A Swarm Intelligence Approach to the Power Dispatch Problem

D.C. Secui, I. Felea, S. Dzitac, L. Popper

Dinu Călin Secui, Ioan Felea, Simona Dzitac, Laurenţiu Popper
University of Oradea, Faculty of Energy Engineering, Romania
E-mail: csecui@uoradea.ro, ifelea@uoradea.ro, simona.dzitac@gmail.com,
director@perfect-service.ro

Abstract: This paper examines how two techniques of the Particle Swarm Optimization method (PSO) can be used to solve the Economic Power Dispatch (EPD) problem. The mathematical model of the EPD is a nonlinear one, PSO algorithms being considered efficient in solving this kind of models. Also, PSO has been successfully applied in many complex optimization problems in power systems. The PSO techniques presented here are applied to three case studies, which analyze power systems having four, six, respectively twenty generating units.

Keywords: economic dispatch problem, constrained optimization, particle swarm optimization.

1 Introduction

An important issue in optimizing the power systems is the economic power dispatch. This problem consists in determining the power generated by the plant units of a system in order to minimize the total generation cost of units, taking into account the active power balance and the constraints imposed to the capability of the units.

The mathematical optimization model is nonlinear, where both the objective function and the restrictions imposed by equations describing the system functionality are nonlinear.

The mathematical model can be solved using conventional optimization techniques such as the lambda iteration method, the gradient method and others [1, 2], if constraints are considered as linear, the objective function is continuous and the domain of the values is convex. Some disadvantages that arise in these situations - long solving time, objective function discontinuity etc. - may be overcome by applying the artificial intelligence techniques. The most common optimization techniques based upon artificial intelligence used for solving economic power dispatch problems are: the genetic algorithm [3–6], the Hopfield neural networks [2, 7], the differential algorithm [8], the evolutionary programming [9, 10], fuzzy-optimization [12, 13], tabu search [14], particle swarm optimization [15, 16, 27, 28]. Also, the EPD can be formulated as a multi-objective optimization problem [11, 13, 17].

TWe mention that the particle swarm optimization method was successfully applied to other optimization problems, such as optimal power flow [18–20], reactive power optimization and voltage control [23], power loss reduction in distribution systems [24], network reconfiguration [25], unit commitment problem [26], due to its good convergence, low computational time and good quality solutions.

In this paper there are proposed two versions of applying PSO method for solving EPD, comparing the results obtained for three systems consisting of four, six and respectively, twenty generating units.

The paper is structured as follows: section 2 exposes the EPD problem, section 3 and 4 describe the proposal for solving this problem using PSO algorithm, section 5 presents the results obtained through the application of PSO algorithm for three power systems, and section 6 outlines the conclusions drawn.

2 Formulation of the economic power dispatch problem

We consider a power system containing n generating units, each unit having its own generated power P_j, j = 1...n. The total load required in the system is considered to be known and equal to P_D. The fuel

cost $(F_j(P_j))$ for each generator j is represented by a quadratic function:

$$F_j(P_j) = a_j \cdot P_j^2 + b_j \cdot P_j + c_j \tag{1}$$

where: a_j, b_j and c_j are fuel cost coefficients of generator j;
P_j represent the power of generator j.

EPD solution consists in determining the Pj powers of generating units, so that the total fuel cost of the entire system to be minimal, respecting the restriction of power balance on the overall system and the inequality restrictions for each unit j.

The objective function is:

$$\min F = \sum_{j=1}^{n} F_j(P_j) \tag{2}$$

The problem constraints are given by relations (3) and (4):

$$F_p = \sum_{j=1}^{n} P_j - P_D - \Delta P = 0 \tag{3}$$

$$P_j^{min} \leq P_j \leq P_j^{max}, j = 1, 2, ..., n \tag{4}$$

Where: P_j^{min} and P_j^{max} represent the maximum and the minimum operation limits of a generator j;

The power loss at the level of the entire system is a quadratic function in relation to variables P_j and it is calculated by using constant B coefficient formula:

$$\Delta P = \sum_{i=1}^{n} \sum_{j=1}^{n} P_i \cdot B_{ij} \cdot P_j + \sum_{i=1}^{n} B_{0i} \cdot P_i + B_{00} \tag{5}$$

Where B_{ij} is an element of the loss coefficient matrix, B_{i0} is the element i of the loss coefficient vector, and B_{00} is the loss coefficient constant.

3   Presentation of different Particle Swarm Optimization techniques

PSO is a heuristic algorithm, used for solving nonlinear and noncontinuous optimization problems, being introduced by Kennedy and Eberhart [22], in 1995. Since then several techniques in applying the PSO method have been developed, but in the current paper only two techniques are presented, namely the classical PSO (PSO Classical) and PSO with time varying acceleration coefficients (PSO Accelerated).

Classical PSO: To search for the optimal solution in a space with the dimension n, PSO uses a population of NP particles. For a given particle i within NP population, vector solutions at a certain iteration k are represented by $X_i^k = (x_{i1}^k, x_{i2}^k, ..., x_{ij}^k, ..., x_{in}^k)$. In any optimization process, switching from one solution (x_{ij}^k) to another solution (x_{ij}^{k+1}) is accomplished by using the velocity of particles, represented by the vector $V_i^k = (v_{i1}^k, v_{i2}^k, ..., v_{ij}^k, ..., v_{in}^k)$, according to the relation:

$$X_i^{k+1} = X_i^k + V_i^{k+1}, i = 1, 2, ..., NP \tag{6}$$

The updated velocity of the particle in the next iteration (k+1) is given by the relation:

$$X_i^{k+1} = \omega \cdot V_i^k + c_1 \cdot r_1 \cdot (Pbest_i^k - X_i^k) + c_2 \cdot r_2 \cdot (Gbest^k - X_i^k) \tag{7}$$

Where: V_i^k, V_i^{k+1} represent the velocity vector of particle i at iteration k, respectively $k+1$;
X_i^k, X_i^{k+1} represent the solution vector of particle i at iteration k, respectively $k+1$;

$Pbest_i^k$ represent the best solution vector of particle i, until iteration k;

$Gbest^k$ represent the vector corresponding to the best solution of the group, until iteration k;

c_1 and c_2 are coefficients corresponding to cognitive and social behavior;

r_1 and r_2 are random numbers between 0 and 1, and w is the inertia weight factor determined using the relation:

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{k_{max}} \cdot k \tag{8}$$

Where ω_{max} and ω_{min} represent the initial, respectively final weights, k_{max} is maximum iteration number and k is current iteration number for the algorithm.

PSO with time varying acceleration coefficients tries to improve the global search in the early stages of the optimization process and to accelerate the convergence of the particles to the global optimum in the final part of the process. In this case, the calculation of the velocity (V_i^{k+1}) and of the solution (X_i^{k+1}) for the next iteration is done with relations (6) and (7), and according to [21], the coefficients c_1 and c_2 are determined by the relations:

$$c_1 = (c_{1f} - c_{1i}) \cdot \frac{k}{k_{max}} + c_{1i} \tag{9}$$

$$c_2 = (c_{2f} - c_{2i}) \cdot \frac{k}{k_{max}} + c_{2i} \tag{10}$$

Where c_{1i}, c_{2i}, c_{1f} and c_{2f} are initial and final weights for cognitive and social acceleration coefficients.

4   The methodology based on PSO for solving the EPD problem

The implementation of PSO techniques for solving the EPD problem involves the following steps: Step1. *Initialization of the parameters and of PSO solution.* The PSO algorithm parameters are set in reference to ω_{max}, ω_{min}, k_{max}, number of particles (NP), coefficients c_1 and c_2 (for PSO Classical), respectively, c_{1i}, c_{2i}, c_{1f} and c_{2f} (PSO Accelerated). Initially a population of *NP* particles is randomly formed. Each particle defines a possible solution to the problem, which should respect the constraints given by relations (3) and (4).

Step 2. *Evaluation of the objective function F and of the auxiliary function f.* The problem contains an equality restriction shown by relation (3). Thus, the auxiliary function *f* is formed, using the relation:

$$f = F + \alpha \cdot F_p^2 \tag{11}$$

Where α is the penalty factor.

For each particle and each iteration the values of function f will be calculated, and by comparing them solutions *Pbest_i* and *Gbest* are selected. At the end of the optimization process, functions *f* and *F* will have approximately equal values, according to the calculation error admitted by choosing factor α.

Step 3. *Update velocity and solution.* The minimum (V_j^{min}) and the maximum limits (V_j^{max}) of the velocity for each generating unit j are calculated:

$$V_j^{max} = \beta \cdot (P_j^{max} - P_j^{min}) \text{ and } V_j^{min} = -V_j^{max} \tag{12}$$

Where factor β was considered between 0.05 and 0.1.

The update of the particle position and velocity is done with relations (6) and (7). For each solution (X_i) it is verified if the components x_{ij} satisfy the constrain (4). If the constrain is satisfied, then the

Unit	P^{min}[MW]	P^{max}[MW]	a[$/$MW^2]	b[$/MW]	c[$]
1	30	120	0.00875	18.24	750
2	50	160	0.00754	18.87	680
3	50	200	0.00310	19.05	650
4	100	300	0.00423	17.90	900

Table 1: Cost coefficients and limits of generated powers for a thermal power plant with four units (CS4)

Unit	P^{min}[MW]	P^{max}[MW]	a[$/$MW^2]	b[$/MW]	c[$]
1	100	600	0.001562	7.92	561
2	100	400	0.00194	7.85	310
3	50	200	0.00482	7.97	78
4	140	590	0.00139	7.06	500
5	110	440	0.00184	7.46	295
6	110	440	0.00184	7.46	295

Table 2: Cost coefficients and limits of generated powers for a thermal power plant with six units (CS6)

calculated value for x_{ij} is kept. Otherwise x_{ij} is set with the value nearest to the limit of the domain (P_j^{max} or P_j^{min}).

The vectors *Pbest* and *Gbest* are obtained based on the evaluation of the auxiliary function f and on the comparison of the f values calculated in two consecutive steps. If the new value of function f is better than the previous value of f for previous *Gbest*, then *Gbest* is set at the new value. Similarly, *Pbest* vector is updated.

Step 4. *Stopping process*. In this paper the criterion of stopping the calculation process is given through achieving the maximum number of iterations set.

5   Numerical examples and simulation results

In this section three case studies on how EPD solving by applying the two PSO techniques (PSO Classical and PSO Accelerated) are presented. The objective function is given by the relation (2) and restrictions (3) and (4). All case studies were implemented in Mathcad, on a personal computer having a 1.58 GHz processor and 896 MB of RAM.

5.1   Description of tested systems

Case study 1 - four unit system (CS4). The first system (CS4) is a thermal power plant having four generating units, where the total power losses (ΔP) are considered zero. The data for the four generators (cost coefficients and limits of generated powers) are presented in Table 1. The total power demanded in the system is P_D=520 MW.

Case study 2 - six unit system (CS6). The second system (CS6) is a thermal power plant with six generating units, where the total (ΔP) is considered zero. The data for the six generators (cost coefficients and limits of generated powers) are presented in Table 2 [5]. The total power demanded in the system is P_D=1800 MW.

Case study 3 - twenty unit system (CS20). The third system contains twenty units, and the demanded power in the system is P_D=2500MW. The data for the generators and the values of the coefficients B_{ij} are available in [9].

Scenarios	c_{1i}	c_{2f}	Best F [$/hour]	Worst F [$/hour]	Average F [$/hour]	Standard deviation F [$/hour]
1	1.5	1.5	12919.76	12920.77	12919.84	0.026
2	1.5	2	12919.76	12920.36	1291981	0.015
3	1.5	2.5	12919.76	12921.01	12919.84	0.035
4	2	1.5	12919.77	12921.00	12919.83	0.028
5	2	2	12919.76	12920.04	12919.79	0.007
6	2	2.5	12919.76	12920.42	12919.80	0.016
7	2.5	1.5	12919.76	12922.35	12919.84	0.052
8	2.5	2	12919.76	12920.17	12919.80	0.009
9	2.5	2.5	12919.76	12920.27	12919.80	0.013

Table 3: The influence of the coefficients c1i,c2f upon the results obtained through PSO Accelerated, for one hundred trials (CS4)

5.2   PSO parameters and PSO convergence

PSO parameters involved in the calculation process may affect the algorithm performances and quality of the solutions. For the system having four generators (CS4) the parameters were set to values: $W_{min} = 0.4, W_{max} = 1, c_1 = 2.75, c_2 = 1.75$, NP=6, $k_{max} = 15$ (PSO Classical) and respectively $W_{min} = 0.4$, $W_{max} = 1, c_{1i} = 2, c_{2i} = 0.4, c_{1f} = 0.4, c_{2f} = 2$, NP=6, $k_{max} = 15$ (PSO Accelerated).

In case of PSO Accelerated algorithm, in order to assess the influence of the coefficients on EPD problem solving, 100 distinct trials were performed, noting the best value for F (Best F), the worst value for F (worst F), the average value for F (Average F) and the standard deviation of cost F. Table 3 shows the values Best F, Worst F, Average F and standard deviation F, considering that the values of coefficients (c_{1i}, c_{2f}) vary between limits $c_{1i}, c_{2f} \in [1.5, 2.5]$, and the values of coefficient (c_{1f}) and (c_{2i}) are constant and equal to $c_{1f} = c_{2i} = 0.4$.

Regarding the four unit system (CS4), in Table 3 it can be noticed that coefficient changes do not affect the solutions, but the best results are obtained for scenario 5, where $c_{1i} = 2, c_{2f} = 2$.

For the six generator system (CS6) in both PSO Classical and PSO Accelerated algorithm, the coefficients have little influence upon the outcomes, considering that they vary within the limits c1,c2 [1, 4], respective c1i,c2f [1.5, 2.5] and c1f, c2i [0.1, 0.4]. The results are shown in Table 4, the parameters being set to values: wmin=0.4, wmax=1, c1=2, c2=2, NP=15, kmax=30 (PSO Classical), and wmin=0.4, wmax=0.9, c1i=2.5, c2i=0.2, c1f=0.4, c2f=1.6, NP=15, kmax=30 (PSO Accelerated).

In case of the twenty unit system (CS20), PSO Classical was applied taking into account the following settings: w1=0.4, w2=1.1, c1=2, c2=2, NP=500, kmax=200 and =0.1; for PSO Accelerated it was considered: w1=0.4, w2=1.1, c1i=2.5, c2f=2, c1f=0.4, c2i=0.2, NP=500, kmax=200 and =0.1.

The number of particles that constitute the population is another important factor in the PSO algorithm. In case of the four unit system, for both methods (PSO Classical and PSO Accelerated), the changes in cost F in relation to particles number (NP) were graphically represented, considering the number of iterations set at k_{max}=15 (Fig. 1). Also, in Fig. 2 the variation of cost F in relation to the number of iterations (k_{max}) is represented, considering the number of particles set to NP=10.

Analyzing the diagram shown in Fig. 1 it is found that for both algorithms (PSO Classical and PSO Accelerated) the best solution (Best F) is obtained considering a population consisting of 5-6 particles. Fig. 2 shows that the converging process towards the best value of F is obtained after fifteen iterations for both algorithms, the initial solution being different.

In the six unit system, increasing the number of variables involves a higher number of particles (NP=15) and a higher number of iterations to (k_{max}=30) in order to obtain the same solution, presented in Table 4.
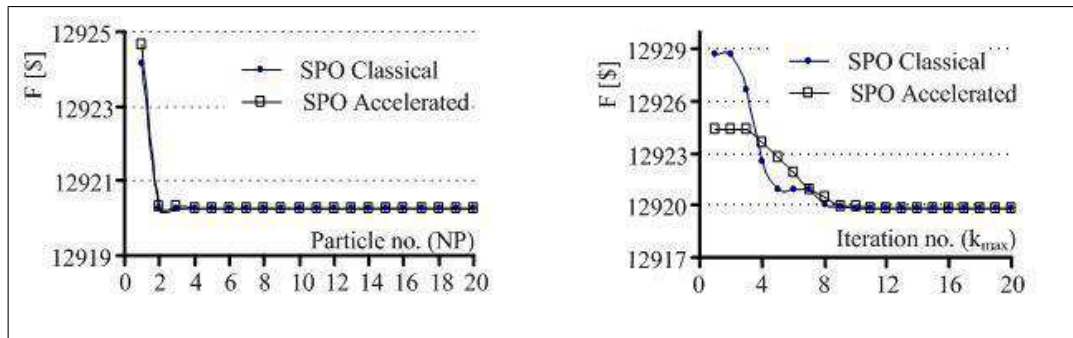
Figure 1: F convergence with particles number (NP)/CS4; Figure 2. F convergence with k_{max} for CS4

Algorithms	P1 [MW]	P2 [MW]	P3 [MW]	P4 [MW]	P5 [MW]	P6 [MW]	Cost F [$/hour]
FLCGA [3]	250.49	215.43	109.92	572.84	325.66	325.66	16585.85
PGA [5]	248.14	217.74	75.20	587.80	335.56	335.56	16579.33
AECGA [6]	248.07	217.73	75.30	587.70	335.60	335.60	16579.33
IHN [7]	248.08	217.74	75.18	587.90	335.55	335.55	16579.33
PSO Classical	247.95	218.44	75.16	587.58	335.43	335.44	16579.33
PSO Accelerated	248.00	217.71	75.16	588.02	335.52	335.59	16579.33

Table 4: The comparison of the results obtained for the system with six generating units (CS6)

To assess the efficiency of PSO algorithms, they are compared to other four algorithms previously presented using the same data, available in [5]. It can be seen that both PSO Classical, and PSO Accelerated algorithm reach the same cost F as the algorithms presented in [3], [5], [6] and [7]. The resulting solution and the cost value are comprised in Table 4.

5.3 The assessment of the solutions

The quality of the solutions was assessed by determining the values Best F, Worst F, Average F and standard deviation F considering 100 trials.

FFor the system with four generating units the results are presented in Table 5. In order to assess the convergence process and quality of the solutions the average of cost values F (Average F - Fig.3) and its standard deviation (Standard deviation F - Fig.4) for each iteration are determined, considering only one trial. The system considered in Fig.3 and Fig.4 proves that both algorithms converge quickly toward the best solution for F, the curves presenting a continuous decrease in relation to the number of iterations.

The best solution obtained in 100 trials, using algorithms PSO Classical and PSO Accelerated, is presented in Table 5, together with the results of the gradient method.

In case of the six units system (CS6), the values Best F, Worst F, Average F and standard deviation F considering 100 trials are presented in Table 6.

Algorithm robustness was tested starting from different initial solutions, randomly obtained, and retaining the best value of F for a number of algorithm trials within the [1-50] interval.

For the system having twenty units (CS20) the following values were obtained, considering 100 trials (Table 7). In Table 8 is presented the solution for the twenty units system, and the comparison of the results with those obtained in [2].
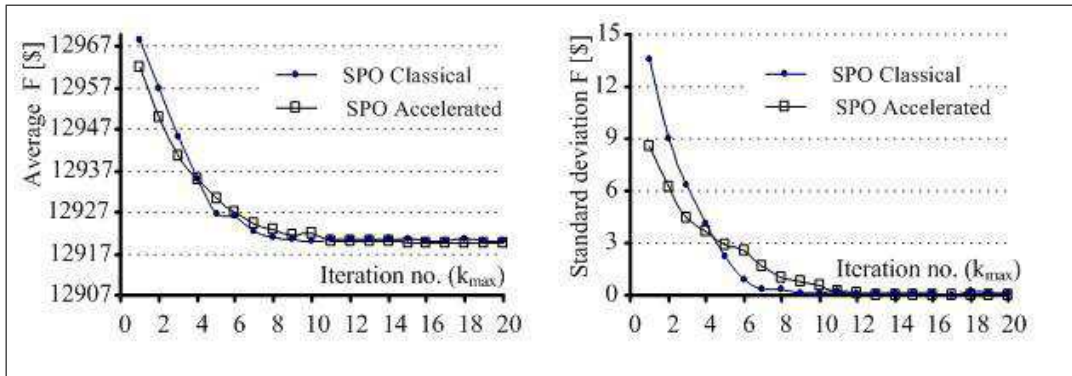
Figure 3. Average variation of *F* with k_{max} (CS4);   Figure 4. Variation of standard dev. *F* with k_{max}(CS4)

Unit power output [MW]	PSO Classical	PSO Accelerated	Gradient method
P1	88.554	92.536	92.493
P2	65.340	65.539	65.559
P3	134.662	130.293	130.431
P4	231.444	231.632	231.517
Total power [MW]	520.00	520.00	520.00
Total generated cost (Best *F*)[$/hour]	12919.96	12919.76	12919.76
CPU time (s)	< 1	< 1	< 1

Table 5: The best solutions (Best F) obtained for the system with four generating units (CS4)

Algorithm	Best *F* [$/hour]	Worst *F* [$/hour]	Average *F* [$/hour]	Standard deviation *F* [$/hour]
PSO Classical	16579.33	16582.64	16579.51	0.0650
PSO Accelerated	16579.33	16581.93	16579.49	0.0362

Table 6: The values Best *F*, Worst *F*, Average *F* and standard deviation *F* for CS6

Algorithm	Best *F* [$/hour]	Worst *F* [$/hour]	Average *F* [$/hour]	Standard deviation *F* [$/hour]
PSO Classical	62457.1805	62469.64	62462.45	0.4346
PSO Accelerated	62456.4380	62466.63	62461.05	0.3340

Table 7: The values Best *F*, Worst *F*, Average *F* and standard deviation *F* for CS20

The value of cost *F* obtained in Table 7 presents a smaller dispersion for both PSO algorithms, showing a high quality of the solutions. The calculation of power losses in the transmission lines was achieved with an error of 0.03 MW. It is also seen that PSO Accelerated algorithm gets a solution as good as the algorithms presented in [2], but PSO Classical leads to a farther solution, comparing to the solutions presented in Table 8.

Unit power output [MW]	PSO Accelerated	PSO Classical	Lambda-iteration method [2]	Hopfield neural network [2]
P1	511.1808	514.2219	512.7805	512.7804
P2	171.9583	163.3928	169.1033	169.1035
P3	125.9410	125.7172	126.8898	126.8897
P4	99.6666	99.4738	102.8657	102.8656
P5	114.9377	112.8359	113.6836	113.6836
P6	75.1378	76.8715	73.5710	73.5709
P7	113.2613	109.2679	115.2878	115.2876
P8	116.2341	116.8953	116.3994	106.3994
P9	101.5174	100.5633	100.4062	100.4063
P10	102.5556	113.8036	106.0267	106.0267
P11	150.6753	154.2216	150.2394	150.2395
P12	292.5836	289.6717	292.7648	292.7647
P13	120.2476	118.0356	119.1154	119.1155
P14	34.9866	38.2194	30.8340	30.8342
P15	117.3186	115.2359	115.8057	115.8056
P16	36.1563	36.4369	36.2545	36.2545
P17	68.7329	69.8707	66.8590	66.8590
P18	83.7616	81.9408	87.9720	87.9720
P19	98.9061	103.8655	100.8033	100.8033
P20	56.4137	51.6055	54.3050	54.3050
Total power [MW]	2592.1728	2592.1468	2591.9670	2591.9669
Total generation cost [$/hour]	62456.4380	62457.1805	62456.6391	62456.6341
Total lines losses [MW]	92.1728	92.1468	91.9670	91.9669

Table 8: The best solutions (Best *F*) obtained for the system with twenty generating units (CS20)

6   Conclusions

In this paper, the economic power dispatch problem is solved using two PSO techniques, namely, PSO Classical and PSO Accelerated. Both techniques are effective in solving this problem, but PSO Accelerated leads to a better quality of solutions and a lower computing time.

For EPD problems with small number of variables and linear restrictions, the classical solving techniques (gradient method, lambda iteration method) are also applicable, obtaining the same results as PSO. For EPD with nonlinear restrictions, PSO techniques are more effective, having a better convergence, robustness and stability, indicated by low values of standard deviation. The number of particles and the number of iterations required to obtain stable solutions are related to a reduced computing time.

PSO techniques are also compared with other techniques, such as Hopfield neural network, the results being almost identical for our applications.

Bibliography

[1] Lee, F.N. and Breipohl, A.M., *Reserve contrained economic dispatch with prohibited operating zones*, IEEE Transaction Power Systems, Vol. 8 (1), pp: 246-254, 1993

[2] Su, C.T. and Lin C.T., *New Approach with a Hopfield Modeling Framework to Economic Dispatch*, IEEE Transaction Power Systems, Vol. 15 (2), pp: 541-545, 2000

[3] Song, Y.H., Wang, G.S., Wang, P.V. and Johns, A.T., *Environmental/Economic Dispatch Using Fuzzy Logic Controlled Genetic Algorithms*, IEE Generation, Transmission and Distribution, Vol. 144(4), pp: 377-382, 1997

[4] Chiang C.L., *Improved Genetic Algorithm for Power Economic Dispatch of Units with Valve-Point Effects and Multiple Fuels*, IEEE Trans. Power Systems, Vol.20(4), pp: 1690-1699, 2005

[5] Yalcinoz, T., Altun, H. and Uzam, M., *Economic Dispatch Solution Using a Genetic Algorithm Based on Arithmetic Crossover*, IEEE Power Tech Conference, Vol.2, pp: 4, Porto, 2001

[6] Song, Y.H. and Chou C.S.V., *Advanced Engineered Conditioning Genetic Approach to Power Economic Dispatch*, IEE Generation, Transmission and Distribution, Vol.144(3),pp: 285-292, 1997

[7] Yalcinoz, T. and Short M.J., *Large scale Economic Dispatch Using an Improved Hopfield Neural Network*, IEE Generation, Transmission and Distribution, Vol. 144(2), pp: 181-185, 1997

[8] Coelho, Ld.S. and Mariani, V.C., *Combining of Chaotic Differential Evolution and Quadratic Programming for Economic Dispatch Optimization with Valve-Point Effect*, IEEE Transactions on Power Systems, Vol. 21(3), pp: 1465- 1465. 2006.

[9] Sinha, N., Chakrabarti, R. and Chattopadhyay P.K., *Evolutionary Programming Techniques For Economic Load Dispatch*, IEEE Transaction Evolutionary Computation, Vol.7(1), pp: 83-94, 2003

[10] Venkatesh, P., Gnanadass, R. and Padhy N.P, *Comparison And Application Of Evolutionary Programming Techniques To Combined Economic Emission Dispatch With Line Flow Constraints*, IEEE Transactions on Power Systems, Vol.18(2), pp: 688-697, 2003

[11] Singh, L. and Dhillon, J.S., *Fuzzy Satisfying Multiobjective Thermal Power Dispatch Based On Surrogate Worth Trade-Off Method, Electric Power Components And Systems*, Vol. 36(1), pp: 93-108, 2008

[12] Attaviriyanupap, P., Kita, H., Tanaka, E. and Hasegawa J., *A Fuzzy-Optimization Approach to Dynamic Economic Dispatch Considering uncertainties*, IEEE Trans. Power Systems, Vol.19(3), pp: 1299-1307, 2004

[13] Brar, Y.S., Dhillon, J.S. and Kothari, D.P., *Multiobjective Load Dispatch By Fuzzy Logic Searching Weightage Pattern, Electric Power Systems Research*, Vol. 63, pp: 149-160, 2002

[14] Lin, W.M., Cheng, F.S. and Tsay, M.T., *An Improved Tabu Search For Economic Dispatch With Multiple Minima*, IEEE Transaction Power Systems, 17(1), pp: 108-112, 2002

[15] AlRashidi, M.R. and El-Hawary M.E., *Hybrid Particle Swarm Optimization Approach for Solving the Discrete OPF Problem Considering the Valve Loading Effects*, IEEE Transaction Power Systems, Vol. 22 (4), pp: 2030-2038, 2007

[16] Gaing, Z.L., *Particle Swarm Optimization to Solving the Economic Dispatch Considering the Generator Constraints*, IEEE Transaction Power Systems, Vol. 18 (3), pp: 1187-1195, 2003

[17] Rotar, C., Dumitrescu, D. and Lung, R.I., *Guided hyperplane evolutionary algorithm, Proceedings of the 9th annual conference on Genetic and evolutionary computation*, London, pp: 884 - 891, 2007

[18] Yumbla, P.E.O., Ramirez, J.M. and Coello, C.A.C., *Optimal Power Flow Subject to Security Constraints Solved With a Particle Swarm Optimizer*, IEEE Transaction Power Systems, Vol. 23(1), pp: 33-40, 2008

[19] Makeechev, V.A., Soukhanov, O.A. and Sharov, Y.V., *Hierarchical Algorithms Of Functional Modelling For Solution Of Optimal Operation Problems in Electrical Power Systems*, International Journal of Electrical Power & Energy Systems, Vol. 30 (6-7), pp: 415-427, 2008

[20] Singh, L. and Dhillon J.S., *Secure Multiobjective Real And Reactive Power Allocation Of Thermal Power Units*, International Journal Of Electrical Power & Energy Systems, Vol. 30, pp: 594-602, 2008

[21] Ratnaweera, A., Halgamuge, S.K. and Watson, H,C., *Self-Organizing Hierarchical Particle Swarm Optimizer With Time Varying Acceleration Coefficients*, IEEE Transaction Evolutionary Computation, Vol. 8(3), pp: 240-255, 2004

[22] Kennedy, J.F., Eberhart, R.C. and Shi, R.C., *Swarm Intelligence*, San Francisco (CA, USA): Morgan Kaufmann Publishers, 2001.

[23] Yoshida, H., Kawata, K., Fukuyama, Y., Takayama, S. and Nakanishi, Y., *A Particle Swarm Optimization For Reactive Power And Voltage Control Considering Voltage Security Assessment*, Transactions of the Institute of Electrical Engineers of Japan. B, Vol. 119-B, 12, pp: 1462-1469, 1999

[24] Gavrilas, M., Iovanov, O. and Sfintes, C.V., *Enhanced Particle Swarm Optimization Method For Power Loss Reduction In Distribution Systems*, 19th International Conference on Electricity Distribution, Vienna, paper 0088 (4 pp), 2007

[25] Liu, Y. and Gu X., *Skeleton-Network Reconfiguration Based on Topological Characteristics of Scale-Free Networks and Discrete Particle Swarm Optimization*, IEEE Transaction Power Systems, Vol. 22 (3), pp: 1267-1274, 2007

[26] Ting, T.O., Rao, M.V.C. and Loo, C.K., *A Novel Approach for Unit Commitment problem via an Effective Hybrid Particle Swarm Optimization*, IEEE Transaction Power Systems, Vol. 21 (1), pp: 411-417, 2006

[27] Krishna Teerth Chaturvedi, Manjaree Pandit, Laxmi Srivastava, *Particle swarm optimization with time varying acceleration coefficients for non-convex economic power dispatch*, International Journal of Electrical Power & Energy Systems, Vol. 31 (6), pp: 249-257, 2009

[28] Leandro dos Santos Coelho, Chu-Sheng Lee, *Solving economic load dispatch problems in power systems using chaotic and Gaussian particle swarm optimization approaches*, International Journal of Electrical Power & Energy Systems, Vol. 30 (5), pp: 297-307, 2008