Unit Commitment by Binary Particle Swarm Optimization

LIU YONG, HOU ZHI-JIAN, JIANG CHUAN-WEN
Department of Electrical Engineering
Shanghai Jiaotong University
Shanghai, 200030
CHINA

Abstract: - A solution to unit commitment using binary particle swarm optimization (BPSO) is presented. The minimum up and down time constraints, start-up and shutdown cost, spinning reserve, and generation limit are taken into account. The minimum up and down time constraints are considered in generating the particles to narrow the search space. Penalty factors are introduced to calculate the fitness of particles, which tend to avoid infeasible combinations. Problem formulation, representation and the simulation results are presented. The results show that the proposed method is effective.

Key-Words: - unit commitment, power system, binary particle swarm optimization, economic dispatch, penalty factors

1 Introduction

Unit commitment (UC) in power systems involves determining a start-up and shutdown schedule of units to meet the forecasted demand over a short term period [1]. The committed units must meet the system forecasted demand and spinning reserve requirement at minimum operating cost, subject to a large set of operating constraints. Hence, the UC problem is quite difficult due to its inherent high-dimensional, non-convex, and non-linear nature. The UC problem can be considered as two linked optimization problems, namely the unit-scheduled problem, which is a combinatorial optimization problem, and the economic dispatch (ED) problem, which is a non-linear programming optimization problem. The solution of the former must satisfy the system capacity requirements, generation limits, and the constraints on start-up and shut-down of the scheduled units during each planning period. The solution of the latter must perform the optimal generation dispatch among the operating units during each specific period of operation to satisfy the system load demand and spinning reserve capacity.

The exact optimal solution can be obtained by a complete enumeration, which cannot be applied to realistic power systems due to its excessive computation time requirements. To solve the unit commitment problem, some optimization techniques are applied to it. For example, there are priority list (PL) [2-3], dynamic programming (DP) [4-6], and Lagrangian relaxation [7-9]. PL methods are very fast but they are highly heuristic and give schedules with relatively high production costs. DP methods are prone to cause the curse of dimensionality. LR methods have problems in modeling plant crew constraints since they introduce coupling. In addition, artificial intelligence methods such as genetic algorithms (GA) and simulated annealing (SA) have been successfully used to solve UC problem [10-13].

Kennedy and Eberhart presented a new evolutionary computation algorithm, the particle swarm optimization (PSO), in 1995 [14]. It is a stochastic optimization technique that simulates the behavior of a flock of birds or the sociological behavior of a group of people. Zwe-Lee Gaing has used it to solve the UC problem [15]. However, the initialization of the particles in the presented method was time-wasted; as well the results were incorrect because it miscalculated the start-up costs.

In this paper, binary particle swarm optimization (BPSO) algorithm is used to solve the UC problem. The formulation of the UC problem is listed in section 2, including spinning reserve, minimum up and down time, and generation limit. The BPSO is described in section 3. The application of BPSO to the UC problem is demonstrated in section 4. Minimum up/down time constraints are considered in producing the particles as well as penalty coefficients are introduced into the evaluation function to avoid infeasible particles. Simulated results in section 5 indicate the efficiency of the methodology, and the conclusions are made in section 6.

2 Problem Formulation

The general problem formulation of unit

7th WSEAS Int. Conf. on MATHEMATICAL METHODS and COMPUTATIONAL TECHNIQUES IN ELECTRICAL ENGINEERING, Sofia, 27-29/10/05 (pp372-377)

commitment is given as follows.
Objective function

$$\min F = \sum_{j=1}^{T}\sum_{i=1}^{N}[(a_i + b_i P_{ij} + c_i P_{ij}^2)]u_{ij} + \sum_{j=1}^{T}\sum_{i=1}^{N}[S_{ij}u_{ij}(1-u_{i(j-1)}) + D_{ij}u_{i(j-1)}(1-u_{ij})] \quad (1)$$

Subject to
(a) System power balance

$$\sum_{i=1}^{N} P_{ij}u_{ij} - P_{Dj} = 0 \quad (2)$$

(b) Spinning reserve requirement

$$\sum_{i=1}^{N} P_i^{\max} u_{ij} \geq P_{Dj} + P_{Rj} \quad (3)$$

(c) Generation limit

$$P_i^{\min} \leq P_{ij} \leq P_i^{\max} \quad (4)$$

(d) Minimum up/down time

$$T_{ij}^{ON} > MUT_i \quad (5)$$

$$T_{ij}^{OFF} > MDT_i \quad (6)$$

where

N number of units,
T scheduling period in hours,
P_{ij} generation of unit *i* for hour *j*,
a_i, b_i, c_i fuel cost coefficients of unit *i*,
u_{ij} on(1)/off(0) status of unit *i* at time *j*,
S_{ij} start-up cost of unit *i* at time *j*, where $S_{ij} = \sigma_i + \delta_i(1 - e^{-T_{ij}^{OFF}/\tau_i})$, and σ_i, δ_i, τ_i are start-up cost coefficients of unit *i*,
D_{ij} shutdown cost of unit *i* at time *j*,
P_{Dj} system load demand at time *j*,
P_{Rj} system spinning reserve required at time *j*,
$P_i^{\min}$ minimum generation limit of unit *i*,
$P_i^{\max}$ maximum generation limit of unit *i*,
T_{ij}^{ON} ON period of unit *i* at time *j*,
T_{ij}^{OFF} OFF period of unit *i* at time *j*,
MUT_i minimum up time of unit *i*,
MDT_i minimum down time of unit *i*,

3 Binary Particle Swarm Optimization (BPSO)

Kennedy and Eberhart first introduced the particle swarm optimization (PSO) method, which is an evolutionary computation technique. Similar to genetic algorithms (GA), PSO is a population based optimization tool. The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, are "flown" through the problem space by following the current optimum particles. Compared to GA, the advantages of PSO are that PSO is easy to implement and there are few parameters to adjust. Therefore, PSO has been successfully applied in many areas.

Each individual in PSO flies in the search space with a velocity which is dynamically adjusted according to its own flying experience and its companions' flying experience. Each individual keeps track of its coordinates in the problem space, which are associated with the best solution (fitness) it has achieved so far. This value is called *pbest*. Another best value that is tracked by the global version of the particle swarm optimizer is the overall best value, and its location, obtained so far by any particle in the population. This location is called *gbest*. At each time step, the particle swarm optimization concept consists of velocity changes of each particle toward its *pbest* and *gbest* locations. Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward *pbest* and *gbest* locations.

If $x_i = (x_{i1}, x_{i2}, \cdots, x_{iD})$ represent the *i*th particle in the D-dimensional space, the binary version of PSO can be formulated as follows [16].

$$v_{id}^{k+1} = w \cdot v_{id}^{k} + c_1 \cdot rand() \cdot (pbest_{id} - x_{id}^{k}) + c_2 \cdot rand() \cdot (gbest_d - x_{id}^{k}) \quad (7)$$

$$x_{id}^{k+1} = \begin{cases} 1 & rand() < S(v_{id}^{k+1}) \\ 0 & otherwise \end{cases} \quad (8)$$

where

v_{id}^k velocity of individual *i* at iteration *k*, $v^{\min} \leq v_i^k \leq v^{\max}$,
w inertia weight factor, often decrease linearly from about 0.9 to 0.4 during a run [17]. $w = w_{\max} - \frac{w_{\max} - w_{\min}}{iter_{\max}} \times iter$.
c_1, c_2 acceleration constant, often set to be 2,
rand() uniform random number between 0 and 1,
x_{id}^k current position of individual *i* at iteration *k*,
$pbest_i$ *pbest* of individual *i*,
gbest *gbest* of the group,

$S(v)$ a sigmoid limiting transformation function, $S(v) = 1/(1+e^{-v})$.

4 Solution Methodology

A proposed binary particle swarm optimization (BPSO) method is proposed in the paper to solve the UC problem. Two modifications are made to the solution. One is using a new method to generate the particles, which insures the particles satisfy minimum up and down time constraints. The other is introducing penalty factors to avoid infeasible individuals. These modifications will prevent random generation and test feasibility step in [15].

4.1 Representation strategy

Before using the BPSO algorithm to solve the UC problem, the representation of a particle must be defined. A particle is also called an individual. Similar to GA, we can define each generator's status as a gene, all available generators' status at each schedule time make up a sub-chromosome, all sub-chromosomes in the scheduling period comprising an individual as shown in Fig.1. For example, for a 10-unit system and 24-hour scheduling period the dimension of an individual is 10*24=240.

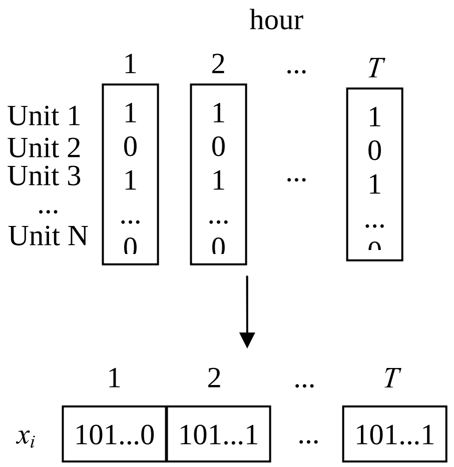

Fig.1 Representation of unit commiment solution

Suppose t_{ij} is the unit status variable which denotes the continuous on/off time of unit i at time j, then

$$u_{ij} = \begin{cases} 1 & t_{ij} < MUT_i \ \& \ u_{i(j-1)} = 1 \\ 0 & t_{ij} < MDT_i \ \& \ u_{i(j-1)} = 0 \\ 0 \text{ or } 1 & \text{otherwise} \end{cases} \quad (9)$$

This means that the status of the units will be determined by the minimum up/down time constraints of the units at first, and then determined by BPSO. Hence, minimum up and down time constraints can be considered when initializing or modifying the individuals. The individual i in the BPSO would be presented as

$$\begin{aligned} x_i &= (x_{i1}, x_{i2}, \cdots, x_{i(N*T)}) \\ &= (u_{11}, u_{21}, \cdots u_{N1}, u_{12}, u_{22}, \cdots u_{N2}, \cdots u_{NT}) \end{aligned} \quad (10)$$

4.2 Evaluation function

The evaluation function is mainly used to provide a measure of how the individual performed in the problem domain. The best individual should have the lowest total generation cost of objective function, and also satisfy system constraints of the UC problem. Therefore, in the BPSO algorithm, we define the evaluation function as

$$\begin{aligned} f &= 1/(F + \sum_{j=1}^{NC} PF_j) \\ &= 1/(F + \sum_{j=1}^{NC} \mu_j \, |VIOL_j|) \end{aligned} \quad (11)$$

where

PF_j penalty associated with violated constraint j,

μ_j penalty multiplier associated with constraint j,

$VOIL_j$ amount of violation of constraint j

The penalty multipliers are chosen sufficiently large to discourage the selection of solutions with violated constraints.

4.3 Implementation of the BPSO solution for UC problem

The procedure of the proposed BPSO method is as shown below.

Step 1 Generate L initial individuals with dimension of $N*T$. The statuses at each scheduling time are determined by the given initial staus and equation (9).

Step 2 Calculate the evaluation value of each initialized individual x_i using the evaluation function f as given by equation (11).

Step 3 Compare each initialized individual's evaluation value with the individual's *pbest*. The individual who owns the best evaluation value among *pbests* is set to be *gbest*.

Step 4 Modify the velocity v_i of each individual x_i according to equation (7).

Step 5 If $v_{id}^{k+1} > v^{\max}$, then $v_{id}^{k+1} = v^{\max}$.
If $v_{id}^{k+1} < v^{\min}$, then $v_{id}^{k+1} = v^{\min}$.

Step 6 Modify the position of individual x_i according to equation (9) and (8). Equation (9) is prior to equation (8) to satisfy the minimum up and down time constraints.

Step 7 Calculate the evaluation value of the new individual. If x_i^{k+1} is better than *pbest*, then the current individual x_i^{k+1} is set to be *pbest*. Subsequently, if the best $pbest_i$ is better than *gbest*, then $pbest_i$ is set to be *gbest*.

Step 8 If the maximum iteration number is reached, then go to step 9. Otherwise, go to step 4.

Step 9 The individual that generated the latest *gbest* indicates the optimal units-scheduled combination during the scheduling period.

5 Simulation Results

Table 1 Data of 10 base units

	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
Pmax(MW)	455	455	130	130	162
Pmin(MW)	150	150	20	20	25
a($/h)	1000	970	700	680	450
b($/MWh)	16.19	17.26	16.60	16.50	19.70
c($/MW^{2}h)	0.00048	0.00031	0.002	0.00211	0.00398
MUT(h)	5	5	2	2	2
MDT(h)	5	5	2	2	2
σ($)	4500	5000	550	560	900
δ($)	4500	5000	550	560	900
τ(h)	4	4	2	2	2
initial status(h)	8	8	-5	-5	-6

	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
Pmax(MW)	80	85	55	55	55
Pmin(MW)	20	25	10	10	10
a($/h)	370	480	660	665	670
b($/MWh)	22.26	27.74	25.92	27.27	27.79
c($/MW^{2}h)	0.00712	0.00079	0.00413	0.00222	0.00173
MUT(h)	2	1	0	0	0
MDT(h)	2	1	0	0	0
σ($)	170	260	30	30	30
δ($)	170	260	30	30	30
τ(h)	2	2	1	1	1
initial status(h)	-3	-3	-1	-1	-1

Table 2 Hourly load demand

Hour	Demand(MW)	Hour	Demand(MW)
1	700	13	1400
2	750	14	1300
3	850	15	1200
4	950	16	1050
5	1000	17	1000
6	1100	18	1100
7	1150	19	1200
8	1200	20	1400
9	1300	21	1300
10	1400	22	1100
11	1450	23	900
12	1500	24	800

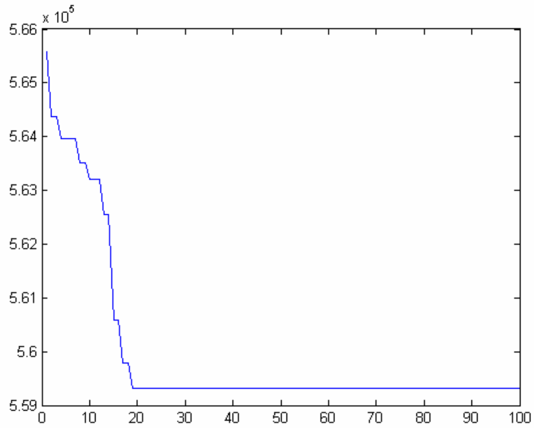

Fig.2 Convergence tendency of the evaluation value

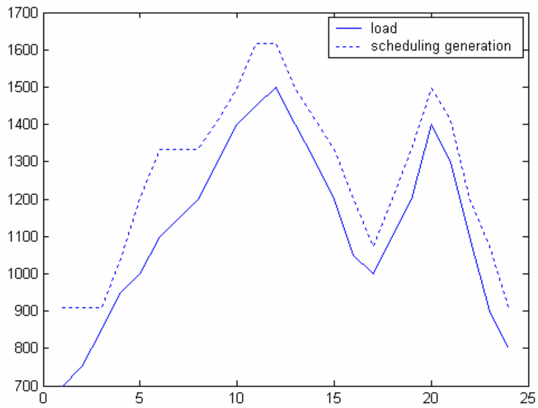

Fig.3 Scheduling generation-load

Table 3 Best individual (combination) by the proposed BPSO method

Hour	Operation Cost ($)	Startup Cost ($)	Spinning Reserve (MW)	Unit Schedule	Generation Schedule (MW)									
1	13683.13	0	210	1100000000	455	245	0	0	0	0	0	0	0	0
2	14554.50	0	160	1100000000	455	295	0	0	0	0	0	0	0	0
3	16301.89	0	60	1100000000	455	395	0	0	0	0	0	0	0	0
4	18637.68	1109.74	90	1101000000	455	365	0	130	0	0	0	0	0	0
5	20020.02	1793.94	202	1101100000	455	390	0	130	25	0	0	0	0	0
6	22387.04	1096.29	232	1111100000	455	360	130	130	25	0	0	0	0	0
7	23261.98	0	182	1111100000	455	410	130	130	25	0	0	0	0	0
8	24150.34	0	132	1111100000	455	455	130	130	30	0	0	0	0	0
9	26588.96	339.31	112	1111110000	455	455	130	130	110	20	0	0	0	0
10	29365.95	519.36	97	1111111000	455	455	130	130	162	43	25	0	0	0
11	31916.06	120.00	167	1111111110	455	455	130	130	162	73	25	10	10	0
12	33205.25	0	117	1111111110	455	455	130	130	162	80	25	53	10	0
13	29365.95	0	97	1111111000	455	455	130	130	162	43	25	0	0	0
14	26588.96	0	112	1111110000	455	455	130	130	110	20	0	0	0	0
15	24150.34	0	132	1111100000	455	455	130	130	30	0	0	0	0	0
16	20895.88	0	152	1101100000	455	440	0	130	25	0	0	0	0	0
17	19608.54	0	72	1100100000	455	455	0	0	90	0	0	0	0	0
18	21891.43	897.67	102	1110100000	455	455	130	0	60	0	0	0	0	0
19	24150.34	913.99	132	1111100000	455	455	130	130	30	0	0	0	0	0
20	29365.95	833.10	97	1111111000	455	455	130	130	162	43	25	0	0	0
21	26588.96	0	112	1111110000	455	455	130	130	110	20	0	0	0	0
22	21891.43	0	102	1110100000	455	455	130	0	60	0	0	0	0	0
23	17684.69	0	172	1100100000	455	420	0	0	25	0	0	0	0	0
24	15427.42	0	110	1100000000	455	345	0	0	0	0	0	0	0	0
Total	551682.71	7623.39	3153	559306.10										

The BPSO program was implemented in MATLAB and executed on a Pentium III 800 personal computer with 256MB RAM. The program is tested on a 10-unit system, which data is given in Table 1 and 2 [15]. The 10-unit system simulation results of [15] were incorrect because the startup costs of the units which start up at the first scheduling time were not calculated. Moreover, the worst generation cost in Table (8) of [15] was also unbelievable.

The spinning reserve is assumed to be 5% of the load demand. The population size is set to be 20, and the iteration is set to be 100. The convergence tendency of the best evaluation value in the population during BPSO processing is shown in Fig.2. Fig.3 shows the scheduling generation and load demand. Table 3 illustrates the solution obtained by the BPSO. Operation, startup costs, spinning for the 24h period, unit on/off schedule and generation supplying the load is also provided in Table 3. 50 trials are performed to examine the quality of the solution. The results are shown in Table 4. As can be seen, the proposed BPSO method has good quality and convergence characteristic.

Table 4 The quality of the solution

best generation cost ($)	559306.10
worst generation cost ($)	562383.57
average generation cost ($)	560894.43
standard deviation ($)	751.21

6 Conclusion

In this paper, a modified BPSO method is proposed to solve the UC problems. A new strategy is employed for representing chromosomes and encoding the problem search space, of which the minimum up and down time constraints are taken into account in initializing and modifying the particles. Thus, the individuals of the BPSO method are all satisfy the minimum up and down time constraints. The penalty coefficients are used to calculate the evaluation value of the individuals, which discourage the infeasible combinations. The feasibility of the proposed method is demonstrated by simulation.

References:

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

[2] Happ H.H., R.C. Johnson, W.J. Wright, Large scale hydro-thermal unit commitment-method and results, *IEEE Trans. on PAS*, Vol.PAS-90, 1971, pp. 1373-1383.

[3] Baldwin, C.J., K.M. Dale, R.F. Dittrich, A study of economic shutdown of generating units in daily dispatch, *AIEE Tr. on PAS*, Vol.78, 1960, pp. 1272-1284.

[4] W.L. Snyder, H.D. Powell, Jr., J.C. Rayburn, Dynamic programming approach to unit commitment, *IEEE Transactions on Power Systems*, Vol.2, No.2, 1987, pp. 339-350.

[5] W.J. Hobbs, G. Hermon, S. Warner, G.B. Sheble, An enhanced dynamic programming approach for unit commitment, *IEEE Transactions on Power Systems*, Vol.3, No.3, 1988, pp. 1201-1205.

[6] Z. Ouyang, S.M. Shahidehpour, An intelligent dynamic programming for unit commitment application, *IEEE Transactions on Power Systems*, Vol.6, No.3, 1991, pp. 1203-1209.

[7] F. Zhuang, F.D. Galiana, Towards a more rigorous and practical unit commitment by Lagrangian relaxation, *IEEE Transactions on Power Systems*, Vol.3, No.2, 1988, pp. 763-773.

[8] S.J. Wang, S.M. Shahidehpour, D.S. Kirschen, S. Mokhtari, G.D. Irisarri, Short-term generation scheduling with transmission and environmental constraints using augmented Lagrangian relaxation, *IEEE Transactions on Power Systems*, Vol.10, No.3, 1994, pp. 1294-1301.

[9] Chuan-Ping Cheng, Chih-Wen Liu, Chun-Chang Liu, Unit commitment by Lagrangian relaxation and genetic algorithms, *IEEE Transactions on Power Systems*, Vol.15, No.2, 2000, pp. 707-714.

[10] Time T. Maifeld, Gerald B. Sheble, Genetic-based unit commitment algorithm, *IEEE Transactions on Power Systems*, Vol.11, No.3, 1996, pp. 1359-1370.

[11] S.A. Kazarlis, A.G. Bakirtzis, V. Petridis, A genetic algorithm solution to the unit commitment problem, *IEEE Transactions on Power Systems*, Vol.11, No.1, 1996, pp. 83-92.

[12] K.S. Swarup, S. Yamashiro, Unit commitment solution methodology using genetic algorithm, *IEEE Transactions on Power Systems*, Vol.17, No.1, 2002, pp. 87-91.

[13] A.H. Mantawy, Youssef L. Abdel-Magid, Shokri Z. Selim, A simulated annealing algorithm for unit commitment, *IEEE Transactions on Power Systems*, Vol.13, No.1, 1998, pp. 197-204.

[14] J. Kennedy, R. Eberhart, Particle swarm optimization, *Proceddings of IEEE International Conference on Neural Networks*, Vol. IV, Perth, Australia, 1995, pp. 1942-1948.

[15] Zwe-Lee Gaing, Discrete particle swarm optimization algorithm for unit commitment, *IEEE Power Engineering Society General Meeting*, Vol.1, 2003, pp. 13-17.

[16] J. Kennedy, R. Eberhart, A discrete binary version of the particle swarm algorithm, *Proceeding of IEEE International Conference on Evolutionary Computation*, Anchorage, 1998, pp. 84-89.

[17] Y. Shi, R.C. Eberhart, Empirical study of particle swarm optimization, *Proceedings of the 1999 Congress on Evolutionary Computation*, Piscataway, 1999, pp. 1945-1950.