International journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available onlne http://www.ijoer.in

Vol.1., Issue.2., 2013

RESEARCH ARTICLE

ISSN: 2321-7758

POWER SYSTEMS USING SOFT COMPUTING FOR ECONOMIC PROBLEMS

S.SATHISH KUMAR[1], P.RAVIKUMAR[2], K.KRISHNA KUMAR[3]
[1-3]Assistant Professor,
Department of Electrical and Electronics Engineering, Jansons institute of technology
Coimbatore, Tamilnadu, India

Article Received: 30/08/2013 | Article Revised on: 05/09/2013 | Article Accepted on:07/09/2013

S.SATHISH KUMAR
Author for Correspondence
Email: sat_meae2k5@rediffmail.com

ABSTRACT: This research paper involved is an optimization problem where objective function is highly non linear, non-convex non-differentiable and may have multiple local minima. Therefore classical optimization methods may not converge or get trapped to any local minima. This research paper presents a comparative study of three different evolutionary algorithms i.e. genetic algorithm, ant colony optimization and particle swarm optimization for solving the economic problem. All the methods are tested on IEEE 30 bus test system.

Keywords -Genetic algorithm, ant colony optimization, and particle swarm optimization, economic problems, evolutionary algorithm

INTRODUCTION

Economic problem in Power System deals with the determination of optimum generation schedule of available generators so that total cost of generation is minimized within the system constraints. Several classical optimization techniques such as lambda iteration method, gradient method. Newton's method, linear programming, Interior point method and dynamic programming have been used to solve the basic economic problem. Lambda iteration method has the difficulty of adjusting lambda for complex cost functions Gradient methods suffer from the problem of convergence in the presence of inequality constraints. Newton's method very much sensitive to the selection of initial conditions. Linear programming approach provides optimal results in less computational time but results are not accurate due to linearization of the problem. Interior point method is faster than linear programming but it may provide infeasible solution if the step size is not chosen properly. Dynamic programming suffers from curse of dimensionality.

Most of the classical optimization techniques need derivative information of the objective function to determine the search direction. But actual fuel cost functions are non-linear, non-convex and non-differentiable because of ramp rate limits, prohibited operating zones, valve point effects and multi-fuel options. Recently some heuristic techniques such as genetic algorithm, genetic algorithm combined with simulated annealing, evolutionary programming, improved tabu search, ant swarm optimization and particle swarm optimization have been used to solve the complex non-linear optimization problem.

In this research paper Economic problem has been solved using three different evolutionary algorithms i.e. genetic algorithm (GA),), ant colony optimization, ACO) and particle swarm optimization (PSO). Performance of each algorithm for solving the Economic problem has been investigated and simulation results are presented in terms of accuracy, reliability and execution time.

The research paper is organized as follows. Section II formulates the Economic problem, Section III presents the brief overview of different evolutionary algorithms, Section IVdescribes the constrain handling method, and Section V concludes.

II.PROBLEM FORMULATION

The objective of solving the Economic problem is to minimize the total generation cost of a power system while satisfying various equality and inequality constraints.

Typically generator cost functions are approximately modeled by quadratic functions. But, actual fuel cost function of any large turbo generator may be much more complicated due to valve point loading effect. The Economic problem considering the valve point loading effect can be expressed as:

$$Minimize F = \sum_{i=1}^{NG} F_{gi}$$

$$= \sum_{i=1}^{NG} (p_i + q_i P_{gi} + r_i p_{gi}^2 + \left| e_i \times \sin\left(f_i \times (P_{gi} - P_{min})\right)\right| \quad (1)$$

Where,

F - Total generation cost;

F_{gi} - generation cost of generator i;

NG -number of generators;

p_{gi} -generation of generator i;

p_{min} -minimum generation of generator i;

p_i,q_i,r_i -cost coefficients of generator i;

e_i,f_i -coefficients of generator i reflecting valve point loading effect.

Subjected to the following constraints:

$$P_{gi} - p_{di} - p_{loss} = 0 \quad (2)$$

$$Q_i - Q_{di} - Q_{loss} = 0 \quad (3)$$

$$P_{min} \leq P_{gi} \leq P_{max} \quad (4)$$

$$Q_{min} \leq Q_{gi} \leq Q_{max} \quad (5)$$

Where,

P_{gi},Q_{gi}-real and reactive power generation at bus i;

P_{di},Q_{di}-real and reactive powerdemand at bus i;

P_{loss},Q_{loss} -total real and reactive power loss;

P_{min},P_{max} -minimum and maximum active Power generation limits ofgenerator i;

Q_{min},Q_{max}-minimum and maximum generation limits of generator I;

EVALUTION ALGORITHM (GA)

This section gives brief description of the evolutionary algorithms used in this research paper as follows:

A. Genetic Algorithm (GA)

In last few decades, GA has been treated as benchmark for various optimization problems. GA consists of four steps i.e. representation, initialization, selection and reproduction with crossover and mutation. Depending on the type of representation genetic algorithms can be broadly classified into two groups (i)Binary coded genetic algorithm (BCGA) and (ii)Real coded genetic algorithm (RCGA).

1) Binary Coded Genetic Algorithm (BCGA): In BCGA, all the real world problems (phenotype space) are encoded to binary representation (genotype). Each element of a solution vector is represented by a binary string $[z_i \in [x,y]] = [a_i^1, a_i^2, \dots, a_i^L] \in [0,1]^L$

The length of the binary string (chromosome) *L* depends on how much accuracy is required. For BCGA, there are several selection methods to form the parent pool such as proportionate reproduction, tournament selection, rank selection, genitor selection, etc. In this research paper binary tournament selection scheme is used because of its less time complexity.

Crossover is a method for sharing information between chromosomes. It combines the features of parent chromosomes to form offspring. Commonly used crossover techniques are one point crossover, two point crossovers, n, point crossover uniform crossover, heuristic crossover, etc. In this paper uniform crossover is used to avoid positional bias. Crossover operation makes a big jump to an area somewhere in between the two parent (explorative) whereas mutation creates random small diversions near the parent (exploitative). Typically mutation probability is very low.

2) Real Coded Genetic Algorithm (RCGA): BCGA has difficulties of binary representation when dealing with continuous search space with large dimensions and improved numerical precision. RCGA does not have such difficulties. In RCGA genes are represented directly as real numbers and a chromosome is a vector of floating point numbers. As RCGA deals directly with real numbers, there is no need of phenotype to genotype conversion and vice versa. Any decision vector x is represented as follows: $\bar{x} = [x_1, x_2, \dots, x_n]$, where n is the number of parameters to be optimized.

There are different types of crossover techniques for RCGA such as at, simple, arithmetic, blend, linear, discrete, logical FCB (connectives Based Crossover) etc. Among different crossover techniques blend (with a=0.5),linear and logical FCB crossover techniques outperform the others . In this research paper blend crossover (with a=1.5) has been used to maintain a balance between exploitation.

There are different types of mutation techniques for RCGA such as nonuniform, real number creep, continuous modal mutation, discrete modal mutation. Among them non-uniform crossover is very effective for RCGA. Therefore, In this study non-uniform crossover is chosen for RCGA.

B. Particle Swarm Optimization (PSO)

In 1995, Kennedy and Eberhart first introduced the PSO method motivated by social behavior of organisms such as fish schooling and bird flocking. PSO is a population based search technique. Each individual potential solution in PSO is called particle. Each particle in a swarm fly around in a multidimensional search space based on its own experience and experience of neighbouring particles. Let define the search space S in *n*-dimension and the swarm consists of N particals.Let at instant t, particle i has its position defined by $x_t^i = \{x_1^i, x_2^i, \dots, x_n^i\}$ and velocity defined by $V_t^i = \{v_1^i, v_2^i, \dots, v_n^i\}$ in variable space S. Velocity and position of each particle in the next generation (time step) can be calculated as:

$$V_{t+1}^i = w \times V_t^i + c_1 \times rand() \times (P_t^i - X_t^i) + c_2 \times Rand() \times (P_t^g - X_t^i) \quad (6)$$

$$X_{t+1}^i = X_t^i + V_{t+1}^i \forall i = 1, \dots, N \quad (7)$$

Where

N -number of particle in the swarm;

W -inertia weight;

c_1, c_2 -acceleration constant ;rand (), - uniform random value in the Rand() range [0,1];

P_t^g -global best at generation t;

P_t^i -best position that particle i could find so far.

Performance of PSO depends on selection of inertia weight(w) maximum velocity v_{max} and acceleration constant (c_1, c_2). The effect of these parameters is illustrated as follows: Inertia weight(w): Suitable selection of weight factor w helps in quick convergence. A large weight factor facilitates global exploration (i.e. searching of new area) while small weight factor facilitate local exploration. Therefore, it is wiser to choose large weight factor for initial iterations and gradually smaller weight factor for successive iterations. In standard PSO linearly decreasing inertia weight w is set as 0.9 at beginning and 0.4 at the end.

Maximum velocity (v_{max}): With no restriction on the maximum velocity of the particles, velocity may become infinitely large. If v_{max} is very low particle may not explore sufficiently and if v_{max} is very high it may oscillate about optimal solu- tion. Therefore, velocity clamping effect has been introduced to avoid the phenomenon of “swarm explosion”. In general maximum velocity is set as 10-20% of dynamic range of each variable. Velocity can be controlled within a band as:

$$v_{max} = v_{ini} - \frac{v_{ini} - v f_{an}}{iter_{max}} \times iter \quad (8)$$

Where v_{ini} is initial velocity,vf_{an} is final velocity $iter$ is iteration number and $iter_{max}$ is number of maximum iterations.

Acceleration constants (c_1, c_2): Acceleration constant called cognitive parameter pulls each particle towards local best position whereas constant c_2 called social parameter pulls the particle towards global best position. Usually the values of c_1 and c_2 are chosen between 0 to 4.

C. Ant Colony Optimization (ACO)

ACO was invented by Marco Dorigo and colleages inspired by the foraging behavior of ant colony. While moving, each ant lays certain amount of pheromone on the path. Ants use the pheromone trails to communicate information among the individuals and based on that each ant decides the shortest path to follow.

ACO technique has been successfully used for difficult discrete combinatorial optimization problems such as Traveling Salesman Problem, Sequential Ordering, and Routing in communication problems, etc. In recent years a large number of literatures have been published where ACO algorithm has been successfully used for continuous optimization problems.

In this research paper, ACO algorithm proposed in has been implemented for the solution of ED problem. The algorithm consists of four stages: Solution construction, Pheromone update, Local Search and Pheromone Re-initialization

1) Solution Construction: In this method, initial position of each ant i.e. initial solution vectors are generated randomly in the feasible search region. In each iteration, artificial ant construct the solution by generating a random number for each variable using the normal distribution $N(\mu_i, \sigma_i^2)$.Mean (μ_i) and standard and deviation (σ_i^2) for each variable i changes with iteration number based on the experience of the colony.

This is synonymous to pheromone update in basic ant colony optimization.

Let $x = [x_1 \dots , x_n]$ is a solution constructed by any ant using normal distribution $N(\mu_i, \sigma_i^2), i \in \{1, \dots, n\}$ associated with each variable xi here n is the number of parameters to be optimized. After construction of each solution, upper and lower bound for each parameter is checked and the solution is modified if it goes out of search space:

$$x_i = \begin{cases} a_i x_i < a_i \\ x_i a_i \le x_i \le b_i \\ b_i x_i > b_i \end{cases} \quad (8)$$

Where b_i and a_i are upper and lower bound of variable x_i respectively

2) ***Pheromone update****:* Pheromone update procedure com posed of pheromone evaporation and pheromone intensification. Pheromone initialization is done as follows:

$$\mu_i(0) = a_i + rand\,(i)\,(b_i - a_i)$$

$$\sigma_i(0) = (b_i - a_i)/2 \quad (9)$$

After construction of solutions, pheromone evaporation phase is performed as follows:

$$\mu_i(t) = (1 - \rho_1)\mu_i(t-1)$$

$$\sigma_i(t) = (1 - \rho_1)\sigma_i(t-1) \quad (10)$$

Where t is iteration number and $\rho_1 \in [0,1]$ is the evaporation parameter, a uniform random number between 0 and 1.

The aim of pheromone intensification is to increase the pheromone value associated with promising solutions. This is done as follows:

$$\mu_i(t) = \mu_i(t) + \rho_2 x^{gb}$$

$$\sigma_i(t) = \sigma_i(t) + \rho_2 |x^{gb} - \mu_i\,(t-1)| \quad (11)$$

Where $\rho_2 \in [0,1]$ is intensification parameter, a uniform random number between 0 and 1 and

X^{gb} is the global best solution found in last (t-1) iteration.

3) ***Local Search****: (*Local search in the vicinity of current solution may improve the solution constructed. Local search is usually made before updating the pheromone. Local search technique proposed in has been used in this research paper.

SIMULATION

Simulation using matlab was performed and the results were carried out for the minimum and maximum transmission lines capacity were in under process.

CONCLUSIONS

In this research paper a comparative study of different evolutionary techniques to solve the power system economic problem is investigated. From the simulation results presented in this paper it is clear that solution. From all these findings, it can be not as good as PSO outperforms the others for the chosen set of parameters for solving the economic problem

REFERENCES

[1] A.I.selvakumar and K.thanushkodi,A New Particle Swarm Optimization Solution To Nonconvex Economic Problems,IEEE Trans.Power System,vol.22,no.1,pp.42-51,Feb.2007.

[2] J.Cai,X.MaL,Li,Y.Yangh,H.Peng,andX.Wang,Chaotic Ant Swarmoptimization To Economic

Problem,Electric Power System Research,vol.77,no.10,pp.1373-1380,Aug.2007.

[3] M.R.Alrashidi and M.E.El-Hawary,A Survey Of Particle Swarm Optimization Applications In Power System Operation,Electric Power Components And Systems, vol.34, no.12, pp.1349-1357, Dec.2006.

[4] J.KENNEDY,R.C.Ebhert,andY.Shi,SwarmIntelligence. MORGANKAUFMANN,San Franscisco,2001.

[5] M.Kong and P.Tian,Ant Colony Optimization And Swarm Intelligence.Springer-VERLAG, 2006, A Direct Application Of Ant Colony Optimization To Function Optimization Problem In Continuous Domain.pp.324-331.

[6] M.Dorigo,M.Birattari,andT.Stutzle,Ant Colony Optimization Artificial Ants As A Computational Intelligence Technique,IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE,pp. 28-39,2006.

[7] J.Hazara and A.K.Sinha, Congestion Management Using Multi Objective Particle Swarm Optimization, IEEETrans. power systems,vol.22,no.4,pp.1726-1734,Nov.2007.