z1[i][2]=yc-x/st;
z1[i][3]=rsq*tap;
z1[i][4]=yc-xsq*tap;
b[j][k]=b[j][k]+xsq;
b[k][j]=b[k][j]+xsq;
b[j][j]=b[j][j]-(xsq/tap)+yc;
b[k][k]=b[k][k]-(xsq*tap)+yc;
g[j][k]=g[j][k]-rsq;
g[k][j]=g[k][j]-rsq;
g[j][j]=g[j][j]+rsq/tap;
g[k][k]=g[k][k]+rsq*tap;
}
if(ncap!=0){
            for(i=1;i<=ncap;i++){
            fscanf(f,"%d%f",&j,&yp[i]);
            k=ind[j];
            b[k][k]=b[k][k]+yp[i];
            }
}
fclose(f);
/*******NEWTON RAPHSON METHOD***********/
ik=0;again=true;
while((ik<maxit)&&(again))
{
            fprintf(f1,"executing iteration number=%d\n",ik+1);
            /*compute power as a function of voltage*/
            for(i=1;i<=n;i++){
            pe[i]=0.0;qe[i]=0.0;
                        for(j=1;j<=n;j++){
                        gij=g[i][j]; bij=b[i][j];
                        if((gij!=0.0)||(bij!=0.0)){
                        angd=ua[i]-ua[j]; uij=um[i]*um[j];
                        cs=cos(angd);
                        sn=sin(angd);
                        pe[i]=pe[i]+uij*(gij*cs+bij*sn);
                        qe[i]=qe[i]+uij*(gij*sn-bij*cs);
                        }
                        }
            }
            again=false;
 for(i=1;i<=nm1;i++){
            dp[i]=pg[i]-pc[i]-pe[i];
            if(fabs(dp[i])>epsp) again=true;
 }
 for(i=1;i<=npq;i++){
            j=nm1+i;
            dq[j]=qg[i]-qc[i]-qe[i];
            if(fabs(dq[j])>epsq) again=true;
 }
```

```
if(again){
ik=ik+1;
for(i=1;i<=nm1;i++){
        ui=um[i]; uui=ui*ui;
        for(j=1;j<=nm1;j++){
        uj=um[j]; angd=ua[i]-ua[j];
        gij=g[i][j]; bij=b[i][j];
        if(i==j) jacob[i][j]=-qe[i]-uui*b[i][i];
        else jacob[i][j]=ui*uj*(gij*sin(angd)-bij*cos(angd));
        }
        for(j=1;j<=npq;j++){
        gij=g[i][j]; bij=b[i][j];
        jj=j+nm1; uj=um[j]; angd=ua[i]-ua[j];
        if(i==j) jacob[i][jj]=pe[i]+uui*g[i][i];
        else jacob[i][jj]=ui*uj*(gij*cos(angd)+bij*sin(angd));
        }
}
for(i=1;i<=npq;i++){
        ii=nm1+i; ui=um[i]; angi=ua[i];
        for(j=1;j<=nm1;j++){
        gij=g[i][j];bij=b[i][j];
        uj=um[j];angj=ua[j];angd=angi-angj;
        if(i==j)jacob[ii][j]=pe[i]-ui*ui*g[i][i];
        else if(j<=npq)jacob[ii][j]=-jacob[j][ii];
        else jacob[ii][j]=-ui*uj*(gij*cos(angd)+bij*sin(angd));
        }
        for(j=1;j<=npq;j++){
        jj=nm1+j;
        if(i!=j)jacob[ii][jj]=jacob[i][j];
        else jacob[ii][ii]=qe[i]-ui*ui*b[i][i];
        }
}
invert(jacob,nj,jacobinv);
for(i=1;i<=nm1;i++){
        da[i]=0;du[i+nm1]=0;
}
for(i=1;i<=nm1;i++){
        for(j=1;j<=nj;j++){
        if(j<=nm1) da[i]=da[i]+jacobinv[i][j]*dp[j];
        else da[i]=da[i]+jacobinv[i][j]*dq[j];
        k=i+nm1;
        if(j<=nm1) du[k]=du[k]+jacobinv[k][j]*dp[j];
        else du[k]=du[k]+jacobinv[k][j]*dq[j];
        }
}
for(i=1;i<=nm1;i++) ua[i]=ua[i]+da[i];
for(i=1;i<=npq;i++) um[i]=um[i]+du[i+nm1]*um[i];
}
}
for(i=1;i<=nm1;i++){
j=ind[i];
if(j>npq){qg[i]=qe[i]+qc[i];da[i]=ua[i]*conv;}
}
pg[n]=pe[n]+pc[n];
if((ik==maxit)&&(again)){
fprintf(f,"The solution is not converged in %d iterations\n",ik);
```

```
}
else{
        fprintf(f,"\n**BUS RESULTS ARE AS FOLLOWS**\n");
        fprintf(f,"\nBUS NO.\tVM\t\tTHETA\t\tPC\t\tQC\n");
        for(i=1;i<=n;i++){
        j=ind[i];
        uad[j]=ua[j]/conv;
        if(typ[i]==1) fprintf(f,"%d\t%f\t",i,um[j]);
        else fprintf(f,"%d\t%f\t",i,um[j]);
        fprintf(f,"%f\t",uad[j]);
        fprintf(f,"%f\t%f\n",pe[j],qe[j]);
        }
/*line flow*/
        pg[n]=pe[n]+pc[n];/*For the swing bus too*/
        qg[n]=qe[n]+qc[n];
        fprintf(f,"**********************LINERESULTS***********************\n");
        for(i=1;i<=72;i++) fprintf(f,"-");
        fprintf(f,"\nFROM\tTO\tPLINE\t\tQLINE\n");
        for(i=5;i<=72;i++) fprintf(f,"-");
        fprintf(f,"\n");
for(i=1;i<=n1;i++){
jj=line[i][1]; kk=line[i][2]; j=ind[jj]; k=ind[kk];
gjj=z1[i][1]; bjj=z1[i][2]; gkk=z1[i][3]; bkk=z1[i][4];
gjk=g[j][k]; bjk=b[j][k]; gkj=g[k][j]; bkj=b[k][j];
angd=ua[j]-ua[k];
usqj=um[j]*um[j]; uskj=um[k]*um[j]; csd=cos(angd);
usqk=um[k]*um[k]; usc=uskj*csd; ssd=sin(angd); ucs=uskj*ssd;
pjk=gjj*usqj+gjk*usc+bjk*ucs;
pkj=gkk*usqk+gkj*usc-bkj*ucs;
qjk=-bjj*usqj-bjk*usc+gjk*ucs;
qkj=-bkk*usqk-bkj*usc-gkj*ucs;
mvajk[i]=sqrt(pjk*pjk+qjk*qjk);
mvakj[i]=sqrt(pkj*pkj+qkj*qkj);
fprintf(f,"%d\t%d\t%f\t%f\n",jj,kk,pjk,qjk);
fprintf(f,"%d\t%d\t%f\t%f\n",kk,jj,pkj,qkj);
}
ploss=0.0; qloss=0.0;
for(i=1;i<=n;i++){
ploss=ploss+pg[i]-pc[i];
qloss=qloss+qg[i]-qc[i];
}
        fprintf(f,"The total line losses are\n");
        fprintf(f,"%f+j%f\n",ploss,qloss);
fclose(f1);
}
/*************************************************************************/
if(fabs(ploss-Pl)<0.0001)break;
Pl=ploss;
}/*loss loop end*/
/*voltage violation */
for(i=1;i<=n;i++){
        if((um[i]<0.90)||(um[i]>1.10)){
        for(tt=1;tt<=total_bit_reqd;tt++)
        bit[ge][tt]=random(2);
goto top;
        }
```

```
}
/*mva violation*/
for(i=1;i<=n1;i++){
if((mvajk[i]>MVAM[i])||(mvakj[i]>MVAM[i])){
        for(tt=1;tt<=total_bit_reqd;tt++)
        bit[ge][tt]=random(2);
goto top;
}
}
co[ge]=0;
for(i=1;i<=ngen;i++)
co[ge]+=A[i]+B[i]*Pg[ge][i]*100+C[i]*Pg[ge][i]*Pg[ge][i]*10000;
for(i=1;i<=ngen;i++) printf("%.3f\t",Pg[ge][i]);
printf("%.3f\n",co[ge]);
if(optco[tt1]>co[ge]){
        optco[tt1]=co[ge];
        for(j=1;j<=ngen;j++)
        optPg[tt1][j]=Pg[ge][j];
        }
/*calculation of fitness value*/
co[ge]=10000.0/co[ge]*co[ge];fitness+=co[ge];
}/*ge loop end*/
/*calculation of probabilities*/
for(ge=1;ge<=pop_size;ge++) prob[ge]=co[ge]/fitness;
/*calculation of cumalative probabilities*/
for(ge=1;ge<=pop_size;ge++){
q[ge]=0.0;
        for(j=1;j<=ge;j++) q[ge]+=prob[j];
}
/*reproduction algorithm*/
for(ge=1;ge<=pop_size;ge++){
ra1[ge]=random(100.0)/100.0;
if(ra1[ge]<q[1]) count[ge]=1;
        for(i=1;i<pop_size;i++){
        if((ra1[ge]>q[i])&&(ra1[ge]<q[i+1])){
                        count[ge]=i+1;
                        break;

                }
        }
}
/*crossover algorithm*/
probc=0.8;
k=0;
for(ge=1;ge<=20;ge++){
ra2=random(10000.0)/10000.0;
        if(ra2<probc){
        k+=1;
        temp2[k]=count[ge];
        }
}
temp=k/2.0;
temp=k-2.0*temp;
if(temp==1){k+=1;temp2[k]=count[1];}
for(i=1;i<=k/2;i++){
ra3=random(total_bit_reqd);
```

```
ra4=random(total_bit_reqd-ra3);
          for(j=ra3;j<=ra4;j++){
          temp=bit[temp2[i]][j];
          bit[temp2[i]][j]=bit[temp2[i+1]][j];
          bit[temp2[i+1]][j]=temp;
          }
          for(j=ra3;j<=total_bit_reqd;j++){
          temp=bit[temp2[i]][j];
          bit[temp2[i]][j]=bit[temp2[i+1]][j];
          bit[temp2[i+1]][j]=temp;
          }
}
/*mutation algorithm*/
probm=0.01;
for(ge=1;ge<=pop_size;ge++){
ra2=random(10000.0)/10000.0;
          if(ra2<probm){
          ra2=random(pop_size*total_bit_reqd);
          temp3=ra2/total_bit_reqd;
          temp=ra2-temp3*total_bit_reqd;
          if(bit[temp3+1][temp]==0)bit[temp3+1][temp]=1;
          else bit[temp3+1][temp]=0;
          }
}
printf("%d\t",tt1);
for(i=1;i<=ngen;i++) printf("%.3f\t",100*optPg[tt1][i]);
printf("%f\n",optco[tt1]);
printf("************************\n");
if(OPTCO>optco[tt1]){
          OPTCO=optco[tt1];
          for(j=1;j<=ngen;j++)
          OPTPG[j]=optPg[tt1][j];
          }
}/*super loop end*/
printf("**************%%%%%%%%%%%%%%%%%%************\n");
for(i=1;i<=ngen;i++) printf("%.3f\t",100*OPTPG[i]);
printf("%.3f\n",OPTCO);
ttt=clock()/CLK_TCK;
printf("TIME == %f\n",ttt);
getch();
}
```

```
/*INVERSE PROGRAM*/
#define maxbus 30

void invert(float  zb[2*maxbus][2*maxbus],int size,float yb[2*maxbus][2*maxbus])
{
int i,j,k;
for(i=1;i<=size;i++)
{
for(j=1;j<=size;j++)
yb[i][j]=zb[i][j];
}
for(i=1;i<=size;i++)
{
yb[i][i]=1.0/yb[i][i];
        for(j=1;j<=size;j++)
                if(i!=j){
        yb[j][i]=yb[j][i]*yb[i][i];
        for(k=1;k<=size;k++)
        if(k!=i){
        yb[j][k]=yb[j][k]-yb[j][i]*yb[i][k];
if(j==size) yb[i][k]=-yb[i][i]*yb[i][k];
        }
}
}
k=size-1;
for(j=1;j<=k;j++)
yb[size][j]=-yb[size][size]*yb[size][j];
}
```