Techniques For Determine Economic Load Dispatch Problem Using Particle Swarm Optimization

Ankur Saxena[1]

[1,] *Department of Electrical Engineering, Madhav Institute of Technology &Science, India*

Abstract— **Economic load dispatch (ELD) is the method of allocating generation within all available generating units in such a way the it fulfill the load demand and satisfied constraints to minimized total generation cost of the thermal power plant. Economic load dispatch is a common task in operational planning of power system that require is to be optimized. This paper presents overview of most reliable and very effective particle swarm optimization technique for economic load dispatch. These results have been shown for ELD of three and six unit test systems or generator systems with and without losses. All the final results obtained using PSO will compare with conventional quadratic programming techniques and found to be most convenient.**

Keywords— **Economic Load Dispatch, Particle Swarm Optimization, Quadratic Programming techniques, Operational Planning.**

I.   Introduction

The economic load dispatch (ELD) of power generating units has always occupied an important position in the electric power industry. ELD is a computational process where the total required generation is distributed among the generation units in operation, by minimizing the selected cost criterion, subject to load and operational constraints. For any specified load condition, ELD determines the power output of each plant (and each generating unit within the plant) which will minimize the overall cost of fuel needed to serve the system load [1]. ELD is used in real-time energy management power system control by most programs to allocate the total generation among the available units. ELD focuses upon coordinating the production cost at all power plants operating on the system.

In the traditional ELD problem, the cost function for each generator has been approximately represented by a single quadratic function and is solved using mathematical programming based optimization techniques such as lambda iteration method, gradient-based method, etc [2]. These methods require incremental fuel cost curves which are piecewise linear and monotonically increasing to find the global optimal solution.

Dynamic programming (DP) method [3] is one of the approaches to solve the non-linear and discontinuous ELD problem, but it suffers from the problem of "curse

of dimensionality" or local optimality. In order to

overcome this problem, several alternative methods have been developed such as genetic algorithm [4],

evolutionary programming [5, 6], tabu search [7], neural network [8], and particle swarm optimization [9-11].

Particle swarm optimization (PSO) is suggested by Kennedy and Eberhart based on the analogy of swarm of birds and school of fish [12]. PSO mimics the behavior of individuals in a swarm to maximize the survival of the species. In PSO, each individual makes his decision using his own experience together with other individuals" experiences. The algorithm, which is based on a metaphor of social interaction, searches a space by adjusting the trajectories of moving points in a multidimensional space. The individual particles are drawn stochastically toward the position of present velocity of each individual, their own previous best performance, and the best previous performance of their neighbors. The main advantages of the PSO algorithm are summarized as: simple concept, easy implementation, robustness to control parameters, and computational efficiency when compared with mathematical algorithms and other heuristic optimization techniques [12, 13]. PSO can be easily applied to nonlinear and non-continuous optimization problem.

In this paper, a PSO technique for solving the ELD problem in power system is proposed. The feasibility of the proposed method was demonstrated for a three units and six units system and the results were compared with quadratic programming method [14]. The results indicate the applicability of the proposed method to the practical ELD problem.

The rest of this paper is organized as follow. Section 2 presents the ELD formulation. Section 3 presents quadratic programming method. Section 4 proposes PSO technique to solve ELD problem. Results and discussions are given in section 5, and section 6 gives some conclusions.

II. Economic Load Dispatch Formulation

The objective of an ELD problem is to find the optimal combination of power generations that minimizes the total generation cost while satisfying an equality constraint and inequality constraints.

The fuel cost curve for any unit is assumed to be approximated by segments of quadratic functions of the active power output of the generator. For a given power system network, the problem may be described as optimization

(minimization) of total fuel cost as defined by (1) under a set of operating constraints.

$$F_T = \sum_{i=1}^{n} F(P_i) = \sum_{i=1}^{n} \left(a_i P_i^2 + b_i P_i + c_i \right) \qquad (1)$$

where F_T is total fuel cost of generation in the system

\$/hr), a_i, b_i, and c_i are the cost coefficient of the i th generator, P_i is the power generated by the i th unit and n is the number of generators.

The cost is minimized subjected to the following generator capacities and active power balance constraints.

$$P_{i,\min} \le P_i \le P_{i,\max} \quad \text{for } i = 1,2, \ , n$$

where $P_{i,\ min}$ and $P_{i,\ max}$ are the minimum and maximum power output of the i th unit.

$$P_D = \sum_{i=1}^{n} P_i - P_{Loss} \qquad (2)$$

where P_D is the total power demand and P_{Loss} is total transmission loss.

The transmission loss P_{Loss} can be calculated byusing **B** matrix technique and is defined by (4) as,

$$\begin{matrix} n & n \\ & \\ i=1 & j=1 \end{matrix}$$

where B_{ij} „s are the elements of loss coefficient matrix **B**.

III. Quadratic Programming Method

A linearly constrained optimization problem with a quadratic objective function is called a Quadratic Program (QP). Due to its numerous applications; quadratic programming is often viewed as a discipline in and of itself. Quadratic programming is an efficient optimization technique to trace the global minimum if the objective function is quadratic and the constraints are linear. Quadratic programming is used recursively from the lowest incremental cost regions to highest incremental cost region to find the optimum Minimize

The general $f(x) = cx + \dfrac{1}{2} x^T Q x$

quadratic programming can be written as:

$$P_{Loss} = \sum \sum P_i B_{ij} P_j$$

.

Subject to $Ax \le b$ and $x \ge 0$

where **c** is an n-dimensional row vector describing the coefficients of the linear terms in the objective function, and **Q** 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 an m-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 function $f(\mathbf{x})$ is strictly convex for all feasible points the problem has a unique local minimum which is also the global minimum. A sufficient condition to guarantee strictly convexity is for **Q** to be positive

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 in the following steps:

1) Assume all the plants are operating at lowest incremental cost limits.

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

   where $0 < X_i < 1$ 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.

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

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

exchanges previous experiences among themselves [12]. PSO as an optimization tool provides a population based search procedure in which individuals called particles change their position with time. In a PSO system, particles fly around in a multi dimensional search space. During flight each particles adjust its position according its own experience and the experience of the neighboring particles, making use of the best position encountered by itself and its neighbors.

In the multi-dimensional space where the optimal solution is sought, each particle in the swarm is moved

toward the optimal point by adding a velocity with its position. The velocity of a particle is influenced by three components, namely, inertial, cognitive, and social. The inertial component simulates the inertial

behavior of the bird to fly in the previous direction. The cognitive component models the memory of the bird about its previous best position, and the social component models the memory of the bird about the

best position among the particles. The particles move around the multi-dimensional search space until they find the optimal solution. The modified velocity of each agent can be calculated using the current velocity and the distance from Pbest and Gbest as given below.

$$v_{ij}^{t} = w \times V_{ij}^{t-1} + C_1 \times r_1 \times \left(Pbest_{ij}^{t-1} - X_{ij}^{t-1} \right) +$$
$$C_2 \times r_2 \times \left(Gbest_i^{t-1} - X_{ij}^{t-1} \right)$$
$$i = 1,2, \ , N_D \ ; j = 1,2, \ , N_{par} \qquad (7)$$

Using the above equation, a certain velocity, which gradually gets close to Pbest and Gbest, can be calculated. The current position (searching point in the solution space), each individual moves from the current position to the next one by the modified velocity in (7) using the following equation:

$$X_{ij}^{t} = X_{ij}^{t-1} + V_{ij}^{t} \qquad (8)$$

where,

t	Iteration count
V_{ij}^{t}	Dimension i of the velocity of particle j at iteration t
X_{ij}^{t}	Dimension i of the position of particle j at iteration t
w	Inertia weight
C_1 , C_2	Acceleration coefficients
$Pbest_{ij}^{t}$	Dimension i of the own best position of particle j until iteration t
$Gbest_{ij}^{t}$	Dimension i of the best particle in the swarm at iteration t

N_D	Dimension of the optimization problem (number of decision variables)
N_{par}	Number of particles in the swarm
r_1 , r_2	Two separately generated uniformly distributed random numbers in the range [0, 1]

The following weighting function is usually utilized:

$$\omega = \omega_{max} - \left(\frac{\omega_{max} - \omega_{min}}{Iter_{max}} \right) \times Iter \qquad (9)$$

where,

$\omega_{max} , \omega_{min}$	Initial and final weights
$Iter_{max}$	Maximum iteration number
$Iter$	Current iteration number

Suitable selection of inertia weight in above equation provides a balance between global and local explorations, thus requiring less number of iterations on an average to find a sufficient optimal solution. As originally developed, inertia weight often decreases linearly from about 0.9 to 0.4 during a run.

The algorithmic steps involved in particle swarm optimization technique are as follows:

1) Select the various parameters of PSO.

2) Initialize a population of particles with random positions and velocities in the problem space.

3) Evaluate the desired optimization fitness function for each particle.

4) For each individual particle, compare the particles fitness value with its Pbest. If the current value is better than the Pbest value, then set this value as the Pbest for agent i.

5) Identify the particle that has the best fitness value. The value of its fitness function is identified as Gbest.

6) Compute the new velocities and positions of the particles according to equation (7) & (8).

7) Repeat steps 3-6 until the stopping criterion of maximum generations is met.

The procedure of the particle swarm optimization technique can be summarized in the flowchart of Fig. 1.
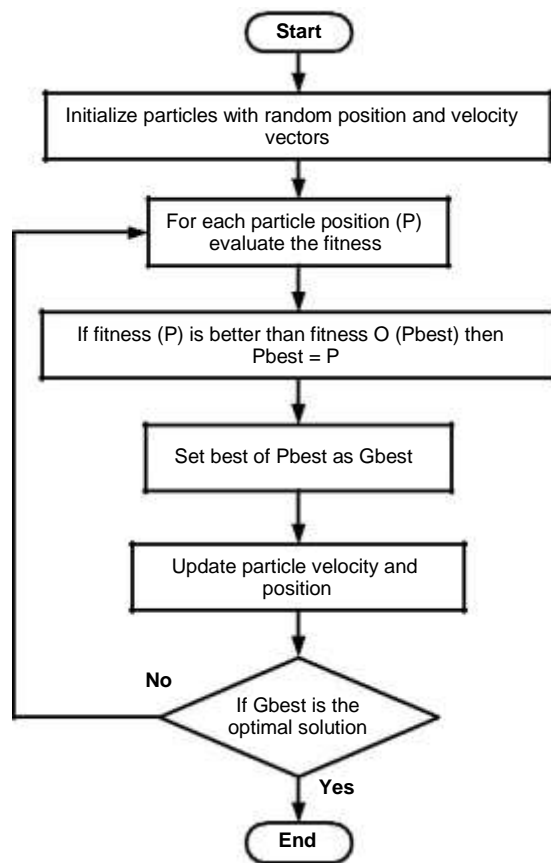
Fig. 1: The flowchart of PSO technique

V. Results And Discussions

To verify the feasibility and effectiveness of the proposed PSO algorithm, two different power systems were tested one is three generating units [15] and other is six generating units [16, 17]. Results of proposed particle swarm optimization (PSO) are compared with quadratic programming methods. A reasonable B-loss coefficients matrix of power system network has been employed to calculate the transmission loss. The software has been written in the MATLAB-7.8 language.

Case Study-1: 3-units system

In this case, a simple power system consists of three-unit thermal power plant is used to demonstrate how the work of the proposed approach. Characteristics of thermal units are given in Table 1, the following coefficient matrix B_{ij} losses.

Table 1: Generating unit capacity and coefficients

Unit	$\overset{min}{P_i}$ (MW)	$\overset{max}{P_i}$ (MW)	a_i ($/MW^2)	b_i ($/MW)	c_i ($)
1	50	250	0.00525	8.663	328.13
2	5	150	0.00609	10.04	136.91
3	15	100	0.00592	9.76	59.16

$$B_{ij} = \begin{bmatrix} 0.000136 & 0.0000175 & 0.000184 \\ 0.000175 & 0.0001540 & 0.000283 \\ 0.000184 & 0.0002830 & 0.001610 \end{bmatrix}$$

By using the proposed PSO technique obtained the results as shown in Table 2 and Table 3. Test results in Table 2 for 3-generator system with load change from 350 MW to 500 MW without taking into account transmission losses. Table 3 shows the optimal power output, total cost of generation, as well as active power loss for the power demands of 375 MW, 400 MW and 550 MW. Table 3 shows that the PSO method is better than conventional method (quadratic programming) for each loading.

Table 2: Best power output for 3-generator system (without loss)

P_{demand} (MW)	P1 (MW)	P2 (MW)	P3 (MW)	Fcost ($/hr)
350	151.09	42.04	56.87	2959.98
375	173.00	37.78	64.22	3219.23
400	186.18	52.70	61.12	3483.73
425	195.85	60.96	68.19	3749.95
550	205.57	61.51	82.92	4017.52
405	219.67	68.15	87.18	4288.92
500	210.38	89.70	99.92	4563.00

Table 3: Best power output for 3-generator system

P_{demand} (MW)	Method	P1 (MW)	P2 (MW)	P3 (MW)	PLoss (MW)	Fcost ($/hr)
375	Conv.	193.82	74.78	15.00	8.61	3333.14
	PSO	189.31	79.13	15.00	**8.44**	**3332.69**
400	Conv.	207.68	87.40	15.00	10.08	3621.53
	PSO	212.56	82.48	15.00	**10.04**	**3620.09**
450	Conv.	235.58	112.89	15.00	13.47	4215.18
	PSO	228.15	119.73	15.00	**12.88**	**4211.08**

Case Study-2: 6-units system

In this case, a standard six-unit thermal power plant (IEEE 30 bus test system) is used to demonstrate how the work of the proposed approach, as shown in Fig. 2. Characteristics of thermal units are given in Table 4, the following coefficient matrix B_{ij} losses.

The simulation results with the proposed PSO technique respectively are shown in Table 5, Table 6 and Table 7 with the variation of loading 700 MW, 800 MW and 900 MW. From the simulation results show that the generation output of each unit is obtained correction reduces the total cost of generation and transmission losses when compared with the conventional method (quadratic programming method).
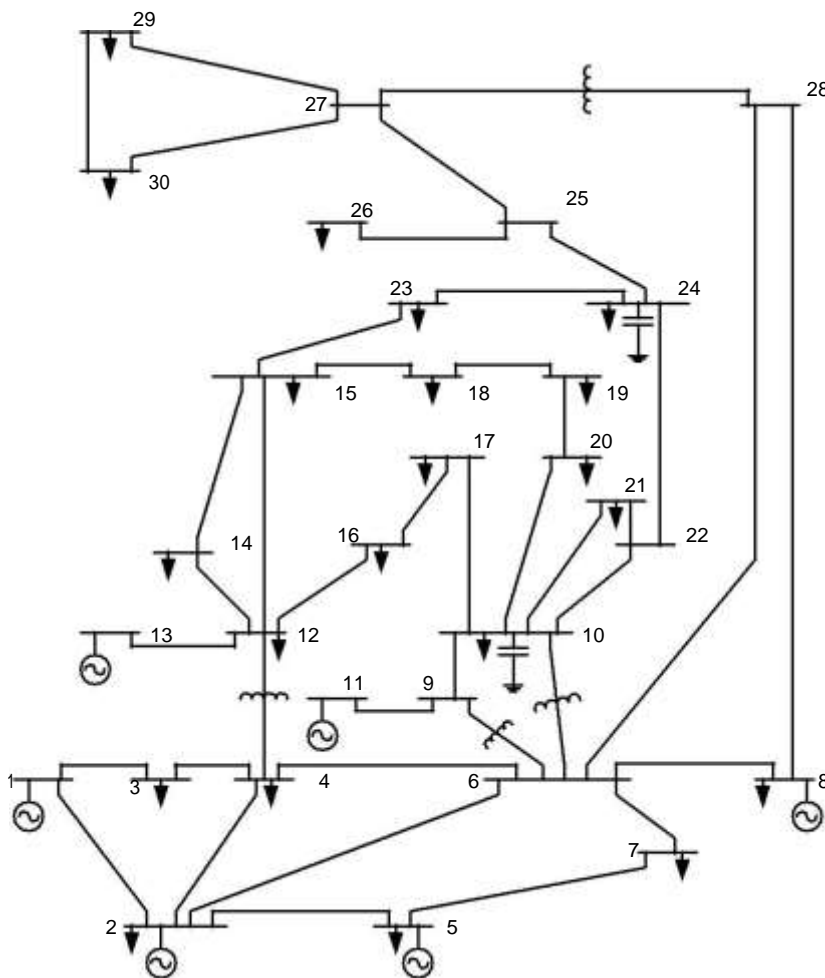

Fig. 2: IEEE 30 bus 6 generator test system

Table 4: Generating unit capacity and coefficients

Unit	P_i^{min} (MW)	P_i^{max} (MW)	a_i ($/MW^2)	b_i ($/MW)	c_i ($)
1	10	125	0.15240	38.53973	756.79886
2	10	150	0.10587	46.15916	451.32513
3	35	225	0.02803	40.39655	1049.9977
4	35	210	0.03546	38.30553	1243.5311
5	130	325	0.02111	36.32782	1658.5596
6	125	315	0.01799	38.27041	1356.6592

$$B_{ij} = \begin{bmatrix} 0.000140 & 0.000017 & 0.000015 & 0.000019 & 0.000026 & 0.000022 \\ 0.000017 & 0.000060 & 0.000013 & 0.000016 & 0.000015 & 0.000020 \\ 0.000015 & 0.000013 & 0.000065 & 0.000017 & 0.000024 & 0.000019 \\ 0.000019 & 0.000016 & 0.000017 & 0.000071 & 0.000030 & 0.000025 \\ 0.000026 & 0.000015 & 0.000024 & 0.000030 & 0.000069 & 0.000032 \\ 0.000022 & 0.000020 & 0.000019 & 0.000025 & 0.000032 & 0.000085 \end{bmatrix}$$

Table 5: Best power output for 6-generator system (P_D = 700 MW)

Unit Output	Conventional	PSO
P1 (MW)	23.90	23.84
P2 (MW)	10.00	10.00
P3 (MW)	95.63	95.57
P4 (MW)	100.70	100.52
P5 (MW)	202.82	202.78
P6 (MW)	182.02	181.52
Total power output (MW)	615.07	**614.23**
Total generation cost ($/hr)	32096.58	**32094.69**
Power losses (MW)	15.07	**14.23**

Table 6: Best power output for 6-generator system (P_D = 800 MW)

Unit Output	Conventional	PSO
P1 (MW)	28.33	28.28
P2 (MW)	10.00	10.00
P3 (MW)	118.95	119.02
P4 (MW)	118.67	118.79
P5 (MW)	230.75	230.78
P6 (MW)	212.80	212.56
Total power output (MW)	719.50	**719.43**
Total generation cost ($/hr)	36914.01	**36912.16**
Power losses (MW)	19.50	**19.43**

Table 7: Best power output for 6-generator system (P_D = 900 MW)

Unit Output	Conventional	PSO
P1 (MW)	32.63	32.67
P2 (MW)	14.48	14.45
P3 (MW)	141.54	141.73
P4 (MW)	136.04	136.56
P5 (MW)	257.65	257.37
P6 (MW)	243.00	242.54
Total power output (MW)	825.34	**825.32**
Total generation cost ($/hr)	41898.45	**41896.66**
Power losses (MW)	25.34	**25.32**

VI. Conclusion

This paper presents an efficient and simple approach for solving the economic load dispatch (ELD) problem. This paper demonstrates with clarity, chronological development and successful application of PSO technique to the solution of ELD. Two test systems 3-generator and 6-generator systems have been tested and the results are compared with quadratic programming method. The proposed approach is relatively simple, reliable and efficient and suitable for practical applications.

References

[1] A.J Wood and B.F. Wollenberg, *Power Generation, Operation, and Control*, John Wiley and Sons, New York, 1984.

[2] Hadi Saadat, *Power System Analysis*, Tata McGraw Hill Publishing Company, New Delhi, 2002.

[3] Z.X. Liang and J. D. Glover. A Zoom Feature for a Dynamic Programming Solution to Economic Dispatch Including Transmission Losses, IEEE Transactions on Power Systems, 1992, 7(2): 544-550.

[4] A. Bakirtzis, et al. Genetic Algorithm Solution to the Economic Dispatch Problem, IEE Proc. Generation, Transmission, and Distribution, 1994, 141(4): 377-382.

[5] H. T. Yang, P. C. Yang, and C. L. Huang. Evolutionary Programming Based Economic Dispatch for Units with Non-Smooth Fuel Cost Functions, IEEE Trans. on Power Systems, 1996, 11(1): 112-118.

[6] N. Sinha, R. Chakrabarti, and P.K. Chattopadhyay. Evolutionary Programming Techniques for Economic Load Dispatch, IEEE Trans. on Evolutionary Computations, 2003, 7(1): 83-94.

[7] W. M. Lin, F. S. Cheng, and M. T. Tsay. An Improved Tabu Search for Economic Dispatch with Multiple Minima, IEEE Trans. on Power Systems, 2002, 17(1): 108-112.

[8] K. Y. Lee, A. Sode-Yome, and J. H. Park, Adaptive Hopfield Neural Network for Economic Load Dispatch. IEEE Trans. on Power Systems, 1998, 13(2): 519-526.

[9] J. B. Park, K. S. Lee, J. R. Shin, and K. Y. Lee. A Particle Swarm Optimization for Economic Dispatch with Nonsmooth Cost Functions, IEEE Trans. on Power Systems, 2005, 20(1): 34-42.

[10] S.Y. Lim, M. Montakhab, and H. Nouri. Economic Dispatch of Power System Using Particle Swarm Optimization with Constriction Factor, International Journal of Innovations in Energy Systems and Power, 2009, 4(2): 29-34.

[11] Z. L. Gaing, Particle Swarm Optimization to Solving the Economic Dispatch Considering the Generator Constraints, IEEE Transactions on Power Systems, 2003, 18(3): 1187-1195.

[12] J. Kennedy, and R. C. Eberhart. Particle Swarm Optimization, *Proceedings of IEEE International Conference on Neural Networks (ICNN'95)*, 1995, 4: 1942-1948, Perth, Australia.

[13] Y. Shi and R. C. Eberhart, Particle Swarm Optimization: Development, Applications, and Resources, Proceedings of the 2001 Congress on Evolutionary Computation, 2001, 1: 81-86.

[14] RMS Danaraj and F Gajendran. Quadratic Programming Solution to Emission and Economic Dispatch Problems, Journal of The Institution of Engineers (India), pt EL, 2005, 86: 129-132.

[15] M. Vanitha, and K. Tanushkodi. Solution to Economic Dispatch Problem By Differential Evolution Algorithm Considering Linear Equality and Inequality Constrains. International Journal of Research and Reviews in Electrical and Computer Engineering, 2011, 1(1): 21-26.

[16] Anis Ahmad, Nitin Singh, and Tarun Varshney. A New Approach for Solving Economic Load Dispatch Problem, MIT International Journal of Electrical and Instrumentation Engineering, 2011, 1(2): 93-98.

[17] M. Sudhakaran, P. Ajay D Vimal Raj, and T. G. Palanivelu. Application of Particle Swarm

Optimization for Economic Load Dispatch Problems, The 14th International Conference on Intelligent System Applications to Power Systems, Kaohsiung, Taiwan, 2007: 668-674.

.