HYBRID GA BASED OPTIMAL POWER FLOW SOLUTIONS

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering

in

Power Systems & Electric Drives

Thapar University, Patiala

By:

SUPRIYA

(Regn No.-80741022)

Under the supervision of:
Mr. Parag Nijhawan
Sr. Lecturer, EIED

JUNE 2009

ELECTRICAL & INSTRUMENTATION ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled **"HYBRID GA Based Optimal Power Flow Solutions"**, in partial fulfilment of the requirement for the award of Master of Engineering in *Power Systems & Electric Drives* submitted in Electrical & Instrumentation Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Mr. Parag Nijhawan, Sr. Lecturer, EIED.

The matter presented in this thesis has not been submitted for the award of any other degree of this and any other university.

Supriya
8/07/09

SUPRIYA

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

Parag
08/07/09

Parag Nijhawan

Sr. Lecturer

Electrical & Instrumentation Engineering Department

Thapar University, Patiala

Countersigned by:

S. Ghosh
10/7/09

Dr. S. Ghosh

Professor & Head

Electrical & Instrumentation Engineering Department

Thapar University

Patiala

RK Sharma
15/7/09

Dr. R.K. Sharma

Dean (Academic Affairs)

Thapar University,

Patiala

i

ACKNOWLEDGEMENT

Words are often less to express one's deep regards. An understanding of work like this may not be the outcome of the efforts of a single person. I take this opportunity to express my profound sense of gratitude to all those who helped me through the duration of the thesis.

I express my foremost and my deepest gratitude to Mr Parag Nijhawan, Sr. Lecturer in EIED, Thapar University, Patiala for his valuable guidance, support and motivation throughout this work. I consider myself extremely fortunate for having got the opportunity to learn and work under his able supervision over the entire period of association with him.

I am very thankful to Dr S. Ghosh, Professor and Head, EIED Department for his constant encouragement, caring words and suggestions towards the successful completion of this work.

Last but not the least I would like to thank the almighty, who has always guided me to work on the right path of my life. I am also indebted to my parents, whose sincere prayers, best wishes, moral support have a constant source of strength and inspiration to me.

Supriya
08/07/09

SUPRIYA

(80741022)

ABSTRACT

An Optimal Power Flow is highly constrained and non optimization problem. The objective of the thesis is to present a solution of the optimal power flow problem through simulated annealing based genetic algorithm. Optimal power flow is generally considered as a minimization of the objective function. The main objective of this is to minimize the fuel cost and to keep the voltages, power outputs of generator within prescribed limits. In this the individual cost of each generating unit is assumed to be a function of active power generation. The proposed method solves the optimal power flow problem subjected to power balance equality constraints, limits on the control variable, limits on the dependent variables.

In order to solve the optimal power flow problem there are various classical methods such as Non linear programming (NLP), Linear programming(LP), Quadratic programming (QP), Newton based techniques, interior point methods etc. But these methods suffer from certain drawbacks, such as insecure convergence, algorithm complexity, week handling of qualitative constraints. Thus it becomes essential to develop optimization techniques that are efficient to overcome these drawbacks and handle such difficulties. Now a day's artificial intelligence techniques are used for solving the optimal power flow problem. Genetic algorithm is one of the best strategies for solving such problems because of their inherent parallel search capability. The searching ability of these methods can be improved by properly blending their characteristic features. In this thesis the simulated annealing (SA) are intermixed with genetic algorithm (GA) so as to develop a hybrid algorithm which helps in searching the better optimizing solution for IEEE 30 bus system.

CONTENTS

List of Figures

List of Tables

CHAPTER 1
OVERVIEW

1.1 BACKGROUND

For the growth of the economy power is considered as an important infrastructure. Throughout the entire world, with the change in time and infrastructure the electric power industry has gone under various changes for the improvement in power quality and it will continue for the several next decades also. Previously state wise individual power system was used but now days the interconnection of the power systems is done which has got many advantages. In the interconnected power system, the real and the reactive power of the generators are required to vary within certain limits in order to meet the particular load demand with minimum fuel cost. In the generation plant there are two factors which are to be considered at every load change. These are division of load and economic factor. But due to the deregulation in the industry and some unsolved problems, OPF is used to handle such systems .OPF have become reliable enough for the practical use and have taken the place of standard power system analysis tool [1]. Thus the optimal power flow solution is a program for scheduling power generation in such a way to minimize the fuel cost. The optimal power flow is used to achieve the following benefits:

1. Cost saving
2. Reduced system loss
3. Improved voltage control
4. Improved system security

Optimal power flow system is used to optimize the power flow solution for the large scale power system. In this thesis optimal power flow solution for the IEEE30 bus system is obtained. While going for the OPF system it is to make ensure that load demands are satisfied and no transmission system elements are overloaded.

Conventional methods such as Newton based method, Interior point method, Quadratic programming, Gradient based methods are reported in Literature review of the optimal power flow problem. But these methods suffer from certain limitations and are not able to solve the optimal power flow problem efficiently. This is because the conventional method depends upon the first and the second derivative of the objective function. Due to the above mentioned limitations the conventional methods are not used and are replaced by the artificial intelligence techniques such as genetic algorithm, fuzzy logic, artificial neural network etc. and heuristic functions such as hill climbing, Best search, A* algorithm, Simulated annealing or combination of these.

In this thesis the combination of artificial method and heuristic method is used to solve the IEEE 30 bus system for optimal power flow problem. The artificial technique and heuristic method used in this thesis is genetic algorithm and simulated annealing respectively. GA is a random search method and does not require derivative information. In SA the concept of annealing i.e. temperature is used.GA and SA is used to obtain the OPF solution for the IEEE 30 bus system. The results obtained are compared with those already reported in literature [2].

1.2 ORGANIZATION OF THESIS

The thesis is divided into six chapters. Chapter 1 gives the introductory part of the thesis. Chapter 2 reviews the range of published material on the OPF and methods applied to solve optimal power flow problem. Chapter 3 represents the formulation of flow problem. Chapter 4 describes genetic algorithm and its advantages. This chapter also introduces the concept of hybrid GA. Chapter 5 presents the algorithm of hybrid genetic algorithm based optimal power flow problem. The thesis ends up with Chapter 6 which gives the results of hybrid GA based OPF problem.

CHAPTER 2
LITERATURE REVIEW

2.1 INTRODUCTION

Power system engineers face many challenges. The challenges include that the system should work well and also meet the needs of the utility and consumers. Moreover it is also required to search economic alternatives, analyze the system performance and cost with precision. Traditionally the system was designed such that when the generation is dispatched economically there will be no violations of limits. Then economic dispatch is used for the transmission purposes but it is not sufficient because it ignores the constraints imposed by the transmission system [3].Thus the main objective is to optimize the power system operating objective function, such as generation cost or transmission losses, by satisfying the set of system operating conditions and constraints [4].

Several classical search methods and optimization techniques had been used from the last decade for solving the optimization problems. Now a day's several artificial intelligent computer based mathematical techniques are used for solving the optimization problem as classical search methods suffers from various drawbacks.

2.2 MATHEMATICAL TECHNIQUES

The various mathematical techniques are used for solving the OPF problem. Linear programming method requires the linearization of objective function as well as constraints with nonnegative variables. It linearizes both the objective function and the constraints in each iteration and it is better than one which linearize only objective function once. Newton-Raphson method is widely used to solve optimal power flow problem. K.L.Lo *et al.* [5] proposed Newton-like load flow methods, the Fixed Newton method and the modification of the right-hand-side vector method for line outage simulation that is a part of contingency analysis. X. Tong *et al.* [6] presented the semi smooth Newton-type algorithms for solving OPF problems. Quadratic Programming method is a special form of nonlinear programming whose objective function is quadratic and constraints are linear. J.A.Momoh [7] presented an extension of basic Kuhn-Tucker conditions and employing a generalized Quadratic-Based model for OPF. Karmarkar proposed a new method in 1984 for solving large-scale linear programming problems very efficiently by using linear programming method. It is known as an interior method since it finds improved search directions strictly in the interior of the feasible space. The drawback in applying linear programming is that the input output function is to be expressed as a set of linear functions, which may lead to loss of accuracy

As these methods suffers from the various drawbacks and can be explained as [3]

1 In some methods there is a problem in handling inequality constraints. This problem presents the differences of potentially non-convex or even disjoint feasible regions.

2. Linear programming method and non linear programming method are not suitable for constraints problem.

3. In Newton method the inequality constraints are added as quadratic penalty terms to the problem objective and multiplied by appropriate penalty multiplier. Newton method suffers from the difficulty in handling inequality constraints

4. These are not able to provide the optimal solution and usually getting stuck at a local optimum

5. All these methods are based on the assumption of continuity and differentiability of the objective function, which is not true in a practical system

6. All these methods cannot be applied with discrete variables which are transformer taps.

Due to these limitations, many other powerful deterministic, probalistic and stochastic techniques are used for solving the large dimensional optimization problem. These techniques include the evolutionary programming, genetic algorithm, Fuzzy logic method, Ant colony optimization or particle swarm optimization, genetic algorithm methods, heuristic methods etc.

Evolutionary Programming is a stochastic optimization method in the area of evolutionary computation, which uses the mechanics of evolution to produce optimal solution for a problem. The working of this consists of evolving a population of candidate solutions towards the global minimum through the use of mutation operator and selection scheme. A number of evolutionary inspired optimization techniques were developed Jason Yuryevich in 1999[8].Artificial Neural Network is an interconnected group of artificial neurons that uses a mathematical model or computational model for information processing based on a connectionist approach to computation. This is the part of artificial intelligence and is some was related with the human brain. The concept of neurons is basically given by Raman Y.Cajal (1911) and has made the study of its functioning comparatively easier. ANN can be applied to various optimization problems of the system. M.Tarafdar gives the application of ANN on power system [9]. Genetic Algorithm is proposed by Holland and is known to be an efficient search and optimization mechanism which incorporates the rules of natural selection. The development of parallel computers and microprocessors have made this algorithm one of the real interest and the birth of a new field of research, experimentation and application known as evolutionary computation.GA has a capability of parallelism and is used for solving stochastic optimization problems [10]. GA was originally proposed by Holland and then reformulated and customized by many other scientists. The genetic algorithm is characterized by the following [11].

1. The GA work with the base in the code of the variables group (artificial genetic strings) and not with the variables in themselves.

2. The GA work with a set of potential solutions (population) instead of trying to improve a single solution.

3. The GA do not use information obtained directly from the object function, of its derivatives, or of any other auxiliary knowledge of the same one.

4. The GA applies probabilistic transition rules, not deterministic rules.

There are some difference between GA and traditional searching algorithms [12] and can be summarized as follows:

1. The algorithms work with a population of string, searching many peaks in parallel, as opposed to a single point.

2. GA work directly with strings of characters representing the parameters set not the parameters themselves.

3. GA use objective function information instead of derivatives or others auxiliary knowledge.

4. GA has the potential to find solutions in many different areas of the search space simultaneously.

Genetic algorithm is similar to that of natural selection or with the biological constraints [13].In order to find the best solution or survival of the fittest from GA various genetic operators are used [20][14]. With the help of these operators the optimal solution can be found [15]. Although GA is used widely still it has some limitations in GA and do not allow to find solution in global optima [16]-[18]. Alex Rogers et al. describe the problem of genetic drift. [19].Tomoyuki et al. describes the parallel simulated annealing using genetic crossover to avoid the GA to stuck in local optima [20]. Liu Juan, Cai Zi-Xang, Liu Jian-qin introduced the novel approach of GA so as to avoid the GA to get stuck into local optima [21].Simulated annealing is one of the important algorithms to solve the optimization problem. It was first introduced by S.Kirpatrick [22].It resembles with the physical evaluation of a solid from a high temperature state to a thermal equilibrium state. It searches randomly around a neighbourhood of a present searching point. In order to find the optimal solution without getting stuck into local optima SA is used as a heuristic method with GA[23][24]. The next searching point can be accepted when the fitness value of the next point is worse than that of the present

CHAPTER 3
OPTIMAL POWER FLOW

3.1 INTRODUCTION

Optimal power flow is discussed by Carpentier in 1962. J.A. Momoh in 1972 studied about optimal power flow and describes the challenges of optimal power flow [1]. It describes OPF as a large non linear mathematical programming. Any power network problem in a steady state involves the adjustment of some controllable quantities to achieve a desired operating condition and can be formulated as an optimal power flow problem.

A wide variety of mathematical techniques are there to solve the optimal power flow problem such as nonlinear programming (NLP), quadratic programming (QP), linear programming, Newton-based techniques, and interior point methods. Generally, NLP based procedures have many drawbacks, such as insecure convergence properties and algorithmic complexity. QP-based techniques have some disadvantages associated with the piecewise quadratic cost approximation. Newton-based techniques have a drawback of the convergence characteristics that are sensitive to the initial conditions, and they may even fail to converge due to the inappropriate initial conditions. Although LP methods are fast and reliable, they have some disadvantages associated with the piecewise linear cost approximation. Interior point methods have been reported as computationally efficient; however, if the step size is not chosen properly, the sub linear problem may have a solution that is infeasible in the original nonlinear domain. Unfortunately, the problem of the OPF is a highly nonlinear and a multimodal optimization problem (i.e., there exist more than one local optimum). Hence, local optimization techniques, which are well elaborated, are not suitable for such a problem. Moreover, there is no local criterion to decide whether a local solution is also the global solution. Therefore, conventional optimization methods that make use of derivatives and gradients, in general, are not able to locate or identify the global optimum. On the other hand, many mathematical assumptions, such as convex, analytic, and differential objective functions, have to be given to simplify the problem; however, the OPF problem is an optimization problem with, in general, non convex, non-smooth, and nondifferentiable objective functions. Hence, it becomes essential to develop optimization techniques that are efficient to overcome these drawbacks and handle such difficulties. Recently, heuristic algorithms, such as genetic algorithms (GA) and evolutionary programming, have been proposed for solving the OPF problem

3.2 POWER FLOW PROBLEM VARIABLES

Power flow is basically a study of steady state operation of power system. Power flow is a tool for investigating and computing these constraints. It basically computes the following four variables:

1. Voltage magnitude (V)

2. Voltage angle (δ)

3. Real power (P)

4. Reactive power (Q)

Out of four variables, two are variables are known or specified at each bus. Based on the specified variables, there can be following 3 types of buses in a power system network:

1. Voltage controlled bus / Generator bus (PV bus)

2. Load bus (PQ bus)

3. Slack bus / Swing bus

3.2.1Voltage Controlled bus

This is also known as generation bus and P-V bus, and n this bus the voltage magnitude corresponding to generation voltage and true or active power P corresponding to its ratings are specified. Voltage magnitude is maintained constant at a specified value by injection of reactive power. The reactive power generation Q and phase angle δ of the voltage are to be computed.

3.2.2Load bus

This is called the P-Q bus and at this bus the total injected power is specified. i.e. active and reactive power injected into the network at this bus. It is required to specify only P and Q at such a bus as at a load bus voltage can be permitted to vary within the permissible values, i.e. 5%. At this bus magnitude and phase angle of the voltage are to be computed.

3.3.3 Slack bus

One of the generation buses in the power system is chosen as a slack bus or swing bus. At this bus the magnitude and phase angle are specified. If slack bus is not specified then the generation bus with usually with a maximum active power P is taken as reference bus.

Thus we can say that at each bus out of four variables two variables are specified and two are unspecified and this can be shown in tabular form

Bus Type	Specified	Unspecified	No of unknowns
Voltage controlled	P,V	Q, δ	$2(N_g-1)$
Load bus	P,Q	V, δ	$2(N-N_g)$
Slack bus	V, δ	P,Q	2
Total	2N	2N	

Table 3.1 Bus types and variables

N = No. of buses (generator and load buses)

N_g = No. of generator including the generator at the slack bus.

3.3 OPTIMAL POWER FLOW

In the power flow the power system which is to be analyzed is assumed to be balanced three phase system under steady state conditions. By calculating the unknown variables in the power flow does not lead to the optimal operating conditions of a power system, because there exists infinitely variable choices in specifying a balanced load flow solution.eg the power generated has to be within the constraints of demand and voltage at each bus and has to be within the constraints of system security. These constraints should be satisfied for a wide range of control values and performance will ensure that the system is operating in optimal manner. This is known as optimal power flow [3].

3.3.1 Definition

The general definition of optimization problem is given by [8].

$$\text{Minimize: } f(x, u) \tag{3.1}$$

$$\text{Subject to: } g(u, x) = 0 \tag{3.2}$$

$$h(u, x) \geq 0 \tag{3.3}$$

Where

f: objective function

g: equality constraints

h:inequality constraints

u and x represents a set of controllable and dependent variables respectively.

In general most of the constraints and objective function of the equations (3.1-3.3) represents the operational constraints and economical aims respectively. Optimization of the variables of these function are either continuous or discrete. Continuous variables are dependent variables like voltage magnitude and voltage angle and reactive power of generation units. Capacitor/reactor bank steps, load shedding ratio and transformer taps are also control variables.

The task is to minimize the objective function, while satisfying the system constraints. The applicability of this definition to different classes of problem is by selecting suitable objective function to be minimized under the suitable sets of controllable quantities and suitable sets of equality and inequality constraints.

3.4 OPF: PROBLEM FORMULATION

The optimal power flow problem can be defined by specifying the following five attributes and can be explained as

1. Objective function

2. The controls

3. The dependent variables

4. The equality

5. Inequality constraints

3.4.1 Objective function

It can be a meaningful scalar function of the variable of the problem. The basic purpose of the OPF is to minimize the operating cost i.e cost minimization. It means to minimize the total cost of real power generation. Thus the objective function is the function of the real power generation [4].

$$F_i = a_i + b_i(Pg_i) + c_i(Pg_i)^2 \tag{3.4}$$

Where:

Pg_i is the amount of generations in MW at generator i

a_i, b_i, c_i are the unit cost curve of the generator i

The objective function for the entire power system is expressed as the sum of quadratic cost model at each power plant and is given as:

$$F_i = \sum_{i=1}^{N} a_i + b_i(Pg_i) + c_i(Pg_i)^2 \tag{3.5}$$

Where

N is the number of generation including the slack bus.

3.4.2 Control Variables

The control variables in an optimal power flow problem are the quantities whose value can be adjusted directly to help minimize the objective function and satisfy the constraints. The control variables can be given as:

1. Active power generation

2. Reactive power generation

3. Transformer tap ratio

4. Generator bus voltage.

Different classes of the optimal power flow problem restrict the quantities that can be controlled. For instance an OPF algorithm for minimizing the active power generation cost might limit the controls to active power generation. The aim of the OPF is to adjust the control variables in order to minimize the total operating cost of meeting the particular load demand for a power system

3.4.3 Dependent variables

These variables are the optimal power flow variables that are not control. These include all type of variables that are free, within limits, to assume value to solve the problem. The main dependent variables are the complex bus voltage angles and magnitude.

3.4.4 Equality Constraints

In order to minimize the equation (3.4) it is essential to know that whether the power system is running under normal conditions, i.e load and losses i.e power demand is satisfied and the network components are operating within limits [4]. This can be achieved by the active and reactive power analysis:

$$P_i = P_{Load} + P_{Loss}$$

$$Q_i = Q_{Load} + Q_{Loss}$$

Where

P_i & Q_i are the active and reactive power outputs.

P_{Load} & Q_{Load} are the active and reactive load power.

P_{Loss} & Q_{Loss} are the active and reactive power loss.

The power flow equations of the network can be given as:

$$g(V, \delta) = 0$$

where

$$g(V, \delta) = \begin{cases} P_i(V,\delta) - P_i^{net} \\ Q_i(V,\delta) - Q_i^{net} \\ P_m(V,\delta) - P_m^{net} \end{cases}$$

P_i & Q_i are the calculated real and reactive power at PQ bus

P_i^{net} & Q_i^{net} are the specified real and reactive power for the PQ bus

V & δ are the magnitude and phase angle of voltage at different buses.

3.4.5 Inequality Constraints

In a power system components and devices have operating limits, & these limits are created for the security constraints. Thus the required objective function can be minimized by maintaining the network components within the security limits. This brings the concept of inequality constraints. The most usual type of inequality constraints are the upper bus voltage limits at generation at load buses, lower bus voltage limits at generation at load buses, lower bus voltages limits at some generators, maximum line loading limits and limits on tap setting .These includes the following:

The inequality constraints on real power generation at bus i^{th}

$$Pg_i^{\min} \le Pg_i \le Pg_i^{\max}$$

$Pg_i^{\min}$ & $Pg_i^{\max}$ are the max and min value of real power generation at i^{th} bus.

The inequality constraints on reactive power generation Qg_i at each PV bus

$$Qg_i^{\min} \le Qg_i \le Qg_i^{\max}$$

$Qg_i^{\min}$ & $Qg_i^{\max}$ are the max and min value of reactive power generation at PV bus

The inequality constraint on Voltage magnitude V of each PQ bus

$$V_i^{\min} \le V_i \le V_i^{\max}$$

V_i^{min} & V_i^{max} are the maximum and minimum values of voltage at bus i

The inequality constraints on phase angle δ of voltages at all buses i

$$\delta_i^{min} \leq \delta_i \leq \delta_i^{max}$$

δ_i^{min} & δ_i^{max} are the maximum and minimum values of the phase angle at buses i

The inequality constraints on MVA flow limit

$$MVA_{ij} \leq MVA_{ij}^{max}$$

MVA_{ij}^{max} = MVA rating of transmission line connecting ith and jth buses

3.5 OPF APPLICATION

The various applications of OPF are described as [1]:

1. Base –case development: This is the most common application of the OPF. Dozens of base cases can be efficiently developed by following the same set of design rules

2. Voltage Instability, Maximum transfers (V-P Curve Approach) or minimum compensation requirements (Q-V Curve Approach) which are to attain voltage stability can be obtained in a single solution. Other constraints such as voltage and/or thermal limitations can also be considered.

3. Flexible AC Transmission Systems (FACTS): OPFs will likely be used to “dispatch” the transmission network (e.g., series and shunt compensation) to overcome post-disturbance thermal and/or voltage violations

4. Economic dispatch, subject to: thermal constraints, voltage constraints, interface constraints (e.g., stability) and spinning reserve constraints. From this dispatch, marginal costs and transmission losses are easily identified

CHAPTER 4
GA AND HYBRID GA

4.1INTRODUCTION

GA is a branch of artificial intelligence, and artificial intelligence is an area of computer science that is concerned with the designing of intelligent computer system which exhibits the characteristics of intelligence in human behaviour. This branch is also related with the automation of intelligent behaviour. In the artificial intelligence we generally go for the probabilistic reasoning and it is predominately known as soft computing. The term soft computing was introduced by Lotif A. Zadeh of the University of California. According to him soft computing differs from the convention computing (Also known as hard computing) in its tolerance to imprecision, uncertainty and partial truth. Conventional computing is more oriented towards the mathematical approaches and therefore demands a high degree of precision. Whereas the soft computing basically inherent the characteristics of biological system and basically consist of Neural Networks (NN), Fuzzy Logic (FL) and Genetic algorithm (GA) [15]. Genetic algorithms are basically a family of computational models which are inspired by the biological evolution. These algorithms encode a potential solution to a specific problem on a simple chromosome like data structure and apply genetic operators to these structures so as to preserve critical information. Genetic algorithms are often seen as a function optimizer .Genetic algorithms can be applied to a wide range of problems. In order to solve the problem by genetic algorithm the first step is to create the population (random) of chromosomes. Then these chromosomes are evaluated in such a way that the chromosome which represents a better solution for the problem will be given more chances than the poorer solution.

4.1.1History of Genetic algorithm

Many human inventions were inspired by the natural selection. Genetic algorithm is one of them. The main idea of this is the survival of the fittest or in other words it is known as natural selection. In nature the individual that has better survival chances will survive for a longer period of time. This in turn provides a better chance to produce offspring with its genetic material. By virtue of this, the entire population consist of a lot of genes from the good population than that of bad population. In other words we can say that the fittest candidate will survive and unfit will not. This force of nature is called natural selection. The father of the original Genetic Algorithm was John Holland who invented it in the early 1970's.and thereafter he and his students contribute much to the development of this field. Holland research was not focused on optimization and domain specific practical problem but was on the concept of adaptation as seen in nature.[13][14]. Many people included biologists, are astonished that life at the level of complexity that we observe could have evolved in the relatively short time suggested by the fossil record [15]. The GA uses this power of evolution to solve optimization problems. As we are saying that the genetic algorithm is related with the nature, so there is some analogy between them and this can be described as

Genetic Algorithm	Nature
Optimization problem	Environment
Feasible solution	Individuals living in that environment
A set of feasible solution	Population of organism
Fitness function	Individual degree of adaptation
Operators used for results	Selection, recombination, mutation in nature

Table 4.1 Analogy b/w GA and nature

4.2 WORKING OF GENETIC ALGORITHM

Like any other optimization problem GA begins with the variables which are to be optimised eg cost function, transmission losses etc .In the end like other optimization algorithms it tries to converge. The input to the GA is the set of potential solutions to that problem, encoded in some fashion and called a fitness function that allows each candidate to examine qualitatively. The GA then evaluates each candidate according to its fitness function. As we are saying that that candidates are generated randomly so some of them may not working properly and are deleted from the pool rest of them shows activity, some are good and some show less activities[16]. The promising candidates are kept and allowed to reproduce. Multiple copies are made from them, but these multiple copies are not perfect as random changes are introduced during the copying

.

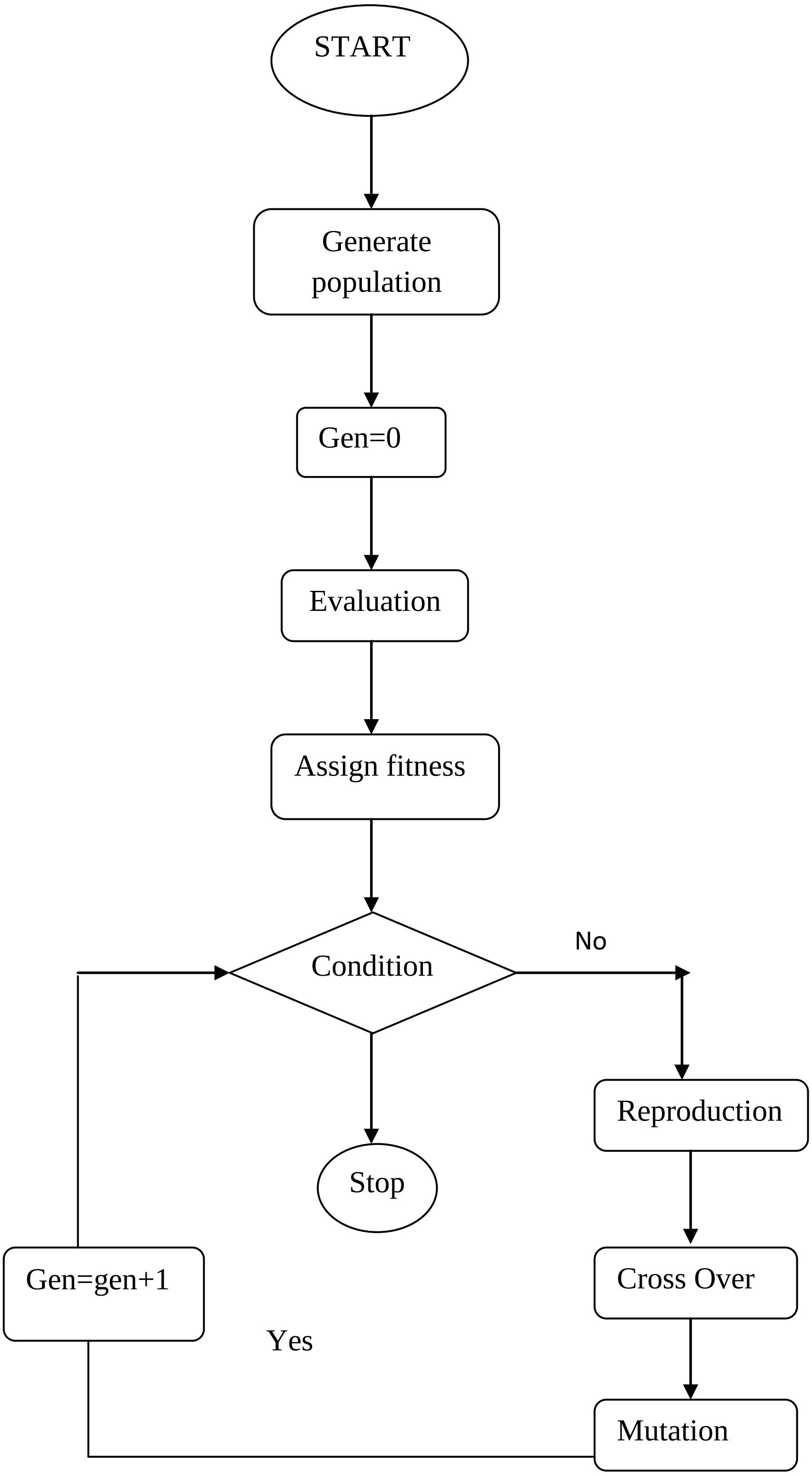


Figure 4.2 Flowchart of GA

These digital offspring then go to the next generation forming a new pool of candidate solution and are subjected to the second round of the fitness evaluation. The candidates which are worse are again deleted from the pool and best ones are considered. And every time GA operators work on it. This process is repeated so as to get the better solution. The simple working of GA can be under stood by flow chart shown above [14].

Thus if the conditions / best population is not generated then we go for the reproduction, crossover and mutation and then generation is incremented by 1 and fitness is checked.

4.3 GENETIC OPERATORS

In the genetic algorithm basically we use three types of operators and these can be described as [15]

1 Reproduction operator

2. Crossover operator

3. Mutation operator

4.3.1 Reproduction operator

It is also known as selection operator. The purpose of this operator is to make duplicates of good solution and eliminates bad solution in a population, while keeping the population size constant and this can be done by following:

1. Identify a good solution in population

2. Making multiple copies of solution

3. Eliminate bad solutions from the population so that multiple copies of good solutions can be placed in the population.

There exist a number of reproduction operators in GA literature, but the essential idea in all of them is that the above average strings are picked from the current population and there multiple copies are inserted in the mating pool. The various reproduction operators are:

1. Tournament selection

2. Ranking selection

3. Roulette wheel selection

Out of these operators roulette wheel selection operator is widely used .It is basically the proportionate operator where a string is selected in a mating pool with a probability proportional to its fitness. Thus the i^{th} string in the population is selected with a probability proportional to F_i .Since the population size is kept constant in a GA, the sum of the probability of each string being selected from the mating pools must be one. Thus the probability of the ith string is given by [15].

$$p_i = F_i \Big/ \sum_{i=1}^{n} F_i$$

N=population size

F=fitness

One way to implement this scheme is to imagine a roulette wheel with its circumference marked for each string proportionate to the string's fitness. The roulette wheel is spun n times, and by each time selecting an instance of the string chosen by the roulette wheel pointer. Since the circumference of the wheel is marked according to string fitness and this roulette wheel is expected to make $F_i \big/ \overline{F_i}$ copies of the i^{th} string in the mating pool. The average fitness of the population is calculated as

$$\overline{F_i} = \sum_{i=1}^{n} F_i$$

The following figure shows a roulette wheel for each individual (population)

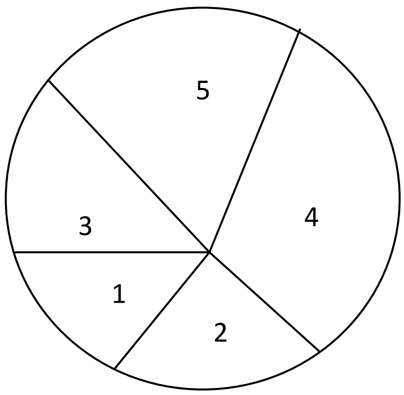


Figure 4.3 Roulette wheel

The different fitness value of each population is given in following table:

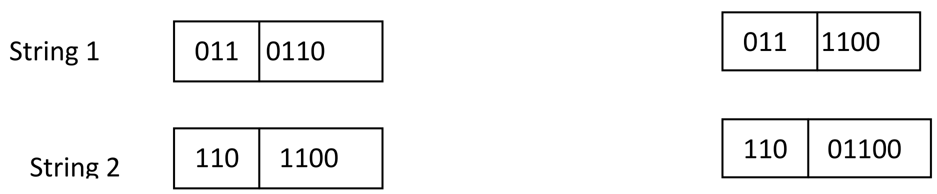
POPULATION	FITNESS VALUE
1	20
2	20.7
3	25
4	40
5	35

Table 4.4 Fitness value of each population

Since the 4^{th} value has a greatest fitness value as compared to other, so the roulette wheel will chose this value as compared to other individual thus by knowing the fitness value of each string the cumulative probability of each string can be easily calculate

4.3.2 Crossover operator

As the reproduction phase is over the pool is enriched with better solution. Reproduction operator makes clones of good string, but does not create new ones. Cross over operator is applied to the mating pool so as to form a string. Cross over operator is a recombination operator and will work in three steps. Firstly the reproduction operator selects a random pair of two individuals strings for mating, then a cross site is selected at random along the string length and the position values are swapped between two strings following the cross sites. The method of crossover used in GA is one point crossover. In this method a crossover site is selected randomly. The portion right of the selected site of these two strings are exchanged to form a new pair of strings. The new strings are thus a combination of two old strings



Before Cross over After Cross over

Figure 4.4 Crossover Operator

4.3.3 Mutation operator

Mutation adds new information in a random way search process and helps in getting stuck into the local optima [14] .It introduces diversity in the population when the population tends to become homogeneous due to the repeated use of reproduction and crossover operator. Mutation operates at the bit level, when the bits are being copied from the current string to the new string; there is probability that each bit become mutated. A simple genetic algorithm treats the mutation only as a secondary operator with the role of restoring the lost genetic materials. This probability is quite a small value, called the mutation probability. It is the probability of mutation which is used to calculate number of bits to be mutated. The mutation operator preserves the diversity among the population which is also very important for the search

The need of the mutation operator is to create a point in the neighbourhood of the current point. Eg consider a following population of 8 bit string

0110 1011

0011 1101

0001 0110

0111 1100

It is noted that in above four strings the left most bit is zero .The true optimum solution requires a one in that position, then neither reproduction nor crossover operator described above will be able to create one in that position

0110 1011

0011 1101

0001 0110

1111 1100

Figure 4.5 Mutation Operator

These three operators are simple to operate. By applying these operators on the current population a new population can be created. This new generated population is used to generate subsequent population and so on. The values of the objective function of the individuals of the new population are again determined by decoding the strings. These values express the fitness of the solutions of the new generations. This completes one cycle of genetic algorithm called a generation. In each generation if the solution is improved, it is stored as the best solution. This is repeated till convergence.

4.4 GA BASED OPF

Optimal power flow is generally solved by classical methods. Optimal power is considered as the minimization of an objective function within the constraints. Objective function can be either transmission losses, fuel cost etc. Constraints are the physical laws governing the power generation transmission system and the operating limits of the equipment [12].

The optimal power flow is limited by:

1. The high dimensions of the system

2. Incomplete domain dependent knowledge

These are due to the numerical optimization procedures which are based on the successive linearization which uses the first and second derivative of the objective function and their constraints.

A genetic algorithm offers a powerful approach to these optimization problems and it is possible due to the availability of high performance computers. Now a day’s these type of algorithms have been used extensively to solve the global optimization problem. Genetic algorithm is parallel and global search technique and it converges more towards the global solution because it evaluates simultaneously many points in parameter space. A simple Genetic Algorithm is an iterative procedure, which maintains a constant size population P of candidate solutions. During each iteration step (generation) three genetic operators (reproduction, crossover, and mutation) are there to generate new populations (offspring), and the chromosomes of the new populations are evaluated via the value of the fitness which is related to cost function. Based on these genetic operators and the evaluations, the better new populations of candidate solution are formed. The cost function for the OPF problem is defined by

$$F_i = \sum_{i=1}^{N} a_i + b_i(pg_i) + c_i(pg_i)^2$$

Our objective is to search P_{gi} in limits and to minimize the cost function and the losses.

4.5 LIMITATIONS IN GA

All though GA has got many advantages but at the same time it suffers from following disadvantages which can be described as:

1. Genetic drift

2. Premature convergence

3. Deceptive fitness function

4 Sometimes cross over does not occur

4.5.1 Genetic drift

This term is borrowed from the population genetics where it is used to explain change in gene frequency through random sampling of the population. This phenomenon is observed in the genetic algorithm due to the stochastic nature of the selection operator and is one of the mechanism through which the population converges to a single member [19]. Prugel –Bennett and Shapiro analyzes the selection scheme that shows that the change in mean fitness at each generation is a function of population fitness variance. Thus there is a loss in population fitness variance due to genetic drift and has a direct effect on performance of genetic algorithm.

4.5.2 Premature convergence

In genetic algorithm the term premature convergence means that a population for an optimization problem converged too early resulting in being suboptimal point. Relative to the other members in the population, if some of the average individuals have extra ordinary fitness, the mechanics of genetic algorithm may lead the iteration to a premature convergence which is defined as a convergence to a suboptimal point. In this the parental solution through the aid of genetic operator are not able to generate offspring's that are superior to their parents. Premature convergence can happen in case of loss of genetic variation. When a GA is applied to solve large scale and complex real world problems then the problem of pre mature convergence occurs. In this way the solving producer gets trapped in the local optimum and most of the operators can't produce offspring's suppressing their parents.[21].There are another two factors which lead to this undesired problem of genetic algorithm.

1. The selective pressure applied by the selection operator

2. The way by which the recombination or crossover is implemented.

4.5.3 Deceptive fitness function

The building blocks hypothesises suggest that Genetic algorithm work by combining low order building blocks to form higher building blocks .Therefore if in a function the low order building block do not combine to form higher order building blocks then the GA has a difficulty in optimizing the function and thus they form building blocks for suboptimal solution.[17]. Thus it will lead to the deceptive function and movement towards the optima actually reduces fitness

4.5.4 Sometimes cross over does not occur

Some times GA due to the improper handling of constraints cross over does not take place. Which leads to the loss of information from the chromosome by packing everything in a single number.[18].Moreover if the population becomes dominate for the longer time then it will converge and do not allow crossover to take place

Due to all these problems sometimes GA does not able to find good solution .In order to avoid this we go for the hybrid genetic algorithm.

4.6 HYBRID GA

Genetic algorithm proposed by Holland is an efficient search and optimization mechanism which incorporates the rules of natural selection.GA is being used in all kind of numerical and optimization problem with a high degree of success. This capability of GA is due to its inherent parallelism, which enables search in complex landscapes and to optimize various objective functions of interests in various fields of applications. One of the most important fields of application is solving stochastic optimization problems. GA was originally proposed by Holland, and then reformulated and customized by many other scientists [23]. In this thesis, an attempt is made to add more power to GA by using an SA like selection operator. GA is a specific form of a general optimizer. As many other search processes like SA, evolution strategies, evolutionary programming, tabu search etc are successfully applicable to a general optimization problem, inter-mixing the salient features of these algorithms may be found to be more effective in specific application areas. To develop the hybrid algorithm, we have combined the features of SA to the basic GA framework and arranged the major operations into several stages. The capability of SA for selecting the fittest candidate solutions are used as input to the cross-over stage [10]. Though SA takes some time to cool down to the equilibrium state, it eliminates the dependency of the selection process on a complete pool of candidate solutions required in conventional method at the selection stage. Both SA and GA are randomized guided search methods and are combined to result in hybrid genetic algorithm (HGA).

4.7 BASICS OF SIMULATED ANNEALING

Simulated Annealing (SA) is motivated by an analogy to annealing in solids. The idea of SA comes from a paper published by Metropolis et al in 1953. Basically annealing is defined as a process which is related with the temperature i.e. sudden heating and cooling. In general terms a perfect defect-free crystal is much tougher than a crystal with lots of defects. Such a crystal can be produced by a carefully designed schedule of heating and cooling. This technique is used for finding a global extreme (minimum or maximum) of a function and was introduced by S. Kirkpartick, C.D. Gelatt and M.P. Vecchi Science in 1983. It has been used in many applications, e.g., for designing integrated circuits with millions of elements placed so as to minimize interference between their connecting wires [22]. This method is inspired by experimental observations on crystallization. At high temperatures, the atoms are free to move around the sample. As the temperature is reduced, the atoms tend to crystallize into a solid. If the sample is quenched, i.e., cooled very rapidly, then the solid is usually polycrystalline or amorphous in form. If the sample is annealed, i.e., cooled slowly, then the sample stands a better chance of forming a perfect crystal, which is the global minimum energy configuration of the system. Quenching typically leads to the bottom of the nearest valley and annealing allows the system to explore the landscape and settle down into one of the lower valleys. The key to successful annealing is to use a good annealing schedule, i.e. a protocol for gradually reducing the temperature of the sample. Simulated annealing in statistical physics and is based on the concept of thermal equilibrium at temperature T.

Physical annealing	Simulated annealing
System States	Feasible solution
Energy	Cost
Change of state	Neighboring solution
Temperature	Control parameter
Frozen state	Heuristic solution.

Table 5.1 Analogy b/w Physical annealing and simulated annealing

4.8 ADVANTAGES OF HYBRID GENETIC ALGORITHM

The advantages of hybrid genetic algorithm can be explained as [10][23]

1. Capability Enhancement

A technique can be utilized within a genetic algorithm to enhance the search capabilities. A genetic algorithm is normally viewed as a global search method that can capture the global view of a problem domain. Different techniques can be incorporated within a genetic algorithm to improve its performance in different ways. When a genetic algorithm as a global search method is combined with a problem-specific method as a local method, the overall search capability can be enhanced.

2. Improving solution quality

Local search methods and genetic algorithms are usually viewed as two complementary tools. A local search algorithm's ability to locate local optima with high accuracy complements the ability of genetic algorithms to capture a global view of the search space. Holland suggested that the genetic algorithm should be used as a pre-processor for performing the initial search, before invoking a local search method to optimize the final population. Bilchev and Parmee for example, used their ant colony optimization model for continuous search spaces as local search method to improve the quality of the solutions produced by a genetic algorithm in order to solve a real-world, heavily constrained, engineering design problem.

3. Improving efficiency:

The efficiency of a local search in reaching a local optimum integrates the efficiency of a genetic algorithm in isolating the most promising basins of the search space. Therefore, incorporating a local search into a genetic algorithm can result in an efficient algorithm. The efficiency of the search can be enhanced in terms of the time needed to reach the global solution, and/or the memory needed to process the population

CHAPPTER 5

HYBRID GA BASED OPF

5.1 INTRODUCTION

Simulated annealing like genetic algorithm is an optimization procedure that performs randomized search in large, complex and multimodal search space for providing a near optimal solution. In simulated annealing a problem state is defined by the values of a number of parameters. The advantage of this method is that it accepts the worst solution with probability without discarding it.

5.2 OPTIMAL POWER FLOW via SA

In order to solve OPF as an optimization problem by using SA it is necessary to define an energy function E. It helps in finding the optimal solution and converges the solution and minimum generating cost [2]. The energy function is given by

$$E=\alpha\frac{\begin{bmatrix}\Delta P_p \\ \Delta Q_p\end{bmatrix}}{T}+\frac{F_i}{\beta}$$

Here

α & β =weighting factors

T= temperature.

$$\Delta P_p = P_G - P_c - \sum_{q=1}^{ng} V_p V_q \, Y_{pq} \cos(\partial_p - \partial_q - \theta_{pq})$$

$$\Delta Q_p = Q_G - Q_c - \sum_{q=1}^{ng} V_p V_q \, Y_{pq} \sin(\partial_p - \partial_q - \theta_{pq})$$

Here

P_G and Q_G are real and reactive power generation at bus

P_c and Q_c are the real and reactive power demands at bus

V_p and V_q are the voltage magnitude at bus P and Q

Y_{pq} are the admittance magnitude.

δ_p and δ_q are the voltage bus angle at bus P and Q

θ_{pq} is the admittance angle

In this the weighting factors are used in order to obtain a normalized energy function. The first term is divide by the temperature T and it is done to obtain the optimal solution [2] In the SA technique two factors plays an important role and can be given as:

1. Acceptance criteria

2. Cooling Schedule

5.2.1 Acceptance criteria

The most important characteristic of SA is the probalistic acceptance of the bad solution.[2]. The law of thermodynamics states that at temperature T the probability of an increase in energy of magnitude δE is given by

$$P(\delta E) = \exp \frac{\delta E}{kT}$$

Where

k is constant and known as Boltzmann constant.

E= energy

The simulation in the Metropolis algorithm calculates the new energy of the system. If the energy has decreased then the system moves to this state. If the energy has increased then the new state is accepted using the probability returned by the above formula.

A certain number of iterations are carried out at each temperature and then the temperature is decreased. This is repeated until the system freezes into a steady state.

5.2.2 Cooling schedule

In the SA algorithm there is one important term i.e cooling schedule and it assumes that the annealing process continues till the temperature reduces to minimum value [23]. The cooling schedule for the simulated annealing consists of the three steps and can be described as:

1. Starting temperature
2. Temperature decrement
3. Final temperature.

Starting temperature

The starting temperature must be hot enough so that it allows move to almost any neighborhood state. If it is not so then the ending solution will be very close to the initial solution, or we can say that we simply implement Hill Climbing algorithm. If the starting temperature is high then the search can move to any neighbor and thus transform the search into a random search. The search will continue to be random until the temperature is cool enough to start acting as a simulated annealing algorithm

Temperature decrement

Once we have our starting and stopping temperature we need to get from one to the other. That is, we need to decrement our temperature so that we eventually arrive at the stopping criterion.

The way in which we decrement our temperature is critical to the success of the algorithm. Theory states that we should allow enough iteration at each temperature so that the system stabilizes at that temperature. Unfortunately, theory also states that the number of iterations at each temperature to achieve this might be exponential to the problem size. As this is impractical we need to compromise [24]. We can either do this by doing a large number of iterations at a few temperatures, a small number of iterations at many temperatures or a balance between the two.

Final temperature

In the temperature decrease until it reaches zero. However, this can make the algorithm run for a lot longer, especially when a geometric cooling schedule is being used .In practice, it is not necessary to let the temperature reach zero because as it approaches zero the chances of accepting a worse move are almost the same as the temperature being equal to zero.

Therefore, the stopping criteria can either be a suitably low temperature or when the system is “frozen” at the current temperature (i.e. no better or worse moves are being accepted).

5.3 ALGORITHM AND FLOW CHART OF HYBRID GA

5.3.1 Algorithm of Hybrid GA

Step1: Enter the generator data, bus data, capacitor/reactor data, transformer data, and transmission line data.

Step 2: Assume suitable population size (pop size), maximum number of generations i.e (gen_max).

Step 3: Initially set the population counter is equal to zero.

Step 4: Generate the chromosome randomly.

Step 5: Run Newton-Raphson method for each set of generating patterns Pgi corresponding to a particular generation and then determine, slack bus generation, bus voltage magnitudes and phase angles at all the buses. Also calculate power flow in each transmission line of the system.

Step 6: Check the following Constraints:

*Check the voltage magnitude violation:

$$V_i^{min} \leq V_i \leq V_i^{max}$$

*Check the bus voltage phase angle violation:

$$\phi_i^{min} \leq \phi_i \leq \phi_i^{max}$$

*Check the reactive power limits at all generator buses.

If any of the limits is violated go to step 4.

Step 7: If all the above constraints are satisfied, increment the population counter by 1. If population size is less than or equal to population size , go to step 4, otherwise go to next step.

Step8: Apply Simulated Annealing

*initialize the value of temp (T) any assumed parameter

* Calculate the value of energy E as

$$E=\alpha\frac{\begin{bmatrix}\Delta P_p\\ \Delta Q_p\end{bmatrix}}{T}+\frac{F_i}{\beta}$$

*Accept this energy with probability

$$P = \begin{cases} 1 & \text{if } E < 0 \\ \exp\dfrac{\partial E}{kT} & \text{otherwise} \end{cases}$$

Lower the value of T and find the optimized value.

Step9: Find and store minimum cost among all valid individual parents and corresponding generation pattern.

Step 10: Apply the crossover operator.

Step11: Again Run power flow using Newton-Raphson method for each set of new generating patterns and hence determine, slack bus generation, bus voltage magnitudes and phase angles at all the buses. Also calculate power flow in each transmission line of the system.

Step 12: Check constraints as mentioned in step 6.

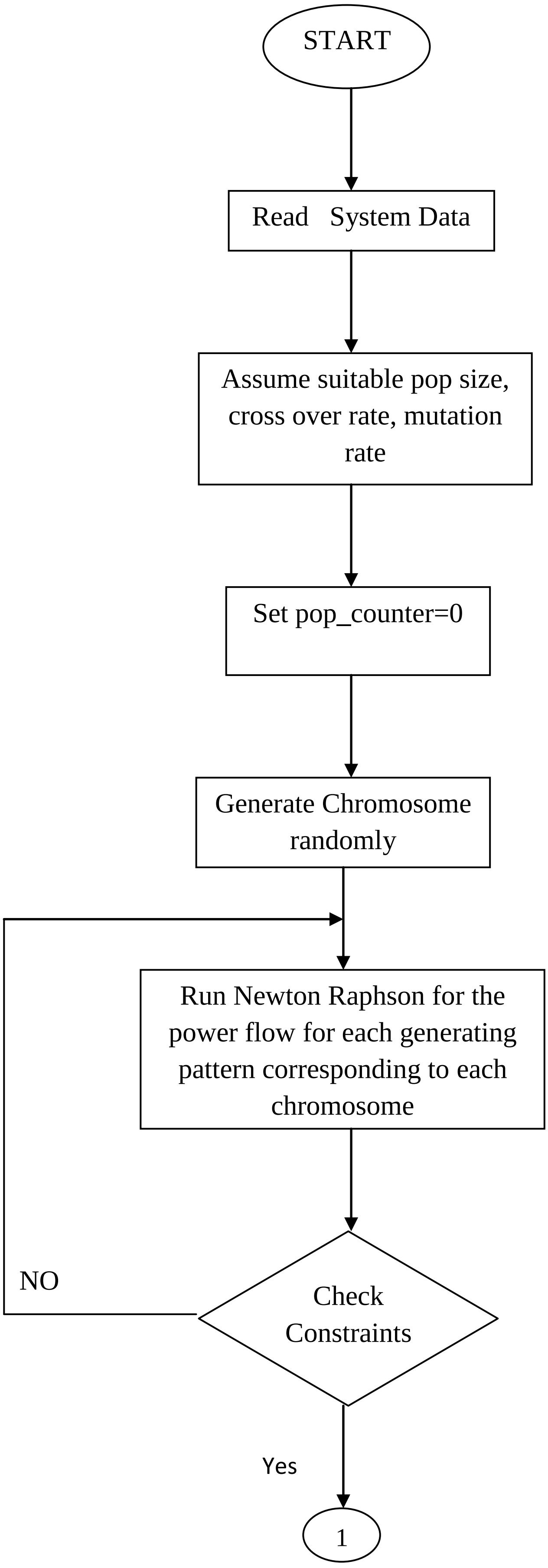
Step13: If all the constraints are satisfied, the individual of the new population becomes valid otherwise it becomes invalid.

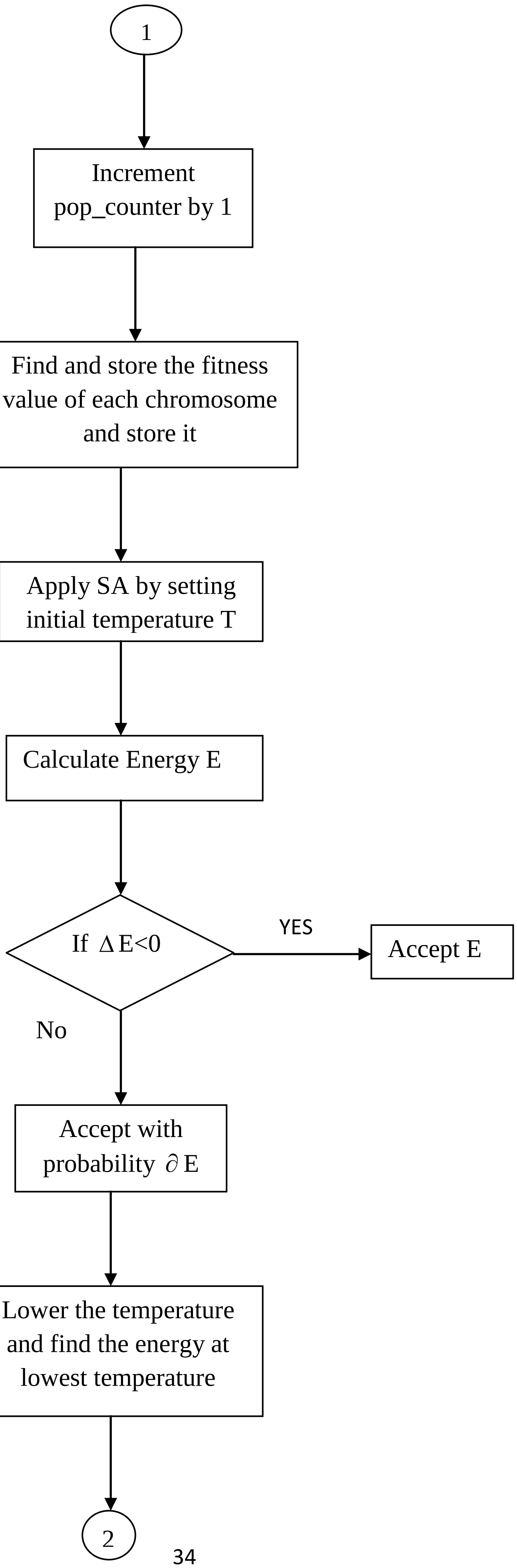
Step 14: Apply the mutation operator to the calculated generation patterns.

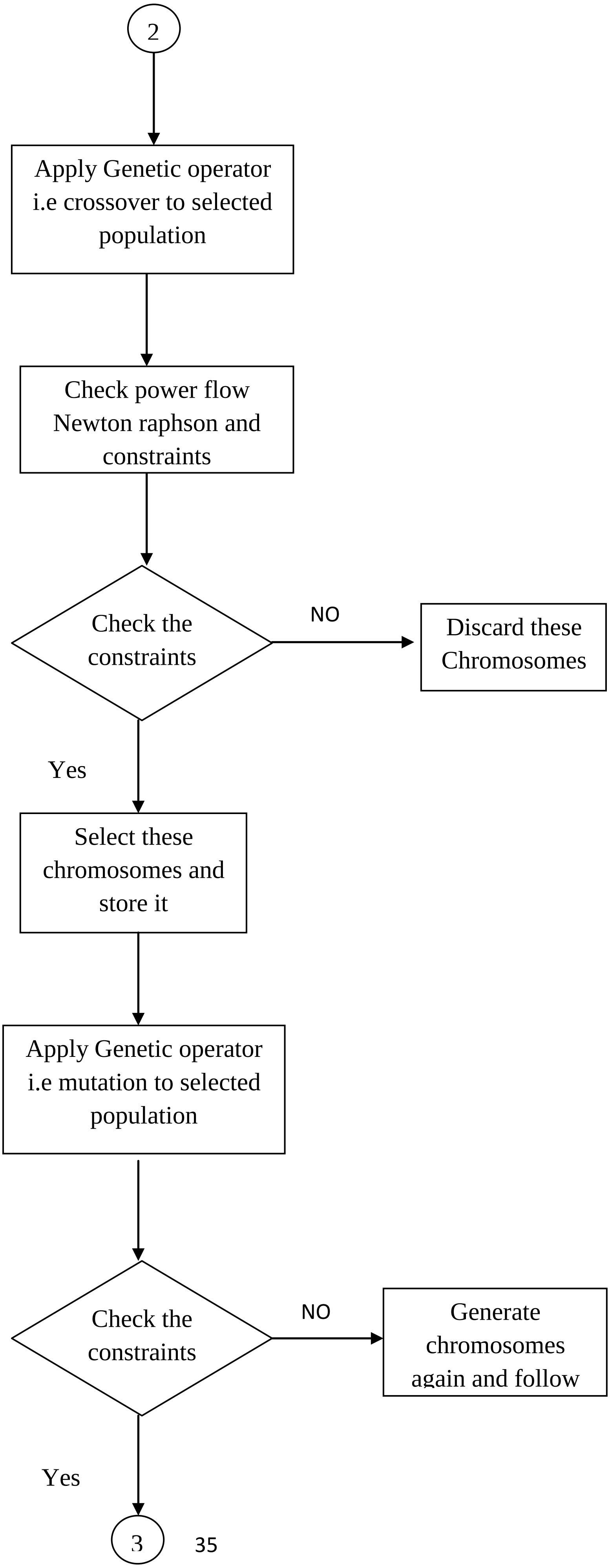
Step 15: Run power flow using the Newton raphson and check all the constraints as mentioned in step 6.

Step 16: Find the optimum solution among all population group

5.3.2 Flow chart of hybrid GA




1
Increment pop_counter by 1
Find and store the fitness value of each chromosome and store it
Apply SA by setting initial temperature T
Calculate Energy E
If ΔE<0
YES
Accept E
No
Accept with probability ∂E
Lower the temperature and find the energy at lowest temperature
2


2
Apply Genetic operator i.e crossover to selected population
Check power flow Newton raphson and constraints
Check the constraints
NO
Discard these Chromosomes
Yes
Select these chromosomes and store it
Apply Genetic operator i.e mutation to selected population
Check the constraints
NO
Generate chromosomes again and follow
Yes
3

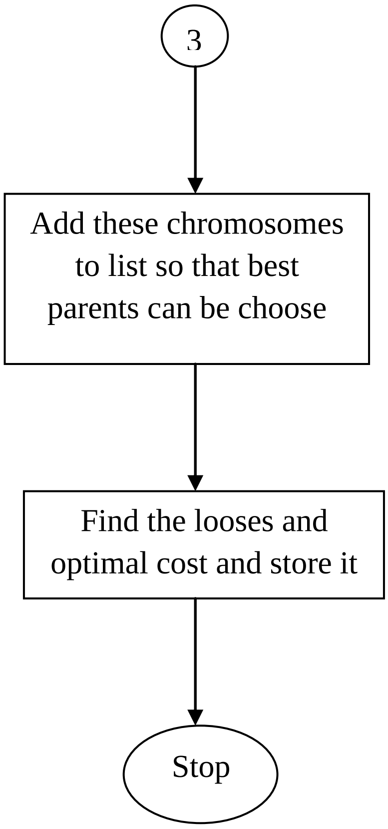


Figure 5.1 Flow Chart oh Hybrid GA

CHAPTER 6

RESULT &FUTURE SCOPE

6.1 HYBRID GA BASED RESULTS

The Hybrid GA based optimization method is applied to IEEE 30 bus system. Its bus data line data, cost coefficients are given in Appendix A. The generator real power is taken as a control variable.

After running the program the different values of generation are shown in table.

Table 6.1 Line losses with Hybrid GA

LINE		Power at bus and line flow			Line loss		Transformer Tap
From	To	MW	Mvar	MVA	MW	Mvar	
1		179.014	-9.498	179.221			
	2	124.694	-9.237	125.036	2.75	2.402	
	3	54.310	-0.253	54.310	1.211	0.559	
2		0.244	3.093	3.038			
	1	-121.97	11.640	122.534	2.175	2.402	
	4	26.149	-3.264	26.352	0.366	-2.768	
	5	58.106	1.279	58.120	1.499	1.952	
	6	37.959	-6.562	38.122	0.796	-1.532	
3		-2.400	-1.200	2.683			
	1	-53.099	0.812	53.105	1.211	0.559	
	4	50.699	-2.012	50.739	0.322	0.042	
4		-7.600	-1.600	7.767			
	2	-25.783	0.495	25.788	0.366	-2.768	
	3	-50.377	2.053	50.419	0.322	0.042	

	6	51.960	-14.613	53.975	0.331	0.211	
	12	16.601	10.464	19.625	0.000	0.820	0.932
5		-69.483	3.488	69.570			
	2	-56.607	0.704	56.612	1.499	1.952	
	7	-12.876	2.784	13.174	0.082	-1.860	
6		-14.664	9.545	0.000			
	2	-37.164	5.030	37.502	0.796	-1.532	
	4	-51.629	14.823	53.715	0.331	0.211	
	7	36.102	5.563	36.528	0.344	-0.693	
	8	19.519	-9.192	21.575	0.053	-0.757	
	9	9.470	-7.420	12.030	-0.000	0.276	0.978
	10	9.038	0.741	9.068	0.000	0.411	0.969
	28	14.664	-9.545	17.497	0.036	-13.431	
7		-22.800	-10.900	25.272			
	5	12.958	-4.644	13.765	0.082	-1.860	
	6	-35.758	-6.256	36.301	0.344	-0.693	
8		-19.144	7.169	20.421			
	6	-19.466	8.436	21.215	0.053	-0.757	
	8	0. 322	-1.267	1.308	0.001	-4.468	
9		0.169	5.091	0.000			
	6	-9.470	7.695	12.202	-0.000	0.276	
	11	-18.257	-10.757	21.191	0.000	0.832	

	10	27.896	8.153	29.063	0.000	0.828	
10		-5.801	17.000	17.962			
	6	-9.038	-0.330	9.044	0.000	0.411	
	9	-27.896	-7.325	28.842	0.000	0.828	
	20	6.652	3.834	7.678	0.050	0.111	
	17	0.969	4.613	4.714	0.007	0.017	
	21	15.848	10.977	19.279	0.117	0.252	
	22	7.664	5.231	9.279	0.057	0.117	
11		18.257	11.589	25.011			
	9	18.257	11.589	21.625	0.000	0.832	
12		-11.200	-7.500	13.479			
	4	-16.601	-9.644	19.199	0.000	0.820	
	13	-37.646	-11.217	39.282	0.000	1.878	
	14	9.011	2.407	9.327	0.093	0.194	
	15	22.327	7.578	23.578	0.320	0.630	
	16	11.709	3.376	12.186	0.122	0.256	
13		37.646	13.095	39.838			
	12	37.646	13.095	39.859	0.000	1.878	
14		-6.200	-1.600	6.403			
	12	-8.918	-2.213	9.188	0.093	0.194	
	15	2.718	0.613	2.786	0.015	0.014	
		-8.200	-2.500	8.573			

15	12	-22.007	-6.948	23.078	0.320	0.630	
	14	-2.702	-0.599	2.768	0.015	0.014	
	18	8.393	1.471	8.521	0.071	0.144	
	23	8.116	3.576	8.869	0.071	0.144	
16		-3.500	-1.800	3.936			
	12	-11.587	-3.120	11.999	0.122	0.256	
	17	8.087	1.320	8.194	0.050	0.116	
17		-9.000	-5.800	10.707			
	16	-8.037	-1.204	8.127	0.050	0.116	
	10	-0.963	-4.596	4.696	0.007	0.017	
18		-3.200	-0.900	3.324			
	15	-8.323	-1.327	8.428	0.071	0.144	
	19	5.123	0.427	5.140	0.016	0.032	
19		-9.500	-3.400	10.090			
	18	-5.107	-0.395	5.122	0.016	0.032	
	20	-4.393	-3.005	5.322	0.009	0.018	
20		-2.200	-0.700	2.309			
	19	4.402	3.023	5.340	0.009	0.018	
	10	-6.602	-3.723	7.579	0.050	0.111	
21		-17.500	-11.200	20.777			
	10	-15.731	-10.725	19.040	0.117	0.252	
	22	-1.769	-0.475	1.831	0.000	0.001	

22		0.000	0.000	0.000			
	10	-7.608	-5.114	9.167	0.057	0.117	
	21	1.769	0.476	1.832	0.000	0.001	
	24	5.839	4.639	7.457	0.059	0.092	
23		-3.200	-1.600	3.578			
	15	-8.045	-3.432	8.746	0.071	0.144	
	24	4.845	1.832	5.180	0.033	0.068	
24		-8.700	-6.699	10.981			
	22	-5.779	-4.546	7.353	0.059	0.092	
	23	-4.812	-1.764	5.125	0.033	0.068	
	25	1.891	-0.389	1.931	0.007	0.012	
25		0.000	0.000	0.000			
	24	-1.885	0.401	1.927	0.007	0.012	
	26	3.544	2.366	4.261	0.044	0.066	
	27	-1.660	-2.767	3.227	0.011		
26		-3.500	-2.300	4.188			
	25	-3.500	-2.300	4.188	0.044	0.066	
27		0.000	0.000	0.000			
	25	1.671	2.788	3.250	0.011	0.021	
	28	-14.949	-6.113	16.151	0.000	0.974	
	29	6.188	1.666	6.409	0.085	0.161	

	30	7.090	1.659	7.282	0.160	0.301	
28		0.000	0.000	0.000			
	27	14.949	7.087	16.544	0.000	0.974	0.968
	8	-0.321	-3.201	3.217	0.001	-4.468	
	6	-14.628	-3.887	15.136	0.036	-13.431	
29		-2.400	-0.900	2.563			
	27	-6.103	-1.505	6.286	0.085	0.161	
	30	3.703	0.605	3.752	0.033	0.063	
30		-10.600	-1.900	10.769			
	27	-6.930	-1.358	7.062	0.160	0.301	
	29	-3.670	-0.542	3.710	0.033	0.063	
Total Losses					9.18 MW		

6.2 PARAMETERS USED IN HYBRID GA:
The following parameters are used in Hybrid GA

Table 6.2 Parameters used in Hybrid GA

Population size	20
Cross over probabilty	0.8
Mutation probability	0.8
Initial Temperature	10000
Alpha (α)	0.5
Beta (β)	0.5

6.3 COMPARISON OF RESULTS

The comparison of the obtained results with those reported in literature [2] is represented below**:**

Table 6.3 Comparison of results

	ED+LF [2]	Alsac –Stott [2]	OPFSA [2]	**Hybrid GA (Results obtained from Proposed Algorithm)**
Unit 1(MW)	192.65	138.56	173.15	**193.4018**
Unit 2(MW)	48.92	57.56	48.54	**21.9355**
Unit 5(MW)	19.26	24.56	19.23	**24.7165**
Unit 8(MW)	10.58	35.00	12.81	**10.8553**
Unit 11(MW)	10.79	17.93	11.64	**18.4262**
Unit 13(MW)	12.24	16.91	13.21	**37.6461**
Overall cost(£/hr)	805.45	813.74	799.45	**799.96**
Losses(MW)	11.04	7.13	9.20	**9.18**
Violating limits	0	0	0	**0**

6.4 FUTURE SCOPE

The primary objective of the optimal power flow is to minimize the transmission losses and fuel cost within the limits given so that the security limits of the network does not get affected. In this thesis Hybrid Genetic algorithm is used to solve the optimal power flow for IEEE 30 bus system. Hybrid Genetic algorithm is used so as to overcome the limitations of Genetic algorithm

The suggestion for future work in this area is to explore and investigate the role of shunt Capacitor or series capacitor for the reactive power compensation or the role of FACTS devices in OPF solution. The model of the FACTS devices may be included in the problem, and solution be obtained using hybrid GA.

APPENDIX A

Table A-1: Bus Data

Bus no	Bus code	Voltage Mag	Angle Degree	Load (MW)	Load (Mvar)	Gen (MW)	Gen (Mvar)	Gen (Q_{min})	Gen (Q_{max})	Injected (Mvar)
1	1	1.05	0.0	0.0	0.0	50	-20	0	150	0
2	2	1.033	0.0	21.70	12.7	20	-30	0	60	0
3	0	1.0	0.0	2.4	1.2	0	0.0	0	0	0
4	0	1.0	0.0	7.6	1.6	0	0.0	0	0	0
5	2	1.0058	0.0	94.2	19.0	15	-15	0	60	0
6	0	1.0	0.0	0.0	0.0	0	0.0	0	0	0
7	0	1.0	0.0	22.0	10.9	0	0.0	0	0	0
8	2	1.023	0.0	30.0	30.0	10	-15	0	50	0
9	0	1.0	0.0	0.0	0.0	0	0.0	0	0	0
10	0	1.0	0.0	5.8	2.0	0	0.0	0	0	19
11	2	1.0913	0.0	0.0	0.0	10	-10	0	-40	0
12	0	1.0	0.0	11.2	7.5	0	0	0	0	0
13	2	1.0883	0.0	0	0.0	12	-15	0	45	0
14	0	1.0	0.0	6.2	1.6	0	0	0	0	0
15	0	1.0	0.0	8.2	2.5	0	0	0	0	0
16	0	1.0	0.0	3.5	1.8	0	0	0	0	0
17	0	1.0	0.0	9.0	5.8	0	0	0	0	0
18	0	1.0	0.0	3.2	0.9	0	0	0	0	0
19	0	1.0	0.0	9.5	3.4	0	0	0	0	0
20	0	1.0	0.0	2.2	0.7	0	0	0	0	0
21	0	1.0	0.0	17.5	11.2	0	0	0	0	0
22	0	1.0	0.0	0	0.0	0	0	0	0	0
23	0	1.0	0.0	3.2	1.6	0	0	0	0	0
24	0	1.0	0.0	8.7	6.7	0	0	0	0	0
25	0	1.0	0.0	0	0.0	0	0	0	0	0

26	0	1.0	0.0	3.5	2.3	0	0	0	0	0
27	0	1.0	0.0	0	0.0	0	0	0	0	0
28	0	1.0	0.0	0	0.0	0	0	0	0	0
29	0	1.0	0.0	2.4	0.9	0	0	0	0	0
30	0	1.0	0.0	10.6	1.9	0	0	0	0	0

TABLE A-2 LINE DATA

Bus from	Bus to	R p.u	X p.u	½ B p.u	Tap at bus
1	2	0.0192	0.0575	0.02640	1
1	3	0.0452	0.1852	0.02040	1
2	4	0.0570	0.1737	0.01840	1
3	4	0.0132	0.0379	0.00420	1
2	5	0.0472	0.1983	0.02090	1
2	6	0.0581	0.1763	0.01870	1
4	6	0.0119	0.0414	0.00450	1
5	7	0.0460	0.1160	0.01020	1
6	7	0.0267	0.0820	0.00850	1
6	8	0.0120	0.0420	0.00450	1
6	9	0.0000	0.2080	0.00000	0.978
6	10	0.0000	0.5560	0.00000	0.969
9	11	0.0000	0.2080	0.00000	1
9	10	0.0000	0.1100	0.00000	1
4	12	0.0000	0.2560	0.00000	0.932
12	13	0.0000	0.1400	0.00000	1
12	14	0.1231	0.2599	0.00000	1
12	15	0.0662	0.1304	0.00000	1
12	16	0.0945	0.1987	0.00000	1
14	15	0.2210	0.1997	0.00000	1
16	17	0.0824	0.1923	0.00000	1
15	18	0.1073	0.2185	0.00000	1

18	19	0.0639	0.2192	0.00000	1
19	20	0.0340	0.0680	0.00000	1
10	20	0.0936	0.2090	0.00000	1
10	17	0.0324	0.0845	0.00000	1
10	21	0.0348	0.0749	0.00000	1
10	22	0.0727	0.1499	0.00000	1
21	22	0.0116	0.0236	0.00000	1
15	23	0.1000	0.2020	0.00000	1
22	24	0.1150	0.1790	0.00000	1
23	24	0.1320	0.2700	0.00000	1
24	25	0.1885	0.3292	0.00000	1
25	26	0.2544	0.3800	0.00000	1
25	27	0.1093	0.2087	0.00000	1
28	27	0.000	0.3960	0.00000	01.968
27	29	0.2198	o.4153	0.00000	1
27	30	0.3202	0.6027	0.00000	1
29	30	0.2399	0.4533	0.00000	1
8	28	0.0636	0.2000	0.0214	1
6	28	0.0169	0.0599	0.065	1

TABLE A-3 COST COFFICIENTS

A	b	c
0.00375	2	0
0.01750	1.75	0
0.06250	1.00	0
0.00834	3.25	0
0.02500	3	0
0.02500	3	0

TABLE A-4 GENERATOR LIMITS

MW(minimum)	MW(maximum)	Mvar(minimum)	Mvar(maximum)
50	200	-30	150
20	80	-30	60
15	50	-15	60
10	35	-15	50
10	30	-10	40
12	40	-15	45

REFERENCES

[1] J.A momoh , R.J Koessler, M.S Bond, B.Stott, D.Sun, A.Papalexopoulos, P.Ristanovic,“Challenges to optimal power flow”, IEEE Power Engineering Society Vol 12 no.1, pp.445-455 February 1997.

[2] C.A Rao – Sepulveda, B.J. Pavez-Lazo, “ A solution to the optimal power flow using simulated annealing”, IEEE Proto Power Tech Conference, Vol 2, pp.10-13 september-2001

[3] Eliphas Shikongo,“ Optimal power flow analysis based on evolutionary algorithm”, Thesis submitted at university of Cape town,23 October 2006.

[4] Allen J.Wood, Bruce F. Wollenberg, “Power generation operation and control”, NewYork, NY: John Wiley & Sons, Inc., 2^{nd} edition, 1996.

[5] K. L. Lo and Z. J. Meng, “Newton-like method for line outage simulation”, IEE proc. Gener. Transm. Distrib , Vol. 151, no. 2, pp. 225-231, March 2004.

[6] X. Tong and M. Lin, “Semismooth Newton type algorithms for solving optimal power flow problems”, in Proc.of IEEE/PES Transmission and Distribution Conference, Dalian, China, pp.1-7, 2005.

[7] J.A Momoh , “ A generalized quadratic based model for optimal power flow”, in proc. of IEEE ,pp. 261-267,1989.

[8] Jason Yuryevich, Kit Po Wong, “Evolutionary programming based optimal power flow algorithm”, IEEE Transactions on Power Systems, Vol. 14, No. 4, Nov. 1999.

[9] M.Tarafdar Haque & A.M Kashtiban , “ Application of neural network in power system: A Review”, Word Academy of science, Engineering and technology, pp 53-57,June 2005

[10] M.K Pakhirs, “A Hybrid Genetic Algorithm using Probabilistic Selection”, IE(I) Journal,Vol.84, May 2003.

[11] A. Rangel-Merino, J. L. López-Bonilla, R. Linares y Miranda , “ Optimization Method based on Genetic Algorithm,” Apeiron , Vol. 12, No. 4, October 2005.

[12] Tarek Bouktir, Linda Slimani and M. Belkacemi, “ Genetic Algorithm for Solving the Optimal Power Flow Problem,”Lenardo journal of sciences, Issue 4, pp.44-58 June 2004.

[13] H John Holland, “Adaptation in Natural and Artificial Systems”, University of Michigan Press, Ann Arbor, 1975.

[14] Kalyanmoy Deb,“Multi Objective Optimization using Evolutionary Algorithm,”John Wiley & Sons, Ltd, 2001.

[15] S.Rajasekaran, G.A Vijayalakshmi pai, “Neural Networks, Fuzzy Logic, and Genetic Algorithm Synthesis and Application”, Prentice –Hall of India private Ltd. New delhi, 2008.

[16] www.wikipedia.com

[17] http://books.google.co.in/books deceptive fitness function.

[18] http://www.talkorigins.org/faqs/genalg/genalg.html#limitations

[19] Alex Rogers, Adam Prugel –Bennett, “Genetic drift in Genetic Algorithm Selection Schemes”, IEEE Transactions on Evolutionary Computation, Vol 3, no.4, November 1999.

[20] Tomoyuki HIROYASU, Mitsunori MIKI, Maki OGURA, “Parallel Simulated Annealing Using Genetic Crossover”, Doshisha University, Japan.

[21] Liu Juan, Cai Zi-Xang, Liu Jian-qin, “ A novel genetic algorithm preventing premature convergence by Chaos operator”, J.Cent.South University ,Vol7,no 4, June 2000.

[22] S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi, “Optimization by Simulated Annealing”, *Science*, New Series, Vol. 220, No. 4598. pp. 671-680, May 13, 1983.

[23] Masaya Yoshikawa, Hironori Yamauchi, and Hidekazu Terai, “ Hybrid Architecture of Genetic Algorithm and Simulated Annealing”, Advance on line publication, 20 Aug 2008

[24] Mandira Chakraborty, Uday K. Chakraborty, “Applying genetic algorithm and Simulated annealing to a Combinatorial Optimization problem”, Internatioanl Conference on Information, Communication and signal Processing (ICICS), Singapore ,9-12 September, 1997