International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 2, Issue 7, July 2013 ISSN 2319 - 4847

Particle Swarm Optimization and Genetic Algorithm based Optimal Power Flow Solutions

Bhupender Sharma[1], Shivani Sehgal[2], Ajay Nain[3]

[1,]*Department of Electrical & Electronics Engg., DVIET,*
Karnal, Affiliated to Kurukshetra university Haryana, India

[2,]*Assistant Professor, Department of Electrical & Electronics Engg., DVIET,*
Karnal, Affiliated to Kurukshetra university Haryana, India

[3,]*Assistant Professor, Department of Electrical Engg,RPIIT Technical Campus*
Bastara, Affiliated to Kurukshetra university Haryana, India

Abstract

This paper present some pointed out factors or parameters related to optimal power flow solutions by using the Particle swarm optimization and Genetic algorithm to observe some refined status and supervise the practicality for a particular electrical network. The work presented in this paper utilizes an optimal power flow problem. The OPF utilizes all control variables to help to minimize the cost of the power system operation. It also yields valuable economic information and insight into the power system. In this way, the optimal power flow problem very adeptly addresses the control and economic problems. After formulating the OPF problems, results are obtained by using genetic algorithm and particle swarm optimization, programming work is carried out in Mat lab. A case study on an IEEE-30 bus system expresses some sound idea in a very positive result oriented manner directed towards the applicability of the proposed approaches in the practical electrical network system. A comparison in both the proposed approaches is made and some important features are extracted out as: for the same electrical network configurations both the methods are proved almost better but not so much comparatively competitive for the optimal power flow. Results show that the algorithm is well competent for optimal power flow under practical constraints and price based conditions.

Keywords— Particle Swarm optimization(PSO), Genetic Algorithm(GA), Optimal Power Flow(OPF)

1. INTRODUCTION

The OPF Problem has been discussed since its introduction by Carpentier in 1962. As the OPF is a very large, non-linear mathematical programming problem, it has taken decades to develop efficient algorithm for its solution. Many different mathematical techniques have been employed for its solution. The problem can be stated as:

"OPF has been applied to regulate generator active power outputs and voltages, shunt capacitors/reactors, transformer tap settings and other controllable variables to minimize the fuel cost, network active power loss, while keeping the load bus voltages, generators reactive power outputs, network power flows and all other state variables in the power system in their operational and secure limits".

The optimal power flow has been frequently solved using classical optimization methods. The OPF has been usually considered as the minimization of an objective function representing the generation cost and/or the transmission loss. The constraints involved are the physical laws governing the power generation-transmission systems and the operating limitations of the equipment.

Effective optimal power flow is limited by (i) the high dimensionality of power systems and (ii) the incomplete domain dependent knowledge of power system engineers. The first limitation is addressed by numerical optimization procedures based on successive linearization using the first and the second derivatives of objective functions and their constraints as the search directions or by linear programming solutions to imprecise models. The advantages of such methods are in their mathematical underpinnings, but disadvantages exist also in the sensitivity to problem formulation, algorithm selection and usually converge to a local minimum. The second limitation, incomplete domain knowledge, precludes also the reliable use of expert systems where rule completeness is not possible. A first comprehensive survey regarding optimal power dispatch was given by H.H.Happ and subsequently an IEEE working group presented bibliography survey of major economic-security functions in 1981. Thereafter in 1985, Carpentair presented a survey and classified the Optimal Power Flow (OPF) algorithms based on their solution methodology. In 1990, Chowdhury did a survey on economic dispatch methods .In 1999, J.A.Momoh et al. presented a review of some selected OPF techniques. Dommel and W.F Tinney (1968) gave realistic method for solving the power flow programs with control variables such as real power, reactive power and transformer ratios automatically adjusted to minimize instantaneous costs or losses. P.H.Chen et al (1995) proposed a new genetic algorithm for solving the Economic Dispatch (ED) problem in large-scale systems. A new encoding method is developed in which the chromosome contains only an encoding of the normalized system incremental cost. Russell Eberhart (1995) presents the optimization of nonlinear functions using particle swarm methodology is described. Implementations of two paradigms are discussed and compared, including a recently developed locally oriented paradigm. M.A.Abido et al.(2000) proposed an efficient and reliable evolutionary based approach to solve the optimal power flow problem. The proposed approach employs particle swarm optimization algorithm for optimal settings of OPF problem control variables. Anastasios G.Bakirtzis (2002) proposed an enhanced genetic algorithm (EGA) for the solution of the optimal power flow (OPF) with both continuous and discrete control variables. M.S.Osman (2003) presents the solution of the OPF using genetic algorithm technique and proposes a new methodology for solving OPF, the main goal is to verify the viability of using genetic algorithm to solve the OPF problem simultaneously composed by the load flow and the economic dispatch problem. Six buses system are used to highlight the goodness of this solution technique.

The work presented in this paper utilizes an optimal power flow problem. The OPF utilizes all control variables to help to minimize the cost of the power system operation. It also yields valuable economic information and insight into the power system. In this way, the optimal power flow problem very adeptly addresses the control and economic problems. After formulating the OPF problems, results are obtained by using genetic algorithm and particle swarm optimization, programming work is carried out in Matlab. A case study on an IEEE-30 bus system expresses some sound idea in a very positive result oriented manner directed towards the applicability of the proposed approaches in the practical electrical network system.

2. PROBLEM FORMULATION

The most commonly used objective in the Optimal Power Flow problem formulation is the minimization of total cost of real power generation. The individual cost of each generating unit is assumed to be functions, only of active power generation and is represented by quadratic curves of second order. The objective function of entire power system can then be written as the sum of the quadratic cost model at each generator.

$F_i = a_i P_{gi}^2 + b_i P_{gi} + c_i$

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

Where n is no. of generators including the slack bus , P_{gi} is the generated active power at bus i. a_i, b_i, c_i are the unit costs curve for ith generator. The cost is optimized with the following constraints.

A. Types of Equality Constraints

While minimizing the cost function, it is necessary to make sure that the generation still supplies the load demand plus losses in transmission lines. Usually the power flow equations are used as equality constraints: The power flow equation of the network

$g(v,\varphi) = 0$

where $g(v,\varphi) = P_i(V, \varphi) - P_{inet}$

$Q_i(V, \varphi) - Q_{inet}$

$P_m(V, \varphi) - P_{mnet}$

Where P_i and Q_i are calculated real and reactive power for PQ bus i respectively.P_{inet} and Q_{inet} are specified real and reactive

power for PQ buses i respectively.

P_m and P_{mnet} are calculated and specified power for PV bus m respectively.

V and,φ are voltage magnitude and phase angle at different buses respectively.

B. Types of Inequality constraints

The inequality constraints on problems represent the system operating constraints. The inequality constraints of the optimal power flow reflect the limits on physical device in the power systems as well as the limits created to ensure system security. The usual types of these constraints are upper bus voltage limits at generations at load buses, lower bus voltage limits at some generators, limits on tap settings and maximum line loading limits.

Generation constraints: Generator voltages, Real power and reactive power outputs are restricted by their limits as follows:

$V_{imin} \le V_i \le V_{imax}$

$P_{gimin} \le P_{gi} \le P_{gimax}$

$Q_{gimin} \le Q_{gi} \le Q_{gimax}$

i=1, 2 ...nG

where nG is the number of generators.

V_{imin} and V_{imax} are respectively minimum and maximum voltage at bus i.

P_{gimin} and $P_{gimax}$are minimum and maximum values of real power generation allowed at generator bus i.

Qgimin and Qgimax are minimum and maximum values of reactive power generation allowed at PV bus i.
Transformer constraints: Transformer tap settings are bounded as follows:
Timin ≤ Ti ≤ Timax , where i=1,2,3.......nT
where nT is the number of transformers.
Phase angle constraints:
φ imin. ≤ φ i ≤ φ imax
Where φ imin. and φ imax are respectively minimum and maximum phase angle at bus i.

3. PARTICLE SWARM OPTIMIZATION

A. Overview

Particle Swarm Optimization Swarm Intelligence (SI) is an innovative distributed intelligent paradigm for solving optimization problems that originally took its inspiration from the biological examples by swarming, flocking and herding phenomena in vertebrates. Particle Swarm Optimization (PSO) incorporates swarming behaviors observed in flocks of birds, schools of fish, or swarms of bees, and even human social behavior, from which the idea is emerged. PSO is a population-based optimization tool, which could be implemented and applied easily to solve various function optimization problems. As an algorithm, the main strength of PSO is its fast convergence, which compares favorably with many global optimization algorithms like Genetic Algorithms (GA) Simulated Annealing (SA) and other global optimization algorithms. For applying PSO successfully, one of the key issues is finding how to map the problem solution into the PSO particle, which directly affects its feasibility and performance Similar to evolutionary algorithm, the PSO technique conducts searches using a population of particles, corresponding to individuals. Each particle represents a candidate solution to the optimal power flow problem. In a PSO system, particles change their positions by flying a round in a multidimensional search space until a relatively unchanged position has been encountered, or until computational limitations are exceeded. In social science context, a PSO system combines a social –only model and a cognition-only model. The social-only component suggests that individuals ignore their own experience and adjust their behavior according to the successful beliefs of the individual in the neighborhood. On the other hand, the cognition-only component treats individuals as isolated beings. A particle changes its position using these models
The advantages of PSO over other traditional optimization techniques can be summarized as follows:
a) PSO is a population-based search algorithm (i.e., PSO has implicit parallelism). This property ensures PSO to be less susceptible to getting trapped on local minima.
b) PSO uses payoff (performance index or objective function) information to guide the search in the problem space. Therefore, PSO can easily deal with non-differentiable objective functions. Additionally, this property relieves PSO of assumptions and approximations, which are often required by traditional optimization models.
c) Stochastic optimization algorithm that can search a complicated and uncertain area. This makes PSO more flexible and roust than conventional methods.
d) Unlike Genetic Algorithm (GA) and other heuristic algorithms, PSO has the flexibility to control the balance between the global and local exploration of the search space. This unique feature of a PSO overcomes the premature convergence problem and enhances the search capability.
e) Unlike the traditional methods, the solution quality of the proposed approach doesn't rely on the initial population. Starting anywhere in the search space, the algorithm ensures the convergence to the optimal solution. The basic elements of the PSO techniques are briefly stated and defined as follows:

Particle X (t): It is a candidate solution represented by an m-dimensional real valued vector, where m is the number of optimized parameters. At time t, the i^{th} particle Xi(t) can be described as $X_i(t)=[x_{i,1}(t) ; x_{i,2}(t) ; \ldots\ldots;x_{i,m}(t)]$.

Population: it is a set of n particles at time t, i.e pop(t) = $[X_1(t), X_2(t),\ldots\ldots X_n(t)]^T$.

Swarm: it is an apparently disorganized population of moving particles that tend to cluster together while each particle seems to be moving in a random direction.

Inertia weight w(t): it is a control parameter that is used to control the impact of the previous velocity on the current velocity. All the control variables transformer tap positions and switch-able shunt capacitor banks are integer variables and not continuous variables. Therefore, the value of the inertia weight is considered to be 1 in this study.

Individual best X* (t): As the particle moves through the search space, it compares its fitness value at the current position to the best fitness value it has ever attained at any time up to the current time. The best position that is associated with the best fitness encountered so far is called the individual best X* (t). For each particle in the swarm, X* (t) can be determined and updated during the search.

Global best X (t):** It is the best position among all of the individual best positions achieved so far.

Stopping criteria: These are the conditions under which the search process will terminate. In this study, the search will terminate if one of following criteria is satisfied:
The number of the iterations since the last change of the best solution is greater than a pre-specified number.
The number of iterations reaches the maximum allowable number.

B. Particle swarm optimization algorithm

In a PSO algorithm, the population has n particles that represent candidate solutions. Each particle is a k-dimensional real-valued vector, where k is the number of the optimized parameters. Therefore, each optimized parameter represents a dimension of the problem space. The modified PSO technique for integer problem can be described in the following steps.

Step 1: (Initialization): Set t=0 and generate random n particles, {Xi(0), i=1,2,..n}. Each article is considered to be solution for the problem and it can be described as Xi(0)=[xi,1(0); xi,2(0); ……;xi,m(0)]. Each control variable will have a range [xmin, xmax]. Each particle in the initial population is evaluated using the objective function f. For each particle , set Xi*(0) =Xi(0) and fi* = fi ,i=1,2,3.....,n. Search for the best value of the objective function fbest .Set the particle associated with fbest as the global best,X**(0),with an objective function of f**.Set the initial value of the inertia weight w(0).In this study the objective function is the optimal power flow ,which will be calculated after running the power flow and meeting all our constraints.

Step 2: Counter Updating: update the counter t= t +1

Step 3: Velocity updating: Using the global best and individual best, the ith particle velocity in the kth dimension in this study (integer problem) is updated according to the following equation:

vi,k(t) = w(t).vi,k(t-1) + b1s1(x*i,k(t-1)-xi,k(t-1)) +b2s2(x**i,k(t-1) – xi,k (t-1)

From the previous equation i is the particle number, b1, b2 are positive constants, s1 s2 are uniformly distributed random
numbers in [0, 1] and k is the kth control variable. Then, check the velocity limits. If the velocity violated its limit, set it at its proper limit. The second term of the above equation represents the cognitive part of the PSO where the particle changes its velocity based on its own thinking and memory. The third term represents the social part of PSO where the particle changes its velocity based on the social-psychological adaptation of knowledge.

Step 4: Position updating: Based on the updated velocity, each particle changes its position according to the following equation:

Xi,k(t) = xi,k(t-1) + vi,k(t)

Step 5: Individual best updating: each particle is evaluated and updated according to the update position.

Step 6: Search for the minimum value in the individual best and its solution has ever been reached so far, and consider it to be the minimum.

Step 7: Stopping criteria: if one of the stopping criteria is satisfied, then stop otherwise go to step-2.

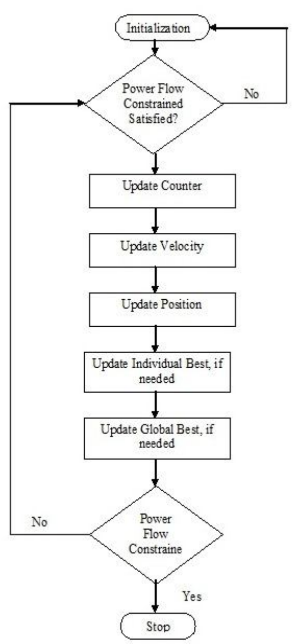

Figure-I Flow chart Of PSO

4. GENETIC ALGORITHM

A. Overview

Genetic Algorithms are general purpose optimization algorithms based on the mechanics of natural selection and genetics. Genetic Algorithms are a family of computational models inspired by evolution. These algorithms encode a potential solution to a specific problem on a simple chromosome-like data structure and apply recombination and mutation operators to these structures so as to preserve critical information. An implementation of a genetic algorithm begins with a population of (usually random) chromosomes. One then evaluates these structures and allocates reproductive opportunities in such a way that those chromosomes which represent a better solution to the target problem are given more chances to reproduce than those chromosomes which are poorer solutions. This is called survival for the fittest. The goodness of a solution is typically defined with respect to the current population. They operate on string structures (chromosomes), typically a concatenated list of binary digits representing a coding of the control parameters (phenotype) of a given problem. Chromosomes themselves are composed of genes. The real value of a control parameter, encoded in a gene, is called an allele. GAs is an attractive alternative to other optimization methods because of their robustness. There are three major differences between GAs and conventional optimization algorithms. First, GAs operates on the encoded string of the problem parameters rather than the actual parameters of the problem. Each string can be thought of as a chromosome that completely describes one candidate solution to the problem. Second, GAs uses a population of points rather than a single point in their search. This allows the GA to explore several areas of the search space simultaneously, reducing the probability of finding local optima. Third, GAs do not require any prior knowledge, space limitations, or special properties of the function to be optimized, such as smoothness, convexity, unimodality, or existence of derivatives. They only require the evaluation of the so-called fitness function (FF) to assign a quality value to every solution produced. The genetic algorithm can be viewed as two stage process. It starts with the current population. Selection is applied to the current population to create an intermediate population. Then recombination and mutation are applied to the intermediate population to create the next population. The process of going from the current population to the next population constitutes one generation in the execution construction of the intermediate population is complete and recombination can occur. This can be viewed as creating the next population from the intermediate population. Crossover is applied to randomly paired strings with a probability denoted Pc. A pair of strings is picked with probability Pc for recombination. These strings form two new strings that are inserted into the next population. After recombination, mutation operator is applied. For each bit in the population, is mutated with some low probability Pm. Typically the mutation rate is applied with less than 1% probability. In some cases mutation is interpreted as randomly generating a new bit in which case, only 50% of the time will the mutation actually change the bit value. After the process of selection, recombination and mutation, the next population can be evaluated. The process of evaluation, selection, recombination and mutation forms one generation in the execution of a genetic algorithm. Assuming an initial random population produced and evaluated, genetic evolution takes place by means of three basic genetic operators.

B. Algorithm

Step 1: Read the database for the generator data, bus data, capacitor/reactor data, transformer data and transmission line data.

Step 2: Assume suitably population size (pop_size), maximum number of generations or populations (gen_max.).

Step 3: Set valid number of population counter .pop_vn=0.

Step 4: Randomly generate the chromosomes.

Step 5: Run power flow for each set of generating patterns Pg, corresponding to a particular generation and after that 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

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

Check the MVA flows violation

$MVA_{ij}^{min}. \le MVA_{ij}^{max}$

Check reactive power limits at all generator buses

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

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

Step 8: Calculate and then store the total cost of generation corresponding to each valid generation pattern of chromosomes.

Step 9: Find and store minimum cost among all valid individual parents and corresponding generation pattern.

Step10: Check if random no.ri < cr (crossover rate) for i=1 to pop_size, select i^{th} chromosome. Apply the crossover operator to that individual.

Step11: Run power flow for each set of new generating patterns and hence determines, slack bus generation, bus voltage magnitudes and phase angle at all the buses. Also calculate power flow in each transmission line of the system.
Step 12: Check system constraints as mentioned in step 6.
Step13: If all the constraints are satisfied, the individual of the new population becomes valid otherwise it become invalid.
Step 14: Apply the mutation operator to the calculated generation patterns.
Step 15: Run power flow and check all the constraints as mentioned in step 6.
Step 16: If all the constraints are satisfied go to next step otherwise go to step 4.
Step 17: Calculate the total cost of all valid patterns.
Step 18: Find the optimum solution among all population groups.

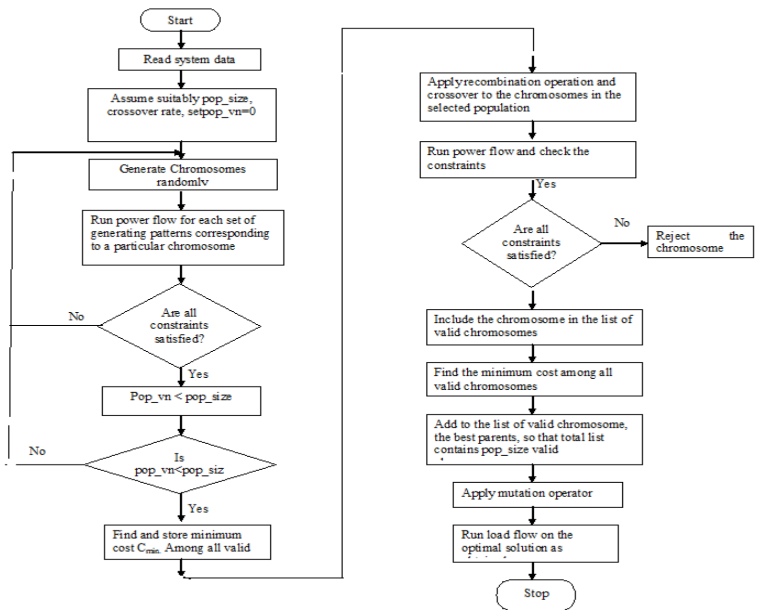

Figure-II Flow Chart of GA

5. TEST RESULTS

In this section, the proposed PSO & GA based solution of the OPF is evaluated using an IEEE 30-bus system. The GA & PSO method are implemented in MATLAB 7.5 to solve the problem of optimal power flow solutions. A comparison in both the proposed approaches is made and some important features are extracted out. Twenty runs have been performed for each case examined. The results which follow are the best solution over these 20 runs.

A. IEEE 30-Bus System

The test system is the IEEE 30-bus, 41-branch system. It has a total of 24 control variables as follows: five unit active power outputs, six generator-bus voltage magnitudes, four transformer-tap settings, and nine bus shunt admittances.
Parameters and data for the PSO Algorithm for optimal power flow
Population size =30
No. of units=6
Maximum no. of iterations=200
No. of generators=5
No. of tap positions=4
Parameters for the Genetic Algorithm
Population size -40,
Maximum no. of iterations=200
No. of units=6,
No. of generators=5
No. of tap positions=4,
String length=155
Elitism probability = 0.15
Crossover probability = 0.95
Mutation probability = 0.001

Table: I Transformer-tap settings

Transformer No.	1	2	3	4
From/To Bus No.	6-9	6-10	4-12	27-28
Final Tap Position in case of PSO	0.9	1.1	1.0	1.0625
Final Tap Position in case of GA	0.9875	1.025	1.012	0.987

Table: II Bus Shunt admittances

Shunt No.	1	2	3	4	5	6	7	8	9
Bus No.	10	12	15	17	20	21	23	24	29
Shunt value(p.u) in case of PSO	0.05	0.02	0.02	0.03	0.04	0.05	0.05	0.02	0.02
Shunt value(p.u) in case of GA	0.04	0.01	0.02	0.03	0.00	0.05	0.03	0.01	0.00

Table: III Comparison of results of PSO & GA Method for real power generation and cost

		PSO			**GA**		
Unit No.	Bus No.	Generated Voltage [p.u]	Real Power Generated [MW]	Real Power Cost [$/h]	Generated Voltage [p.u]	Real Power Generated [MW]	Real Power Cost [$/h]
1	1	1.050	174.16	462.07	1.099	182.013	488.26
2	2	0.976	47.155	121.43	1.076	47.648	123.116
3	5	1.041	24.324	61.304	1.038	22.128	52.731
4	8	0.980	21.106	72.310	1.068	15.921	53.860
5	11	0.990	12.793	42.473	1.098	12.139	40.101
6	13	1.024	13.053	43.419	1.080	12.765	42.371
		Total	292.59	803.17		292.61	800.44

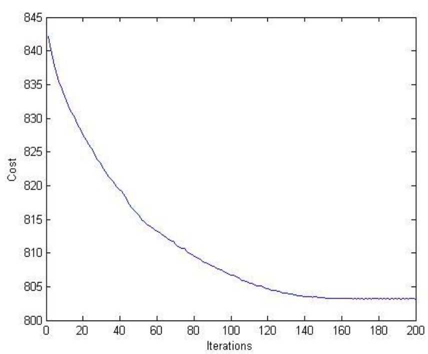

Figure-III Graph between Active power cost ($/h) and iterations in PSO

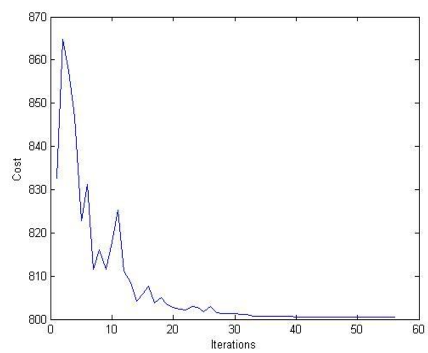

Figure-IV Graph between Active power cost ($/h) and iterations in GA

Table: IV Comparison of results of PSO & GA Method for reactive power generation and cost

		PSO			**GA**		
Unit No.	Bus No.	Gen. Voltage [p.u]	Reactive Power Gen. [Mvar]	Reactive Power Cost [$/h]	Gen.Voltage [p.u]	Reactive Power Gen. [Mvar]	Reactive Power Cost [$/h]
1	1	1.05	10.50	21.434	1.099	3.311	6.66
2	2	0.98	47.155	121.43	1.076	47.64	123.11
3	5	1.04	24.324	61.304	1.038	22.12	52.73
4	8	0.98	21.106	72.310	1.068	15.92	15.92

5	11	0.99	12.793	42.473	1.098	12.13	40.10
6	13	1.02	13.053	43.419	1.080	12.76	42.37
		Total	128.94	362.37		113.91	318.84

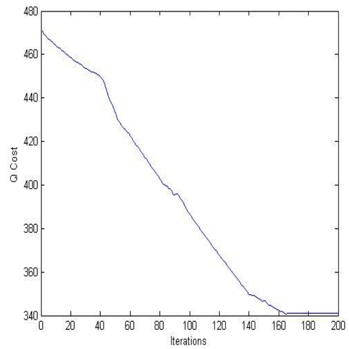

Figure-V Graph between reactive power cost ($/h) And iterations in GA

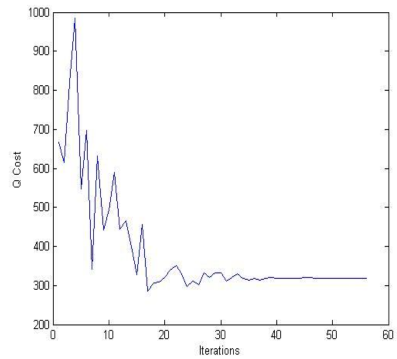

Figure-VI Graph between reactive power cost ($/h) and iterations in PSO

6 . CONCLUSION

Following the case study as discussed in the present work, it is observed that for the same electrical network configurations both the methods are proved almost better but not so much comparatively competitive for the optimal power flow.It is interesting to note that GA and PSO are useful as an optimization technique to solve OPF. The method employs GA and PSO separately to get a feasible point that satisfy the equality and inequality constraints with the desired precision. OPF solutions by using GA and PSO have the advantage not to calculate differential equations neither the Jacobean matrix unlike classical methods. Also the major advantages of these methodologies are that these are relatively versatile for handling various qualitative constraints. These facts permit the definition of any type of objective functions regardless of mathematical condition of continuity, concavities, etc. The main disadvantages of this proposal are the large computing time required to obtain the optimal solution this situation was expected because GA and PSO are stochastic search methods.

REFERENCES

[1.] H.W. Dommel and W. F. Tinney, “Optimal power flow solutions”, IEEE Transactions on Power Apparatus Systems, Vol. PAS-87, pp. 1866–1876, Oct. 1968.

[2.] H. H. Happ, “Optimal power dispatch-A comprehensive survey”, IEEE Transactions on Power Apparatus and Systems, vol.96, no.3, pp. 841-854, 1977.

[3.] IEEE working group, “Description and bibliography of major economic-security functions part-II and III, IEEE Transactions on Power Apparatus and System, vol.100, no.1, pp. 215-235, 1981.

[4.] J. Carpentier, “Optimal power flow, uses, methods and development”, Planning and operation of electrical energy system: proceedings of IFAC symposium, Brazil, pp. 11-21, 1985.

[5.] B. H. Chowdhury, et.al. “Recent advances in economic dispatch”, IEEE Transactions on Power System, vol.5, no.4, pp. 1248-1259, 1990.

[6.] V. Miranda, et.al., “Fuzzy modeling of power system optimal load flow”, IEEE Transactions on Power System, vol. 7, no. 2, pp.843-849, 1992.

[7.] P. H. Chen, et.al., “Large scale economic dispatch by genetic algorithm”, IEEE Transactions on Power System, vol. 10, no. 4, pp. 1919-1926, Nov. 1995.

[8.] Kwang Y. Lee “Optimization Method for Reactive Power Planning by Using a Modified Simple Genetic Algorithm” IEEE Transactions on Power Systems, Vol. 10, No. 4, November 1995.

[9.] Russell Eberhart,et.al., “A New Optimizer Using Particle Swarm Theory” Sixth International Symposium on Micro Machine and Human Science 0-7803-2676-8/95 $4.00 01995 IEEE.

[10.] S. D. Chen, et.al. “A new algorithm based on the Newton-Raphson approach for real-time emission dispatch”, Electric Power System Research, vol. 40, no.2, pp. 137-141, 1997.

[11.] John G. Vlachogiannis and Kwang Y. Lee, Fellow, IEEE “A Comparative Study on Particle SwarmOptimization for Optimal Steady-State Performance of Power Systems” IEEE transction on power system, Vol. 21, no. 4, November 2006.

[12.] 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.

[13.] Yamille del Valle, Student Member, IEEE, Ganesh Kumar Venayagamoorthy, Senior Member, IEEE "Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems" IEEE Transaction on evolutionary computation, vol. 12, no. 2, APRIL 2008.

[14.] Mahdi Rajabzadeh and Masoud A. Golkar "Optimum Allocation of Reactive Power in Real-Time Operation under Deregulated Electricity Market" Journal of Electrical Engineering & Technology Vol. 4, No. 3, pp. 337~345, 2009.

[15.] Mithun Bhaskar "An Expert Based Initial Generation Of Genetic Algorithm with adaptive probability approach for quadratic opf" IU-JEEE Vol. 10(2), (2010), 1257-1266

[16.] Pathak Smita, Prof. B.N.Vaidya "Optimal Power Flow by Particle Swarm Optimization for Reactive Loss Minimization" International Journal of Scientific & Technology Research Volume 1, Issue 1,Feb 2012.

[17.] C. M. Wankhade1, B. P. Saoji & A. P. Vaidya "Comparative Study of GA Based Optimal Power Flow" International Journal on Advanced Electrical and Electronics Engineering, (IJAEEE), ISSN (Print): 2278-8948, Vol.