

Improved PSO Approach for The Solution of Unit Commitment Problem

Y.R. Guo

Department of Mechanical and Electrical Engineering
Shenzhen Research Institute of Xiamen University
Xiamen University, China

J.R. Zhang*

Department of Mechanical and Electrical Engineering
Xiamen University, China
*Corresponding author

Y.H. Wu

Department of Mechanical and Electrical Engineering,
Xiamen University, China

J.D. Ren

Department of Mechanical and Electrical Engineering,
Xiamen University, China

Abstract—Unit commitment problem (UCP) is a large scale nonlinear hybrid integer programming problem and it involves determining on/off states of units and generations of units. This paper proposes an improved particle swarm optimization (IPSO) for the solution of UCP. In the proposed approach, the on/off states of units are limited into feasible schedules by providing a new method related to a time order at first. After that, the problem is transformed into a simple economic load dispatch problem. Then this dispatch problem is solved by an improved priority list technique instead of the classical equal lambda-iteration method. All the above improvements are embedding into the framework of PSO approach for UCP. It is seen from the numerical results that the presented algorithm surely possesses a high quality and a high convergence speed.

Keywords—unit commitment problem; particle swarm optimization; time order; priority list

I. INTRODUCTION

The main purpose of optimal unit commitment problem (UCP) is to determine the on/off state of each generating unit and the generation of every committed unit for a given horizon, under various operating constraints, including fuel constraints, multiple emission requirements, ramp rate limits, minimum up and down time limits and proper spinning reserves. Since the optimal commitment scheduling can save huge amount of costs and improve reliability of power system, many methods have been developed to solve the UCP, such as Lagrange Relax [1], Dynamic Programming [2], and Genetic Algorithm [3].

Based on the analogy of swarm of bird and school of fish, particle swarm optimization (PSO) is suggested by Kennedy and Eberhart in 1995 [4]. It searches not only local optimal solution but also global optimal solution and it can easily deal with various difficult nonlinear constraints, with fewer experienced parameters than other methods, so it can give out a relatively optimal solution quickly. In recent years, it has been used to solve many complex power systems optimization including UCP [5-7] for its flexibility and efficiency.

In this paper, the authors split up traditional UCP into two stages: optimization of on/off states of generating units and

optimization of generations of committed units. The on/off states of generating units are limited into feasible regions at first stage through a novel time order introduced. An improved priority list technique is employed instead of using the equal lambda-iteration technique which only uses the average fuel cost at the full load of units. All the above improvements are embedding into the PSO approach for UCP, and the numerical results show that the new method can really give out good result quickly.

II. FORMULATION OF UCP

A. Objective Function

The UCP aims at finding out the solution that the total production cost can be minimized and satisfy all kinds of constraints [8]. The total production cost F is the sum of the fuel cost and start-up cost for all units during a time horizon. Thus, the objective function of the UCP is:

$$\min F = \sum_{t=1}^T \sum_{i=1}^N [f_i(P(i,t)) + ST_i^t(1-U(i,t-1))]U(i,t) \quad (1)$$

where N is the number of generators; T is the total scheduling periods; $U(i,t)$ is the on/off status of unit i at time t ; $P(i,t)$ is generation of unit i at time t ; $f_i(P(i,t))$ is the fuel cost function of $P(i,t)$, and it can be expressed as:

$$f(P(i,t)) = a(i) + b(i)P(i,t) + c(i)P^2(i,t) \quad (2)$$

where $a(i)$, $b(i)$ and $c(i)$ represent the unit cost coefficients; ST_i^t is the generator start-up cost for restarting a de-committed generating unit, which is associated with the number of hours during which the unit has been off. In general, the start-up cost can be expressed as follows:

$$ST_i^t = \begin{cases} \text{Hotcost}(i), & \text{Mindown}(i) \leq \text{Toff}(i,t) \leq \text{Mindown}(i) + \text{Coldhrs}(i) \\ \text{Coldcost}(i), & \text{Toff}(i,t) > \text{Mindown}(i) + \text{Coldhrs}(i) \end{cases} \quad (3)$$

where $\text{Toff}(i,t)$ is the duration for which unit i has remained offline at hour t and $\text{Mindown}(i)$ is the minimum down-time of the unit i .

B. System And Unit Constraints

(1) System load constrains: $\sum_{i=1}^N P(i,t)U(i,t) = P_D(t)$ where $P_D(t)$ is the system load demand at hour t.

(2) Generation power limits: $P_{\min}(i) \leq P(i,t) \leq P_{\max}(i)$ where $P_{\min}(i)$ and $P_{\max}(i)$ are minimum and maximum generation power of unit i.

(3) Spinning reserve constraints: $\sum_{i=1}^N P_{\max}(i)U(i,t) \geq P_D(t) + P_R(t)$ where $P_R(t)$ is the spinning reserve requirement, which is assumed to be 10% of the hourly demand $P_D(t)$ in this article.

(4) Minimum up-time/down-time constraints: $Ton(i,t) \geq Minup(i)$ and $Toff(i,t) \geq Mindown(i)$ where $Ton(i,t)$ is the duration for which unit i has remained online at hour t and $Minup(i)$ and $Mindown(i)$ is the minimum up-time and down-time of the unit i, respectively.

(5) Initial states constraints:

$Ton(i,1) = Initial(i) + U(i,1)$ if $Initial(i) > 0 \& U(i,1) = 1$
 $Toff(i,1) = -Initial(i) + (1 - U(i,1))$ if $Initial(i) < 0 \& U(i,1) = 0$ where e $Initial(i)$ stands for the initial status of unit i.

III. IMPROVED PSO APPROACH FOR UCP

A. Overview of PSO

A classical PSO usually generates a randomly diversified population with M particles. The fitness of all the individuals will be calculated through some way with accordance to their position which usually stands for the solution of the optimization problem. The particle with the best fitness value will be chose as the gbest and the current position of each particle will be set as its own pbest. Then the iterative process of the PSO will be performed until some preset criteria are met. In the iterative process, all the individuals will adjust their position through the velocity and position update formulations firstly. Then each particle will compare its current new fitness with that of its own best one which means the fitness of its pbest and the winner will replace the pbest. Finally, the best one of all the pbest will be chose as the gbest. The flow chart for classical PSO is shown in fig.1.

B. Improved PSO approach for UCP

There are two types of variable quantities for each particle in UCP, $U(i,t)$ and $P(i,t)$, while $U(i,t)$ is a 0/1 integer variable and $P(i,t)$ is continuous variable. Though PSO performs perfect in solving continuous optimization problems, it is not an effective algorithm for mixed integer optimization problems, and we have to do some changes for UCP when using PSO. Here in the proposed IPSO approach, we change the basic update formulations for the position and velocity of particles into the following expressions according to Ref. [5]:

$$v_{i,d}^{k+1} = \omega_1 \otimes (pbest_{i,d}^k \oplus p_{i,d}^k) + \omega_2 \otimes (gbest_d^k \oplus p_{i,d}^k) \quad (4)$$

$$P_{i,d}^{k+1} = P_{i,d}^k \oplus v_{i,d}^{k+1} \quad (5)$$

where “ \oplus ” denotes “XOR” operator; “ \otimes ” denotes “AND” operator; “+” denotes “OR” operator; ω_1 and ω_2 are two random binary integer numbers uniformly distributed in the range of [0, 1]. Then the solution of the UCP problem is becoming to initialize $U(i, t)$ and optimize $P(i, t)$.

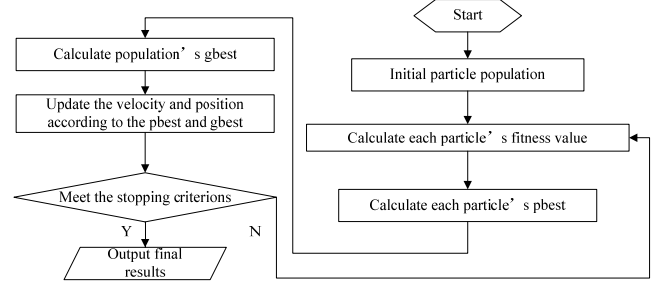


FIGURE 1. FLOW CHART FOR CLASSICAL PSO.

1) Initialization for $U(i,t)$: The first thing for IPSO is to limit all initialization of unit's state into feasible solutions. Traditional methods have fateful disadvantage on dealing with constraints: they usually restart the handling process once there is no proper or feasible solution. In that case, more time will be consumed and the feasible solutions may not be found finally. As for traditional time order (from $t=1$ to $t=T$), scheduling often gets stuck in the situation that the load demand is fitted for the hour t but not fitted for hour $(t+1)$ if the load of time $t+1$ is larger than that of time t, when considering the constrain of minimum up/down times constraint.

The new method points out a way to form a new time order in optimizing on/off states. Finding out the time of load demand's pole which is bigger than load demands that are needed before and later, and form a new time order that follows the direction of decreasing load demand. Start the optimization of states from the time of pole and then schedule according to the new time order. Like the example given in Section 4, the load curve for 24hrs is shown in fig.2. We can see the time of maximum pole for the example is 12, so the new time optimizing order is 12, 13, 11, 14, 10, 15, 9, For the load demand in order r is always bigger than its next order $r+1$, so in this order, the situation referred before can be avoided and the speed can be improved. The basic steps for optimizing the on/off states are as follows:

Step 1: Create a random integer-matrix U to identify the unit states with dimension $N \times T$.

Step 2: Set $t=r_n(n=1,2,\dots,T)$ according to the new time order. For each unit, considering its minimum up/down time constraint, and if the distance from r_n to r_{n-1} is smaller than its minimum up/down time, only consider the constraint to r_{n-1} .

Step 3: $R(t)$ represents the spinning reserve at hour t and can be expressed as

$$R(t) = \sum_{i=1}^N P_{\max}(i)U(i,t) - P_D(t) - P_R(t).$$

Step 4: If $R(t) < 0$, calculate average full-load cost of each uncommitted unit hour t and sort the units in the ascending order to obtain a list of SS1 according to Priority List. Set every $U(i,t)=1$ if not violate its minimum down time constraint in order until $R(t)+P_{\max}(i) \geq 0$. If the distance from r_n to r_{n-1} is smaller than its minimum down time, only consider the constraint to r_{n-1} .

Step 5: Update the $R(t)$.

Step 6: Calculate the average full-load cost of each committed unit at hour t and sort the units in the descending order to obtain a list of SS2 according to Priority List. For every i in SS2, if $R(t) \geq P_{\max}(i)$, than set $U(i, r_n)=0$ if not violate its minimum up time constraint in order. If the distance from r_n to r_{n-1} is small than its minimum up/down time, only consider the constraint to r_{n-1} .

Step 7: Consider the initial state constraints. If fitted, set $t=t_{+1}$ according to the new time order and back to Step 2 until the new time order is over.

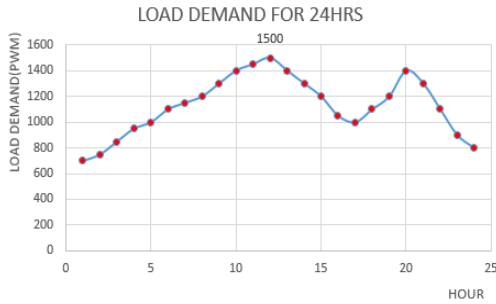


FIGURE II: The load curve for 24hrs.

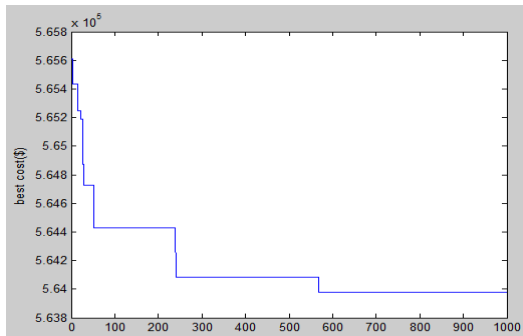


FIGURE III. CONVERGENCE CHARACTERISTICS FOR UCP USING IPSO APPROACH.

2) *Optimization for $P(i,t)$* : After $U(i,t)$ has been determined, the optimization for $P(i,t)$ becomes a simple economic load dispatch problem. Within the feasible unit commitment schedule, both the classical equal lambda-iteration method and the classical PSO can be used to solve the problem. Compared with these methods, Priority list (PL) method is easier and quicker. Traditional PL method is described as follows:

Step 1: The average full-load cost of a unit is defined as the cost per unit of power (\$/MW) when the unit is at its full capacity. This average full-load cost of a unit according to the fuel cost function given by eqn. (2) can be expressed as :

$$\alpha_i = \frac{f_i(P_{\max}(i))}{P_{\max}(i)} = \frac{a_i}{P_{\max}(i)} + b_i + c_i P_{\max}(i)$$

Step 2: The units are ranked by their α in ascending order. Thus, the traditional priority list of units will be formulated based on the order of α_i , in which a unit with the lowest α_i will have the highest priority to be dispatched.

Instead of the traditional PL technique using the average full-load cost of a unit to sort the order of units, the improved PL technique adopted in this paper employs the average cost of a unit at different power generation levels, such as the minimum output power, the average output power and even just b_i without the influence of output changes for simplicity. Through the comparisons among the different orders and their corresponding results obtained by the different power output levels, the best one will be chosen as the final output.

3) *Implementation of the IPSO for solving UCP*: The detailed implementation with the proposed IPSO approach for UCP is described as follows.

Step 1: Initialization each individual's $U(i,t)$ and limit them into feasible solutions.

Step 2: For every individual, compare and find out the optimal $P(i,t)$ for determined $U(i,t)$ with different PL.

Step 3: Compare each particle's objective function F to its best one and update the best position (pbest) of the particle and the particle who owns the best objective F is set as the global best position gbest.

Step 4: Modify the velocity and position of each particle in the swarm using eqns. (4) and (5), respectively.

Step 5: Back to Step 2 until the maximum iteration number is reached.

IV. SIMULATIONS

The IPSO approach gives by the article is applied to UCP for realistic power systems which consist of 10 units along with 24h load demands. The detailed parameters for this test system come from Ref. [5].

TABLE I. RESULT OF $U(I,T),P(I,T)$.

Hour	Unit generation (MW)									
	1	2	3	4	5	6	7	8	9	10
1	455	245	0	0	0	0	0	0	0	0
2	455	295	0	0	0	0	0	0	0	0
3	455	370	0	0	25	0	0	0	0	0
4	455	455	0	0	40	0	0	0	0	0
5	455	390	0	130	25	0	0	0	0	0
6	455	360	130	130	25	0	0	0	0	0
7	455	410	130	130	25	0	0	0	0	0
8	455	455	130	130	30	0	0	0	0	0
9	455	455	130	130	85	20	25	0	0	0
10	455	455	130	130	162	33	25	10	0	0

11	455	455	130	130	162	73	25	10	10	0
12	455	455	130	130	162	80	25	43	10	10
13	455	455	130	130	162	33	25	10	0	0
14	455	455	130	130	85	20	25	0	0	0
15	455	455	130	130	30	0	0	0	0	0
16	455	310	130	130	25	0	0	0	0	0
17	455	260	130	130	25	0	0	0	0	0
18	455	360	130	130	25	0	0	0	0	0
19	455	455	130	130	30	0	0	0	0	0
20	455	455	130	130	162	33	25	10	0	0
21	455	455	130	130	85	20	25	0	0	0
22	455	455	0	0	145	20	25	0	0	0
23	455	425	0	0	0	20	0	0	0	0
24	455	345	0	0	0	0	0	0	0	0

The optimal cost obtained by IPSO in 20 runs is \$563977. The optimal states of all generating units and the generation for them are shown in Table 1. It is found from the table that the schedule states and the generations satisfy all the constraints of UCP. The optimal convergence characteristic for UCP using the proposed IPSO approach is demonstrated in fig. 3. It can be obviously found from this figure that the IPSO has a quite competitive convergence speed. The comparisons of the proposed IPSO with other methods are shown in Table 2, including LR [1], GA [3], HPSO [7], EP [9] and SA [10]. According to the comparison of Table 2, it is clearly seen that the IPSO obtains a better result than other methods considered. Besides, it can be observed that the IPSO has a stable result, which means that IPSO is reliable in solving UCP. Hence, we can get a conclusion that the proposed approach can get competitive results of UCP at some faster convergence speed.

TABLE II. COMPARISON OF THE IPSO WITH OTHER METHODS.

Method	Number of Trials	Population Size	Maximum Generation	Cost (\$)		
				Best	Mean	Worst
LR	/	/	/	566,107	/	/
GA	20	50	500	565,825	/	570,032
EP	20	50	500	564,551	565,352	566,231
HPSO	50	20	1000	563,942	564,772	565,785
SA	10	/	/	565,828	565,988	566,260
IPSO	20	20	1000	563,977	564,006	564,018

V. CONCLUSIONS

The article presents a novel IPSO algorithm for UCP in power systems. This proposed approach splits up the UCP into two stages. A new time order is proposed at the first stage for the optimization of on/off state for each unit. At the second stage, the article proposed a modified priority list technique. Both of these variations can help to improve the convergence speed and simplify the programs. The simulation results provided by the numerical example clearly reveal that the proposed IPSO algorithm can be used as an excellent optimizer in solving UCP.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial supports from the Knowledge Innovation Program of Shenzhen City and the National Undergraduate Innovation Training Programs of China.

REFERENCES

- [1] Balci, H.H. & Valenzuela, J.F., Scheduling electric power generators using particle swarm optimization combined with the Lagrangian relaxation method. *International Journal of Applied Mathematics and Computer Science*, 14(3), pp. 411-421, 2004.
- [2] Snyder, W.L., Powell, H.D. & Rayburn, J.C., Dynamic programming approach to unit commitment. *IEEE Transactions on Power Systems*, 2(2), pp. 339-350, 1987.
- [3] Kazarlis, S.A., Bakirtzis, A.G. & Petridis, V., A genetic algorithm solution to the unit commitment problem. *IEEE Transactions on Power Systems*, 11(1), pp. 83-92, 1996.
- [4] Kennedy, J. & Eberhart, R.C., Particle swarm optimization. *Proc. IEEE Int. Conf. Neural Networks (ICNN'95)*, Perth, Australia, IV, pp. 1942-1948, 1995.
- [5] Xiaohui, Y., Hao, N., Anjun, S., et al., An improved binary particle swarm optimization for unit commitment problem. *Expert Systems with Applications*, 36(4), pp. 8049-8055, 2009.
- [6] Jacob, I.R., Raghuvver, C., Rakesh, G.A., et al., Solution to profit based unit commitment problem using particle swarm optimization. *Applied Soft Computing*, 10, pp. 1247-1256, 2010.
- [7] Ting, T.O., Rao, M.V.C. & Loo, C.K., A novel approach for unit commitment problem via an effective hybrid particle swarm optimization. *IEEE Transactions on Power Systems*, 21(1), pp. 411-418, 2006.
- [8] Álvarez, L.J., Ceciliano-Meza, J.L., Guillén, M.I., et al., A MIQCP formulation to solve the unit commitment problem for large-scale power systems. *International Journal of Electrical Power & Energy Systems*, 36(1), pp. 68-75, 2012.
- [9] Juste, K.A., Kita, H., Tanaka, E., et al., An evolutionary programming solution to the unit commitment problem. *IEEE Transactions on Power Systems*, 14(4), pp. 1452-1459, 1999.
- [10] Simopoulos, D.N., Kavatza, S.D. & Vournas, C.D., Unit commitment by an enhanced simulated annealing algorithm. *IEEE Transactions on Power Systems*, 21(1), pp. 193-201, 2006.