International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-6 Issue-1, October 2016

Optimal Analysis of Economic Load Dispatch using Artificial Intelligence Techniques

Vijay Kumar, Rakesh Kumar

Abstract: ***Applications of artificial intelligence to economic load dispatch problems are discussed in the paper. The fuelcost equation of a thermal plant is generally expressed as continuous quadratic equation. In real situations the fuel cost equations can be discontinuous. Continuous and discontinuous fuel cost equations are explained here as thermal palnts cost equation are continuous which are further a quadratic equation.GA technique used for 30 bus test system have continuous fuel cost equations. Various results compared with conservative quadratic programming methods to analyze superiority of the suggested artificial intelligence technique. A 10-generator system each with distributed areas is considered and particle swarm algorithm engaged to reduce the cost of generation. All obtained results compared with other conventional methods.***

Keywords: GA, ELD, PSO, Evolutionary methods

I. INTRODUCTION

Economic dispatch problem calculate output power of every generating unit so that all demands are satisfied at minimum cost, while satisfying different technical constraints of the network and the generators. In this problem, the generation costs are represented as curves and the overall calculation minimizes the operating cost by finding the point where the total output of the generators equals the total power that must be delivered. It is an important daily optimization task in the operation of a power system. Many optimization methods applied for solving ELD problems. Most algorithms need the incremental cost curves for smooth increasing .Many generating unitshave non-incremental cost curves. To attain accurate dispatch results, approaches without restriction on the shape of fuel cost functions utlilised. Most of conventional methodssuffer from the convergence problem. Moreover, some methods face dimensionality problem especially when solving the large-scale system. One of the most promising research fields developed named as "Evolutionary Techniques", an area utilizing analogies with nature or social systems. Evolutionary techniques finding popularity within research community as design tools and problem solvers due to their capability. and ability to optimize in complex search spaces applied for non-differentiable objective functions. Several modern heuristic tools have evolved in the last two decades that facilitate solving optimization problems that previously difficult or impossible to solve.

Manuscript published on 30 October 2016.
***** Correspondence Author (s)
Dr. Vijay Kumar, Prof, Director-Principal, Ram Devi Jindal Group of Institutions, Basoli (Punjab). India.
Dr. Rakesh Kumar, Assistant Professor, Punjab Institute of Technology, Bhikhiwind (Punjab). India.

These tools include evolutionary computation, simulated annealing, tabu search, particle swarm, etc. Recently, genetic algorithm (GA) and particle swarm optimization (PSO) techniques appeared as promising algorithms for handling the optimization problems. These techniques are finding popularity within research community as design tools and problem solvers because of their versatility and ability to optimize in complex multimodal search spaces applied to non-differentiable cost functions .Genetic Algorithm observed as a general-purpose search method which is an optimization method & a learning mechanism, based loosely on Darwinian principles of biological evolution, reproduction and ''the survival of the "fittest''. GA maintains a set of candidate solutions called population and repeatedly modifies them. At each step, GA selects individuals from the current population to be parents and uses them to produce the children for the next generation. In general, the fittest individuals of any population tend to reproduce and survive to the next generation, thus improving successive generations.GA extensively applied to solve complex design optimization problems because it can handle. both discrete and continuous variables, non-linear objective and constrain functions without requiring gradient information. A Real-Coded Genetic Algorithm (RCGA) is more efficient in view of CPU time and offers higher precision with more consistent results. In PSO, a set of arbitrarily generated solutions propagates in the design space towards the optimal solution over a number of iterations based on large amount of information about the design space that is assimilated and shared by all members of swarm. PSO and GA are stimulated by nature and have proved effective solutions for optimization problems. Both PSO and GA optimization methods used or optimization problems giving almost similar results. Both continuous and discontinuous fuel cost equations are considered in the present paper. RCGA optimization technique is applied to a 6-generator 26-bus test system having continuous fuel cost equations and results are compared with conventional quadratic programming method to show its superiority. Further, PSO is employed to minimize the cost of generation of a 10-generator system each with discontinuous fuel cost equations and the results are compared with conventional methods.

II. PROBLEM FORMULATION

Economic dispatch problem described as minimizing total fuel cost of all committed plants subject to the constraints .

$$\min F = \sum_{i=1} F_i (P_i)$$

Subject to the constraints

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Fi (Pi) = cost function of i th generating unit of fuel
PD = Power load demand
PL = Power losses
Pi min = Minimum out power output limit of i th generating unit
Pi max = Maximum power output limit of i th generating unit

The total fuel cost is to be minimized subject to the constraints. The transmission loss can be determined form Bmn coefficients. The conditions for optimality can be obtained by using Lagrangian multipliers method and Kuhn tucker conditions as follows:

$$2a_i P_i + b_i = \lambda (1 - 2\sum B_{ij}),\ i = 1,2,......, N$$

With the following constraints

$$\sum_{i=1} P_i = PD + PL$$

$$PL = \sum P_i B_{ij} P_j$$

$$P_{i\,min} \leq P_i \leq P_{i\,max}$$

Steps to solve economic load dispatch problem with the constraints explained below:

Step-1:

Allocate lower limit of each plant as generation, evaluate the transmission loss and incremental loss coefficients and Update the demand.

$$P_i = P_{i\,min},\ X_i = 1 - \sum P_i B_{ij},\ PD_{new} = PD + PL_{old}$$

$$\sum_{i=1}^{N} P_i - PD - PL = 0$$

$$P_{i\,min} \leq P_i \leq P_{i\,max},\ i = 1,2,......, N$$

here

Step-2:

Apply quadratic programming to determine the allocation Pi new of each plant.

If the generation hits the limit, it should be fixed to that Limit and the remaining plants only should be considered for next iteration.

Step-3:

Check for the convergence

$$-PL \leq \varepsilon \sum P_i - PD$$

Where ε is the tolerance. Repeat until the convergence criteria is meet.

A brief description about the quadratic programming method is presented in the next section.

F = Operational cost
N = generating units
Pi = output power of i th generating unit
Fi (Pi) = cost function of i th generating unit of fuel
PD = Power load demand
PL = Power losses
Pi min = Minimum out power output limit of i th generating unit
Pi max = Maximum power output limit of i th generating unit

The total fuel cost is to be minimized subject to constraints. The transmission loss determined formBmn coefficients. The conditions for optimality can be obtained by usingLagrangian multipliers method and Kuhn tucker conditions as follows:

$$2a_i P_i + b_i = \lambda (1 - 2\sum B_{ij}),\ i = 1,2,......, N$$

With the following constraints

$$\sum_{i=1} P_i = PD + PL$$

Steps to solve the economic load dispatch problem with the constraints explained below:

Step-1:

Allocate lower limit of each plant as generation, evaluate the transmission loss and incremental loss coefficients and update the demand.

$$P_i = P_{i\,min},\ X_i = 1 - \sum P_i B_{ij},\ PD_{new} = PD + PL_{old}$$

Step-2:

Apply quadratic programming to determine the allocation Pi new of each plant.If the generation hits the limit, it should be fixed to thatlimit and the remaining plants only should be considered fornext iteration.

Step-3:

Check for the convergence

$$-PL \leq \varepsilon \sum P_i - PD$$

Where ε is the tolerance. Repeat until the convergence criteria is meet. A brief description about the quadratic programming method is presented in the next section.

III. QUADRATIC PROGRAMMING TECHNIQUE

Quadratic programming is a proficient optimization technique
to trace the global minimum if the objective function isquadratic and the constraints are linear. Quadraticprogramming used recursively from the lowest incrementalcost regions to highest incremental cost region to find optimum allocation. Once the limits are obtained and the datais rearranged in such a manner that the incremental costlimits of all the plants are in ascending order.The quadratic program may be written as following:

Minimize

4) If for any allocation for a plant, it is violating the limit, itshould be fixed to that limit and the remaining plantsonly should be considered for next iteration.

5) Repeat steps 2, 3, and 4 till a solution is achieved within a specified tolerance.

IV. GENETIC ALGORITHM TECHNIQUE

$$f(x) = cx + TxQx$$

Subject to

$$Ax \leq b \text{ and } x \geq 0$$

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Where c is an n-dimensional row vector describing the coefficients of the linear terms in the objective function, and is an $(n \times n)$ symmetric matrix describing the coefficients of the quadratic terms. If a constant term exists it is dropped from the model. As in linear programming, the decision variables are denoted by the n-dimensional column vector x,and the constraints are defined by an $(m \times n)$ A matrix and anm-dimensional column vector b of right-hand-side coefficients. We assume that a feasible solution exists and that the constraint region is bounded. When the objective functionf(x) is strictly convex for all feasible points the problem has unique local minimum which is also the global minimum. A sufficient condition to guarantee strictly convexity is for Q tobe positive definite. If there are only equality constraints, then the QP can be solved by a linear system. Otherwise, a variety of methods for solving the QP are commonly used, namely; interior point, active set, conjugate gradient, extensions of the simplex

algorithm etc. The direction search algorithm is minor variation of quadratic programming for discontinuous search space. For every demand the following search mechanism is followed between lower and upper limits of those particular plants. For meeting any demand the algorithm is explained inthe following steps:

1) Assume all the plants are operating at lowest incremental

cost limits.

2) Substitute $P_i = L_i + (U_i - L_i) X_i$,

A. Overview

Genetic Algorithm viewed as a general-purpose search method, an optimization method, or a learning mechanism, based loosely on Darwinian principles of biological evolution, reproduction and "the survival of the fittest." GA maintains a set of candidate solutions called population and repeatedly modifies them. At each step, GA selects individuals at random from the current population

to be parents and uses them to produce the children for next generation. Candidate solutions are usually represented as strings of fixed length, called chromosomes. Given a random initial population GA operates in cycles called generations, as follows :

- Each member of the population is evaluated using an objective function or fitness function.
- The population undergoes reproduction in a number of iterations. One or more parents are chosen stochastically, but strings with higher fitness values have higher probability of contributing an offspring.
- Genetic operators, such as crossover and mutation, are applied to parents to produce offspring.
- The offspring are inserted into the population and the process is repeated.

Over successive generations, the population "evolves" toward an optimal solution. GA applied to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including problems in which the objective functions discontinuous, no differentiable, stochastic, or highly nonlinear.

i. Chromosome Representation

$0 < X_i < 1$ and make the objective function quadratic and make the constraints linear by omitting the higher order terms.

3) Solve the ELD using quadratic programming recursively to find the allocation and incremental cost for each plant within limits of that plant.

4) If there is no limit violation for any plant for that particular piece, then it is a local solution

i. Chromosome Representation

B. Implementation of RCGA

Implementation of GA has the determination of six fundamental issues: chromosome representation, selection function, the genetic operators, initialization, termination and evaluation function. Brief descriptions about these issues are provided in the following sections produce better solutions. Real-coded representation is more efficient in terms of CPU time and offers higher precision with more consistent results.

Arithmetic crossover produces two complimentary linear combinations of the parents, where $r = U(0, 1)$:

ii. Selection Function

To produce successive generations, selection of individuals plays a very significant role in a genetic algorithm. The selection function determines which of the individuals survive and move on to the next generation. A probabilistic selection is performed based upon the individual's fitness such that the superior individuals have more chance of being selected. There are several schemes for the selection process: roulette wheel selection and its extensions, scaling techniques, tournament, normal geometric, elitist models and ranking Methods.

The selection approach assigns a probability of selection P_j

to each individuals based on its fitness value. In the present Study, normalized geometric selection function has been used.

In normalized geometric ranking, the probability of selecting an individual P_i is defined as:

$$X' = rX + (1 - r)Y$$
$$Y' = rY + (1 - r)X$$

Non-uniform mutation randomly selects one variable j and sets it equal to a non-uniform random number.

$$x_i' = \begin{cases} x_i + (b_i - x_i) f(G) & \text{if } r_1 < 0.5, \\ x_i + (x_i + a_i) f(G) & \text{if } r_1 \geq 0.5, \\ x_i \end{cases}$$

where,

$$f(G) = (r_2 (1 - G)) b$$

$$P_i = q'(1 - q)^{r-1}$$

r1, r2 = uniform random nos. between 0 to 1.

G = current generation.

Gmax = maximum no. of generations.

$$q' = \frac{q}{1 - (1 - q)^P}$$

b = shape parameter.

iv. Initialization, Termination and Evaluation Function An initial population is needed to start the genetic algorithm procedure. The initial population can be randomly generated or can be taken from other methods.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

GA moves from generation to generation until a stopping criterion is met. The stopping criterion could be maximum number of generations, population convergence criteria, lack of improvement in the best solution over a specified number of generations or target value for the objective function. Evaluation functions or objective functions of many forms can be used in a GA so that the function can map the population into a partially ordered set.

where,

q = probability of selecting the best individual

r = rank of the individual (with best equals 1)

P = population size

iii. Genetic Operators

The basic search mechanism of the GA is provided by thematic operators. There are two basic types of operators: Crossover and mutation. These operators are used to produce new solutions based on existing solutions in the population. Crossover takes two individuals to be parents and produces two new individuals while mutation alters one individual to produce a single new solution. The following genetic operators are usually employed: simple crossover, arithmetic crossover and heuristic crossover as crossover operator and uniform mutation, non-uniform mutation, multi-non-uniform mutation, boundary mutation as mutation operator. Arithmetic crossover and non-uniform mutation are employed in the present study as genetic operators. Crossover generates random number r from a uniform distribution from 1 to mand creates two new individuals by using equations:

v. Parameter Selection for RCGA

For different problems, it is possible that the same parameters

for GA do not give the best solution and so these can be changed according to the situation. The parameters employed

for the implementations of RCGA in the present study are given in Table I. Optimization were performed with the total number of generations set to 100. The optimization processes

is run 20 times and best among the 20 runs are taken as the final solutions.

TABLE I: PARAMETERS USED IN RCGA

x i' = ⎨ ⎧ ⎩ y i otherwise ⎭ ⎫ ⎧x i , if i < r ⎫ ⎩ y i , if i < r ⎭ y i' = ⎨ ⎧

Parameters : 80, 40

Normal geometric [0 0.09]

Arithmetic [2]

Nonuniform [2 1030]

x (jt,)g = the g-th component of the position of particle j at iteration t

pbest j = pbest of particle j

gbest = gbest of the group v (jt,)g = the g-th component of the velocity of particle j at

C. Proposed RCGA Approach

The following steps are followed for the implementation of GA for economic load dispatch problems.

1) Select the plant having maximum capacity and range as a reference plant.

2) Fix the reference plant allocation by equation (5) and (6).

3) Convert the constrained optimization problem as an unconstrained problem by penalty function method as:

iteration

minmaxt , v g ≤ v (jt,)g ≤ v g ;

w = inertia weight factor

c1 , c 2 = cognitive and social acceleration factors respectively

r1 , r2 = random numbers uniformly distributed in the range (0, 1)

F = ∑ Fi (Pi) + 1000 *

i =1

NN N| (∑ Pi − PD − ∑ ∑ Bij Pi P j) | i =1i =1 j =1min

V. RESULTS AND DISCUSSIONS

A. Numerical Example 1

First, continuous quadratic cost curve for the plants is considered. The system consists of 26 bus and the demand of the system was divided into 10 small intervals as shown in Fig. Generating units' data are given in Table.

The cost function coefficients along with minimum and Maximum generation capacity for each fuel option are given in Table III. Table IV, shows the optimal generators' power Outputs for each hour including their corresponding fuel costs using quadratic programming method. Total production cost of 10 intervals is $150065.8. Table V, shows the same using RCGA method. Total production cost of 10 intervals is $153118. It is clear from Table IV and V that RCGA gives better solutions.

g. 1. Load pattern of numerical example 1

The system is found to have minimum and maximum generation capacity of 1273 MW and 3599 MW, respectively.

The load demand is assumed to vary between 2450 MW and 2760 MW in steps of 50 MW. The results obtained by the proposed PSO method are given in Table VII. The results are compared in Table VIII with the results obtained by other methods to show its superiority.

Unit/Cst

Unit-1

Unit-2

Unit-3

Unit-4

Unit-5

Unit-6

TABLE III. DATA FOR EXAMPLE - 1: 26-BUS 6-UNIT TEST SYSTEM

A	B	C	Pmin	1	2	3	4	5	Cost
0.008	8	230	90	333.410	356.112	374.632	397.824	432.121	9423.7
0.0085	9	220	1000	107.323	99.432	286.430	164.234	171.231	10833.3
0.007	7.5	200	90	191.025	181.342	260.232	241.121	257.623	11999.6
0.008	10	210	60	59.421	58.343	77.343	77.880	124.264	13231.0
0.007	9.5	210	60	79.908	88.365	149.909	321.213	157.350	14707.0
0.0085	11	150	60	60	55.767	60	55	67.487	149.230

TABLE V: RESULTS OF RCGA FOR EXAMPLE - 1: 26-BUS, 6-UNIT TEST SYSTEM

U/T										
1	2	3	4	5	6	7	8	9	10	
359.323	377.712	423.819	418.300	451.009	471.090	484.102	479.005	411.110	381.121	Pg1
103.109	109.909	139.901	142.212	165.616	181.232	183.414	183.212	139.097	121.019	Pg2
187.321	189.454	229.313	212.312	253.006	266.717	274.419	265.616	219.571	213.079	Pg3
49.909	57.708	86.565	89.213	119.434	137.709	140.012	139.717	80.917	77.010	Pg4
77.787	84.986	111.090	118.912	149.010	166.045	177.819	171.212	119.631	103.776	Pg5
48.898	53.282	50.672	58.912	57.615	89.034	97.098	93.212	49.910	53.561	Pg6
9599.0	10404.0	12333.0	12331.0	14465.0	15741.0	16550.0	16330.0	11880.0	11202.0	Total Cost

VI. CONCLUSION

In this paper, optimal result obtained using genetic algorithms techniques for economic load dispatch problems for both continuous and discontinuous fuel cost functions. For a continuous fuel cost function with 26 bus, 6 unit test system using both conventional(quadratic programming) and artificial intelligence (real coded genetic algorithm) methods to find the optimum generator allocation. It is observed that outcomes attained using artificial intelligence method are better compared to quadratic programming method.

REFERENCES

1. Jiang and S. Ertem, “Economic dispatch with non-monotonically increasing incremental cost units and transmission system losses”, IEEE Transactions on Power Systems, vol. 10, no. 2, pp. 891-897, May 1995.
2. H.W. Dommel, “Optimal power dispatch”, IEEE Transactions on Power Apparatus and Systems, PAS93 No. 3, pp. 820–830, 1974.
3. N. Ramaraj and K. Nagappan, “Analytical method to optimize generation schedule”, Journal of The Institution of Eng-neers (India), vol. 66, p 240, 1987.
4. C.E. Lin and G.L. Vivianib, “Hierarchical Economic Dispatch of Piecewise Quadratic Cost Functions”, IEEE Transactions of PAS, vol.103, no 6, June, 1984.
5. N. Ramaraj and R Rajaram, “Analytical approach to optimize generation schedule of plant with multiple fuel options”, Journal of The Institution of Engineers (India), vol. 68, p 106, 1987
6. N. Ramaraj and R Rajaram, “Analytical approach to optimize generation schedule of plant with multiple fuel options”, Journal of The Institution of Engineers (India), vol. 68, p 106, 1987.
7. J.V. Guttag, The Specification and Application to Program-ming of Abstract Data Types, Ph.D. Thesis, Dept. of Computer Science, University of Toronto (1975).
8. J.V. Guttag and J.J. Horning, “Formal Specification as a De-sign Tool,”Seventh ACM Symposium on Principles of Program-ming Languages, Las Vegas (1998), pp-2-9.
9. J.V. Guttag, “Notes on Type Abstraction, Version 2,’’ IEEE Transactions on Software Engineering,pp-46-49, vol. SE-6, no. 1 (1980).
10. C.A.R. Hoare, “An Axiomatic Basis for Computer Pro-gramming,”Communications of the ACM, vol. 12, no. 10 (1985).
11. A.Igelais, “Proofs of Correctness of Data Representations,” Acta Informatica, pp- 56,vol. 1, no. 4 (2006).
12. J. H. Holland, Adaptation in Natural and Artficial Systems, Universi-tyof Michigan Press, Ann Arbor, MI, 1975.S. Koziel and Z. Michalewicz, Evolutionary algorithms, homomorphous map-pings, and constrained parameter optimization, Evolutionary Computation 7(1), 19-44, 1999.
13. T.S. Metcalfe, P. Charbonneau, Stellar structure modeling using a parallel genetic algorithm for objective global optimiza-tion, Journal of Computational Physics 185, 176-193, 2003.
14. Z. Michalewicz, A Survey of Constraint Handling Tech-niques in Evolutionary Computation Methods, Evolutionary Pro-gramming, Vol.4, pp.135, 1995.Z. Michalewicz, Genetic algo-rithms + data structures =evolution pro-grams,Berlin,Springer,1996

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)