

Economic Dispatch Problem Based on Improved Particle Swarm Optimization

Peifeng Wu^{1,2,*}, Wenxia Liu¹ and Jianhua Zhang¹

¹School of Electrical and Electronic Engineering, North China Electric Power University, Beijing 102206

²College Of Electrical Engineering, Liaoning University of Technology, Jinzhou, 121000

Received 15 June 2013; Accepted 23 January 2014

Abstract

An improved particle swarm optimization (IPSO) is used to solve economic dispatch problem (EDP). The IPSO has two position updating strategies. In the early stage of iteration, the individual in the population updates the position according to its own best experience with a large probability. In the later stage of iteration, the individual updates the position according to the best experience in the population with a large probability. In addition, the IPSO introduces a mutation operator after position updating, which can increase the diversity of population and prevent the premature convergence. IPSO has been used to solve EDP with valve point effect. The experimental results show that IPSO is an effective algorithm to solve EDP.

Keywords: Improved Particle Swarm Optimization, Economic Dispatch Problem, Position Updating Strategies, Valve Point Effect

1. Introduction

Economic dispatch problem (EDP) is an important optimization problem in modern power system. The objective of EDP is to schedule the generating unit outputs so as to meet the total load in the system at minimum operating cost. EDP can be represented as a complicated optimization problem with many constraints.

Recently, many optimization algorithms are developed to solve EDP. Lin has proposed a novel tabu search which combines EP and quadratic programming methods to solve EDP [1]. Gaing has used an interesting particle swarm optimization algorithm to solve EDP with generator constraints [2]. Su applied the Hopfield network to solve EDP with prohibited operating zones [3]. Sewtohu utilized genetic algorithm (GA) to solve EDP taking account of valve-point effect [4].

Particle swarm optimization (PSO) has been proposed by Kennedy and Eberhart in [5]. PSO simulates the population behavior of birds and is easy to implement. The algorithm has better robustness than many algorithms such as GA, is less sensitive to the nature of the problems and can be used for function optimization [6], control system [7], parameters optimization [8] and neural network [9].

Recently there are many improved particle swarm optimization to enhance the search ability of the algorithm. The comprehensive learning particle swarm optimizer (CLPSO) has been proposed in [6]. CLPSO introduces a novel velocity updating strategy which learns the experience from other individual in the population. Except for the best experience, the better individual also can guide the search procedure. Example-based learning particle swarm

optimization (ELPSO) has been proposed in [10]. ELPSO has the ability to increase the diversity of population during iteration and can improve the performance of PSO. Wang [11] has proposed a novel PSO which is based on chaotic search. If the best solution in the population does not change for long, the method generates a random particle and does chaotic search. The method can avoid the local optima during iteration.

This paper uses improved particle swarm optimization (IPSO) to solve EDP. The IPSO has two different position updating strategies which can increase the diversity of population in the early stage of iteration and accelerate the convergence speed in the later stage of iteration. EDPs for test systems consisting of 13 and 40 thermal units with valve-point effect are used to test the IPSO. The results obtained by IPSO are analyzed and compared with those obtained by other methods.

The remainder of this paper is organized as follows. In Section 2, basic PSO and some other versions of PSO are briefly introduced. In Section 3, IPSO algorithm is introduced. In Section 4, the mathematic model of EDP is introduced. In Section 5, many methods are used to solve EDP and the experimental results are analyzed. Conclusion is summarized in Section 6.

2. Basic Particle Swarm Optimization and Other Versions of Particle Swarm Optimization

2.1 Particle Swarm Optimization

Basic PSO is a population based optimization algorithm. This algorithm simulates the behavior of bird flock to search optima in searching area. The algorithm is initialized with a population of random particles and the whole search procedure can be summarized as velocity updating formula and direction updating equation.

* E-mail address: wupeifengswcd@163.com

In the algorithm, $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$ and $V_i = (v_{i1}, v_{i2}, \dots, v_{in})$ represent the direction vector and velocity vector of the i -th particle, respectively. At each generation, the particles in population update their directions and velocities according to the following two equations:

$$v_{id}^{(t+1)} = \omega v_{id}^{(t)} + c_1 r_1 (p_{id}^{(t)} - x_{id}^{(t)}) + c_2 r_2 (p_{gd}^{(t)} - x_{id}^{(t)}) \quad (1)$$

$$x_{id}^{(t+1)} = x_{id}^{(t)} + v_{id}^{(t+1)} \quad (2)$$

where $p_{id}^{(t)}$ is the best solution obtained by the i -th particle. $p_{gd}^{(t)}$ is the best solution in the whole population. c_1 and c_2 are positive constant parameters called learning factors. r_1 and r_2 are independently and uniformly distributed random number with range (0,1). ω is called inertia weight.

This factor can balance the global search ability and local search ability. The global search ability can be enhanced with the increasing of ω .

2.2 Particle Swarm Optimization

Kennedy [12] proposed a novel algorithm which is named bare bones particle swarm optimization (BBPSO).

BBPSO eliminates the velocity updating equation and position updating equation. The individual in the population are all sampled from Gaussian distribution which is built on the global best (gbest) and the personal best (pbest). The above process is as follows:

$$x_{i,j} = N(\mu_{i,j}, \sigma_{i,j}^2) \quad (3)$$

Where N represents a Gaussian distribution with mean $\mu_{i,j} = (gbest_j + pbest_{i,j}) / 2$ and standard deviation $\sigma_{i,j} = |gbest_j - pbest_{i,j}|$

2.3 Comprehensive Learning Particle Swarm Optimization (CLPSO)

Laing [6] proposed a novel particle swarm optimization which is named CLPSO. CLPSO introduced a different velocity updating strategy which can increase the diversity. In the velocity updating process of traditional PSO, individual will search best solution according to its own best experience or the best experience in the population. CLPSO chooses the best experience of each individual in population randomly. The velocity updating process is as follows:

$$v_{id}^{(t+1)} = \omega v_{id}^{(t)} + c_1 r_1 (pbest_{rid}^{(t)} - x_{id}^{(t)}) \quad (4)$$

where $pbest_{rid}^{(t)}$ is the selected best solution which is used to guide search process.

3. IPSO

IPSO introduces two position updating strategies and a mutation operation so as to improve performance of PSO. The main procedures of IPSO are as follows:

Step 1. Initialize the algorithm parameters of IPSO.

Some parameters of IPSO should be initialized in this step. The parameters are the population size PS; decision probability γ which helps IPSO to choose different position updating strategies; mutation probability pm and the number of iterations K.

Step 2. Initialize population

The population is initialized in the search area $[x_{iL}, x_{iU}] (i = 1, 2, K, D)$ randomly. x_{iL} and x_{iU} are lower and upper bounds for the i -th component. D is the dimension size of problem.

Step 3. Update position

Basic PSO has two main steps. They are velocity updating and position updating. In IPSO, the algorithm excludes the velocity updating and designs two novel position updating.

The main procedures of position updating are as follows:

In order to increase the diversity of population, the individual may update its position according to its own best experience with a large probability in the early stage of iteration. In order to accelerate the convergence speed of the algorithm, the individual may update its position according to the best experience in the population with a large probability in the later stage of iteration. Based on the above illustration, the two position updating strategies can be formulated as follows:

$$x_{id}^{(t+1)} = \begin{cases} x_{id}^{(t)} + 2 \times rand1 \times (p_{id} - x_{id}^{(t)}) & \text{if } rand() < \gamma \\ x_{id}^{(t)} + 2 \times rand2 \times (p_{gd} - x_{id}^{(t)}) & \text{else} \end{cases} \quad (5)$$

Where $rand1$ and $rand2$ are the uniformly generated random numbers in the range of [0, 1]; p_{id} is the d th component of i th individual's best solution; p_{gd} is the d th component of the best solution in the whole population; γ is defined as decision probability, and it is designed to be equal to $\gamma = \sqrt{1 - k/K}$. If γ is satisfied, the individual will choose the first position updating strategies. The new position component $x_{id}^{(t+1)}$ will be located at a random position in the area $(p_{id} - |p_{id} - x_{id}^k|, p_{id} + |p_{id} - x_{id}^k|)$. Otherwise, the new position component will be located at a random position in the area $(p_{gd} - |p_{gd} - x_{id}^k|, p_{gd} + |p_{gd} - x_{id}^k|)$. The dynamically adjusted γ is depicted in Fig.1. As can be seen from Fig. 1, decision probability γ decreases slowly in the early stage of iteration. The individual will search around its own best solution p_{id} . In the later stage of iteration, decision probability γ decreases quickly and the individual will search around the best solution of population p_{gd} .

Step 4. Mutate

For most optimization algorithms, the diversity of population will decrease with iteration. In order to increase the diversity of population and avoid local optima, IPSO introduces mutation operator. After updating the position, IPSO carries out mutation operator. The mutation procedure is as follow:

$$x_{id}^{(t+1)} = x_{iL} + rand() \times (x_{iU} - x_{iL}) \tag{6}$$

Where $rand()$ is the uniformly generated random numbers in the range of $[0, 1]$; x_{iL} and x_{iU} are lower and upper limit of search space, respectively.

Step 5. Check the stopping criterion.

If the maximal number of iteration is reached then computation is terminated, otherwise, $k = k + 1$ and go to Step 3.

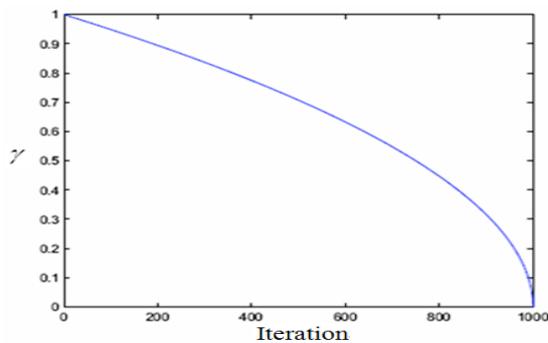


Fig.1 Decision probability γ

4. Mathematical Formulation for EDP

The EDP is used to determine the optimal schedule of the generating units while satisfying various constraints and meeting the load demand of a power system. EDP has many kinds of optimization objective. In electronic industry, fuel cost is still the core issue of power system. Thus, the model used for EDP in this paper is an optimal problem with single objective based on fuel cost. The objective function is as follows:

$$\min F = \sum_{i=1}^M F_i(P_i) \tag{7}$$

where M is the number of generating units, F_i is the fuel cost of the i -th generating unit. During the practical generating process, opening the valve of steam turbines will produce pulse in the heat rate curve of the generator. The above phenomenon is named valve-point effects. The accurate fuel cost can be expressed as follows:

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i + |e_i \sin(f_i(P_i^{\min} - P_i))| \tag{8}$$

where a_i, b_i and c_i are cost coefficients of i th generating unit. e_i and f_i are valve-point effect coefficients of i th generating unit.

EDP has equality and inequality constraints. The two constraints are as follows:

$$\sum_{i=1}^M P_i = P_D + P_L \tag{9}$$

$$P_i^{\min} \leq P_i \leq P_i^{\max} \tag{10}$$

where P_i is the power outputted by generator i ; P_i^{\min} and P_i^{\max} are lower and upper limit of the power outputted by generator i , respectively; P_D is the system load demand and P_L is the transmission loss. The equality constraint is real power balance constraint. The model used in this paper has neglected the transmission loss in system, and let $P_L = 0$. The equality constraint shows that the power should be equal to the summation of transmission loss and load demand. The inequality constraint is volume constraint. The constraint shows that power of each generator should lie between upper and lower limits.

5. Numerical Simulations and Results

To assess the effect of IPSO, this paper utilizes IPSO to solve EDP of power generations with 13 and 40 thermal units. The system data are shown in Table 1-2. Except for IPSO, this paper also uses PSO, BBPSO to solve EDP. Three algorithms are run 30 times, and the parameters setting of the three algorithms are as follows:

PSO[5]: $c_1 = c_2 = 2, PS=100, K = 200$.

BBPSO[12]: $PS=100, K = 200$

IPSO: $PS=100, K = 200, F_u = 0.9, p_m = 0.01$

In order to make the comparison fair, the best, worst, average and mean of the fuel costs are recorded in the experiments. The results are shown in Table 3

Table 1. Data for the 13 thermal units.

Unit	a_i	b_i	c_i	e_i	f_i	P_i^{\min}	P_i^{\max}
1	0.00028	8.10	550	300	0.035	0	680
2	0.00056	8.10	309	200	0.042	0	360
3	0.00056	8.10	307	150	0.042	0	360
4	0.00324	7.74	240	150	0.063	60	180
5	0.00324	7.74	240	150	0.063	60	180
6	0.00324	7.74	240	150	0.063	60	180
7	0.00324	7.74	240	150	0.063	60	180
8	0.00324	7.74	240	150	0.063	60	180
9	0.00324	7.74	240	150	0.063	60	180
10	0.00284	8.60	126	100	0.084	40	120
11	0.00284	8.60	126	100	0.084	40	120
12	0.00284	8.60	126	100	0.084	55	120
13	0.00284	8.60	126	100	0.084	55	120

Table 2. Data for the 40 thermal units

Unit	a_i	b_i	c_i	e_i	f_i	P_i^{\min}	P_i^{\max}
1	0.00690	6.73	94.705	100	0.084	36	114
2	0.00690	6.73	94.705	100	0.084	36	114
3	0.02028	7.07	309.54	100	0.084	60	120
4	0.00942	8.18	369.03	150	0.063	80	190
5	0.01140	5.35	148.89	120	0.077	47	97
6	0.01142	8.05	222.33	100	0.084	68	140
7	0.00357	8.03	287.71	200	0.042	110	300
8	0.00492	6.99	391.98	200	0.042	135	300
9	0.00573	6.60	455.76	200	0.042	135	300
10	0.00606	12.9	722.82	200	0.042	130	300
11	0.00515	12.9	635.20	200	0.042	94	375
12	0.00569	12.8	654.69	200	0.042	94	375
13	0.00421	12.5	913.40	300	0.035	125	500
14	0.00752	8.84	1760.4	300	0.035	125	500
15	0.00708	9.15	1728.3	300	0.035	125	500
16	0.00708	9.15	1728.3	300	0.035	125	500
17	0.00313	7.97	647.85	300	0.035	220	500
18	0.00313	7.95	649.69	300	0.035	220	500
19	0.00313	7.97	647.83	300	0.035	242	550
20	0.00313	7.97	647.81	300	0.035	242	550
21	0.00298	6.63	785.96	300	0.035	254	550
22	0.00298	6.63	785.96	300	0.035	254	550
23	0.00284	6.66	794.53	300	0.035	254	550
24	0.00284	6.66	794.53	300	0.035	254	550
25	0.00277	7.10	801.32	300	0.035	254	550
26	0.00277	7.10	801.32	300	0.035	254	550
27	0.52124	3.33	1055.1	120	0.077	10	150
28	0.52124	3.33	1055.1	120	0.077	10	150
29	0.52124	3.33	1055.1	120	0.077	10	150
30	0.01140	5.35	148.89	120	0.077	47	97
31	0.00160	6.43	222.92	150	0.063	60	190
32	0.00160	6.43	222.92	150	0.063	60	190
33	0.00160	6.43	222.92	150	0.063	60	190
34	0.0001	8.95	107.87	200	0.042	90	200
35	0.0001	8.62	116.58	200	0.042	90	200
36	0.0001	8.62	116.58	200	0.042	90	200
37	0.0161	5.88	307.45	80	0.098	25	110
38	0.0161	5.88	307.45	80	0.098	25	110
39	0.0161	5.88	307.45	80	0.098	25	110
40	0.00313	7.97	647.83	300	0.035	242	550

Table 3. optimization results of power system economic dispatch

Case	Method	Best	Worst	Mean
13 thermal units	PSO	2.41985e+004	2.45378e+004	2.42650e+004
	BBPSO	2.42084e+004	2.47897e+004	2.45073e+004
	IPSO	2.41651e+004	2.43295e+004	2.43859e+004
40 thermal units	PSO	1.22526e+005	1.24327e+005	1.23896e+005
	BBPSO	1.23987e+005	1.26763e+005	1.25768e+005
	IPSO	1.21470e+005	1.23643e+005	1.22760e+005

For the EDP of two test systems, the best fuel costs of IPSO are 2.416517e+004 and 1.214708e+005, respectively. The best fuel costs of BBPSO for the two test systems are 2.420841e+004 and 1.214708e+005, respectively. The best

fuel costs of PSO for the two test systems are 2.419856e+004 and 1.225269e+005, respectively. The best fuel costs of IPSO are smaller than other two algorithms. The comparison of best fuel costs shows that IPSO has the best search ability among the three algorithms. The mean fuel costs of IPSO for the two test systems are 2.438595e+004 and 1.227608e+005, respectively. The mean fuel costs of IPSO is still smaller than other two algorithms. The comparison of mean fuel costs demonstrates that the IPSO has the best robustness among the three algorithms. The dispatch results of EDADE for EDP are shown in Table 4-5.

The convergence properties of the PSO, BBPSO and IPSO in the process of searching for the minimum fuel costs are shown in Fig. 2-3. The two figures show that the convergence property of the IPSO algorithm is superior to that of the other two algorithms and the optimization results of IPSO are better than those of PSO and BBPSO.

Table 4. Dispatch result for 13 thermal power units ($P_D = 2520$)

Power	Generation
p1	628.311181
p2	299.190722
p3	294.904398
p4	159.667638
p5	159.738362
p6	159.736448
p7	159.662628
p8	159.698629
p9	159.690188
p10	77.3169799
p11	77.3605823
p12	92.3953238
p13	92.3269150

Table 5. Dispatch result for 40 thermal power units $P_D = 10500$

Power	Generation	Power	Generation
p1	114	p21	523.289324
p2	111.759982	p22	523.300897
p3	97.409465	p23	523.299391
p4	179.735354	p24	523.307431
p5	96.998156	p25	523.333278
p6	139.989350	p26	523.291080
p7	259.632230	p27	10.021260
p8	284.654156	p28	10.000334
p9	285.055364	p29	10.006793
p10	130	p30	96.999999
p11	168.802053	p31	190
p12	168.808478	p32	190
p13	214.782457	p33	190
p14	394.281798	p34	165.679215
p15	304.569337	p35	199.996662
p16	304.520329	p36	200
p17	489.283963	p37	109.999999
p18	489.283402	p38	110
p19	511.284438	p39	110
p20	511.326861	p40	511.297147

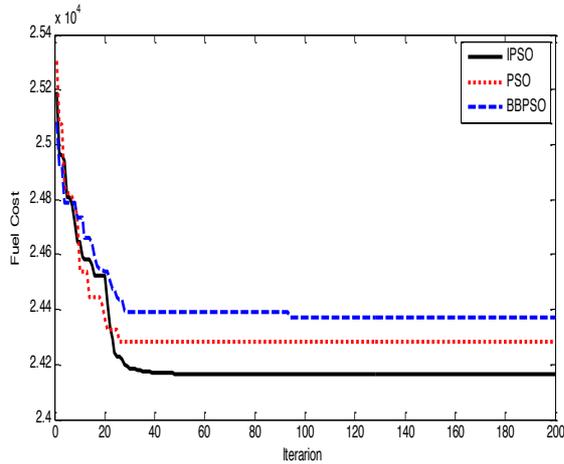


Fig. 2. Convergence characteristics for the three algorithms

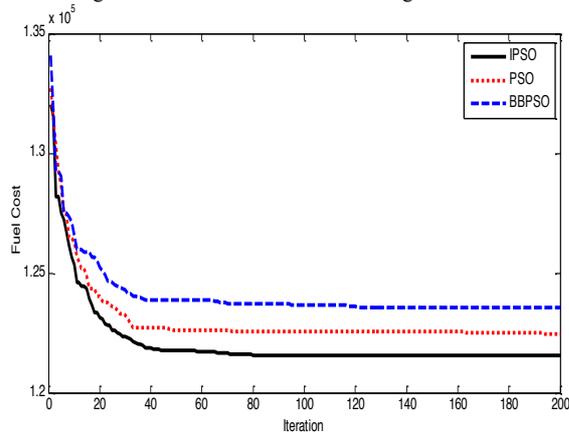


Fig. 3. Convergence characteristics for the three algorithms

After comparing the dispatch results of IPSO to other two PSO versions, this paper compares the results of IPSO with other studies reported in the literature. Tables 6-7 show the results.

Tab. 6 Comparison of dispatch result for 13 thermal power units ($P_D = 2520$)

Method	Fuel Cost
FAMPSO[13]	24169.9176
FAPSO-NM[14]	24169.92
FAPSO[14]	24170.93
PSO[14]	24262.73
GA-SA[15]	24275.71
HGA[15]	24169.92
ESO[16]	24179.59
EP-PSO[16]	24266.44
PSO-SQP[16]	24261.05
GA[15]	24398.23
SA[15]	24970.91
TS[17]	24178.65
MTS[17]	24169.89
EP-SQP[18]	24266.44
EDSA[19]	24169.92
IPSO	24165.17

Tab.7 Comparison of dispatch result for 40 thermal power units ($P_D = 10500$)

Method	Fuel Cost
FAPSO[14]	121712.4
MPSO[15]	122252.27
DEC-SQP[15]	122174.16
NPSO[20]	121704.74
PSO-SQP[15]	122094.67
HDE[15]	121698.51
PSO[15]	123930.45
IFEP[15]	122624.35
EP-SQP[15]	122323.97
MFEP[15]	122647.57
SOHPSO[21]	121501.14
SA[15]	122946.77
GA[14]	122508.88
TS[17]	122432.28
MTS[19]	121532.41
IFEP[15]	122624.35
FEP[15]	122679.71
TM[19]	122477.78
IPSO	121470.81

Table 6-7 show that the fuel costs of EDAD for the above three systems are 24165.17 and 121470.81, respectively. The tables demonstrate that the best fuel cost reported in this paper using IPSO is smaller than recent studies presented in other literatures.

6. Conclusion

EDP is a classical optimization problem in power system which has equality and inequality constraints. Many methods are used to solve the EDP. Most methods may have disadvantages such as getting trapped in local minima. PSO owns strong search ability and can be used to solve most difficult optimization problems.

This paper proposes the IPSO by introducing two novel updating position strategies. IPSO can increase the diversity of population in the early stage of iteration and accelerate the convergence speed in the later stage of iteration.

This paper uses IPSO and many other methods to solve economic dispatch problem with valve-point effect. Numerical results reveal that IPSO can obtain smaller fuel cost than other two algorithms and the methods of recent literatures. Experimental results demonstrate that IPSO is an effective method to solve economic dispatch problem.

References

1. Mwinyiwiwa, B.; Wolanski, Z.; Boon-Teck Oo, “ Microprocessor-implemented SPWM for multiconverters with phase-shifted triangle carriers”, *Industry Applications, IEEE Transactions*, 34(3), 1998, pp.487-494
2. Meiqin, M., Jianhui, S., Chang, L., Guorong, Z., Yuzhu, Z., “Controller for 1kW-5kW wind-solar hybrid generation system”, *Proc. Electrical and Computer Engineering, 2008. CCECE 2008. Canadian Conference on*, May 2008, pp. 1175-1178.
3. Li, R.T.H., Chung, H.S.-H., Chan, T.K.M., “An active modulation technique for single-phase grid-connected CSI”, *IEEE Transactions on Power Electronics* 22 (4) , 2007, pp. 1373-1382.
4. Chen, Y.-M., Cheng, Y.-M., “Amplitude distortion compensation for voltage source dc/ac inverters”, *International Journal of Electronics* 91 (2), 2004, pp. 83-96.
5. Nabae, A., Ogasawara, S., and Akagi, H., “A novel control scheme for PWM controlled inverters”, *Proceedings of IEEE Industrial Applications Society Annual Meeting*, Toronto, Canada, 1985, pp. 473-478.
6. HOLTZ, J., “Pulsewidth modulation – a survey”, *IEEE Transactions on Industrial Electronics*, 39, 1992, pp. 410-420.