Journal of Energy & Environment 4 (2005) 21 – 35

Solution to Unit Commitment Problem with Spinning Reserve and Ramp Rate Constraints Using Ant Colony System

S. P. Simon, N. P. Padhy and R. S. Anand
Department of Electrical Engineering
Indian Institute of Technology Roorkee
Roorkee-247667, India
Email: simondee@iitr.ernet.in
(Received on 29 Aug 2004, revised on 1 Jan 2005)

Abstract

In this paper unit commitment problem (UCP) with spinning reserve and ramp rate constraint has been solved using Ant Colony System (ACS). In this model, foraging behavior of the real ants for finding its food from its start (nest) to its destination (food) is simulated to obtain the optimum solution. A search strategy has been adopted which involves the artificial ants to obtain the minimum cost path for the commitment of thermal units for the given time period. The search space is seen as the multi stage decision, which provides the ant search a competitive edge over conventional Dynamic programming (DP) method. In the first step all possible combinations of the states satisfying the load demand with spinning reserve have been selected for ants search space. Then the artificial ants are allowed to explore the minimum cost path in that search space. The proposed ACS model has performed successfully for a ten-unit generation system and may be extended to practical systems with large number of units.

Introduction

The hard combinatorial unit commitment problem (UCP) determines the optimum schedule of generating units (i.e. switching on and off of N generating units over a period of time for the demand forecasted to be served) by minimizing the over all cost of the power generation while satisfying a set of system constraints. Finding a good solution to the UCP in a reasonable time is very critical since it could mean significant annual financial savings in power generating costs. To “commit” a generating unit is to “turn it on.” that is to bring the unit up to speed, synchronize it to the system, and connect it so that it can deliver power to the network. The problem with “commit enough units and leave them on line” is one of the economics. It is quite expensive to run too many generating units. A great deal of money can be saved by turning units off (decommiting them) when they are not needed. However, the generic UCP can be formulated as to minimize operational cost subject to minimum up-time and down-time constraints, crew constraints, ramp constraints, spinning reserve, unit capability limits, derating of units, unit status, generation constraints and reserve constraints [1-2].

Mostly numerical techniques have been used to solve the unit commitment problem. Different concepts have been evolved over past years, which used many assumptions to simplify and reduce the computational complexity. A number of studies dealing with it have held the assumption that the unit generating capability follows a step change from zero to the rated capacity and vice versa. In fact, when a unit is in the start up process, a pre-warming process must be introduced in order to prevent a brittle failure. Therefore, because of the unit physical limitations, the unit generating capability increases as a ramp function [3]. The spinning reserve is taken into account since there must be a reserve allocated to various units which is given as percentage of forecasted peak demand, or that reserve must be capable of making up the loss of the most heavily loaded unit in a given period of time. These are the reasons that exhaustive enumeration techniques are used to find the optimal solution, since it looks at every possible solution combination.

Optimum schedules of committing units may save millions of dollars per year in production costs, so efforts are made recently by the application of simulated annealing, hybrid models, expert systems, artificial neural networks, fuzzy systems, genetic algorithm and ant colony system (ACS) for the solution of UCP [4-5].

Very few works are carried out applying ACS for scheduling related problems in the field of power systems. Shyh-Jier Huang applied [6] the ACS techniques for the enhancement of hydroelectric generation scheduling. In their computation procedure, colonies of ants are randomly generated and programmed to generate the number of ants within the feasible search space. These ants are positioned on different nodes in the feasible search space. They are allowed to transit between the nodes exactly the same way adopted in the ants solving of the Traveling Salesman Problem (TSP) where a required number of ants are placed in each node. In the work of Shi, Libao et. al. [7] for the problem of optimal unit commitment, concept of random perturbation behavior with a magnifying factor and mutation rate is incorporated with the basic Ant Colony Optimization (ACO) algorithm called the Ant System (AS) model, which is initially tested for solving the TSP [8]. This AS model does not include the concept of exploration and exploitation. Dorigo et al [9] has developed this improved version of AS model (i.e. the ACS model) and proved that ACS is better than AS model by including concept of exploration and exploitation when applied to hard combinatorial TSP problems. Even though Shi, Libao et. al. also used the concept of the ant initiation from the staring node of the search space; they have not worked on the exploration and exploitation concepts of ACS algorithm.

This paper applies the ACS model for solving UCP including ramping and spinning reserve constraints. In the used ACS model ants are allowed one by one at the starting node as continuous flowing ants into the search space and thus the initiation of ants makes the difference. So, the proposed algorithm became relatively simpler compared to the previous algorithms. Also in the proposed model the concept of exploration and exploitation both have been incorporated. So the ACS model proposed in this paper is an improved version of the earlier ant algorithms for solving UCP.

Background of Ant Colony System

Ant algorithms are first proposed by Marco Dorigo and his colleagues in the year 1991 as a multi-agent approach to difficult combinatorial optimization problems like the TSP and the quadratic assignment problem. Features such as positive feedback, distributed computation and constructive greedy heuristic approaches are some of the important characteristics of ant algorithms. In fact, Ant System (AS) which is the first model developed was originally a set of three algorithms called ant-cycle, ant-density, and ant-quantity which were proposed in Dorigo’s doctoral dissertation [8, 10]. There is currently a lot of ongoing activity in the scientific community to extend/apply ant-based algorithms to many different discrete optimization problems.

Ant colony system

Social insects like ants, bees, wasps and termites work by themselves in their simple tasks, independently of other members of the colony. However, when they act as a community, they are able to solve complex problems emerging in their daily lives, by means of mutual cooperation. This emergent behavior of self-organization by a group of social insects is known as "swarm intelligence" which is mainly due to the four basic ingredients a) Positive feedback b) Negative feedback (e.g. saturation, exhaustion or competition), c) Amplification of fluctuations (e.g. random walk, errors, random – task switching) and d) Multiple interaction. An important and interesting behavior of ant colonies is their foraging behavior and, in particular, how ants can find shortest paths between food sources and their nest. While walking from food sources to the nest and vice versa, ants deposit on the ground a chemical substance called pheromone, forming in this way a pheromone trail. The sketch shown in the Fig. 1 gives a general idea how real ants find a shortest path [11]. Ants can smell pheromone and, when choosing their path, they tend to choose, in probability, paths marked by strong pheromone concentrations. The pheromone trail allows the ants to find their way back to the food by their nest mates. The emergence of this shortest path selection behavior can be explained in terms of autocatalysis (positive feedback) and differential path length which uses a simple form of indirect communication mediated by pheromone laying, known as "stigmergy" through the environment, either by physically changing, or by depositing something on the environment [9, 12-16].

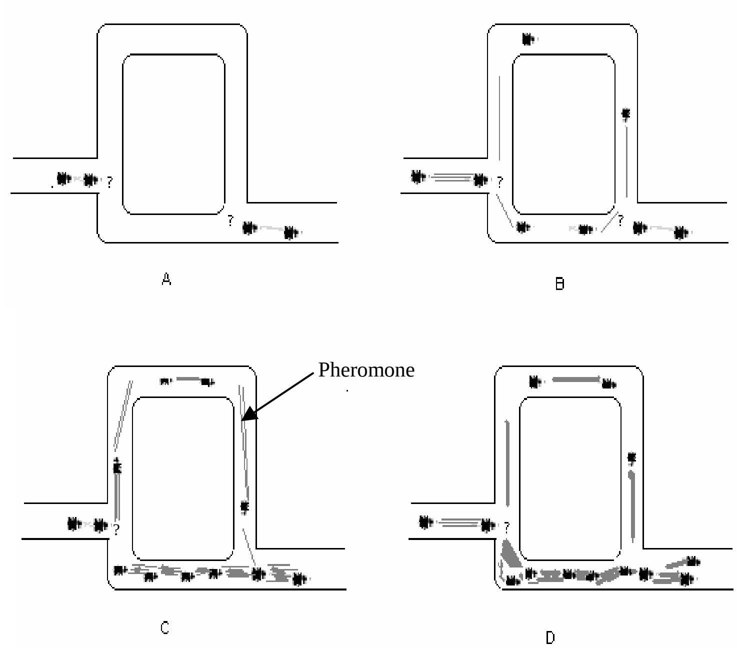

Fig. 1 Sketch showing how real ants find a shortest path

The sequence in Fig. 1 is as follows.

a) Ants arrive at a decision point.
b) Some ants choose the upper path and some the lower path. The choice is random.
c) Since ants move at approximately constant speed, the ants which choose the lower, shorter, path reach the opposite decision point faster than those which choose the upper, longer, path.
d) Pheromone accumulates at a higher rate on the shorter path. The number of dashed lines is approximately proportional to the amount of pheromone deposited by ants.

Artificial ant colony system

An Artificial Ant Colony System (AACS) is a population based heuristic algorithm on agents that simulate the natural behavior of ants developing mechanisms of cooperation and learning which enables the exploration of the positive feedback between agents as a search mechanism. In AACS, the use of: (*i*) a colony of cooperating individuals, (*ii*) an artificial pheromone trail for local stigmergetic communication, (*iii*) a sequence of local moves to find shortest paths, and (*iv*) a stochastic decision policy using local information and no look ahead are the same as real ACS, but artificial ants have also some characteristics which do not find their counterpart in real ants [13, 17]. They are

1. Artificial ants live in a discrete world and their moves consist of transitions from discrete states to discrete states.
2. Artificial ants have an internal state. This private state contains the memory of the ant‘s past actions.
3. Artificial ants deposit an amount of pheromone, which is a function of the quality of the solution found.
4. Artificial ants timing in pheromone laying is problem dependent and often does not reflect real ant’s behavior. For example, in many cases artificial ants update pheromone trails only after having generated a solution.

Essentially, an ACS algorithm performs a loop applying two basic procedures:

1. A procedure specifying how ants construct or modify a solution for the problem in hand;
2. A procedure for updating the pheromone trail.

The construction or modification of a solution is performed in a probabilistic way. The probability of adding a new term to the solution under construction is in turn, a function of a problem-dependent heuristic and the amount of pheromone previously deposited in this trail .The pheromone trails are updated considering the evaporation rate and the quality of the current solution [17-19].

Problem Formulation

The generic ACM based unit commitment problem can be formulated as

Minimize operational cost (OC)

$$OC = \sum_{i=1}^{N} \sum_{t=1}^{T} FC_{it}(P_{it}) + ST_{it} + SD_{it} \text{ \$/hr} \tag{1}$$

where,

$FC_{it}(P_{it})$ (Fuel cost) is the input/output (*I/O*) curve that is modeled with a polynomial curve (normally a quadratic function)

$$FC_{it}(P_{it}) = a_i P_{it}^2 + b_i P_{it} + c_i \text{ \$/hr} \tag{2}$$

where,

a_i, b_i and c_i are cost coefficients.
P_{it} is the power generation of unit i during time period t (MW),
The start-up cost is described by:

$$ST_{it} = TS_{it} F_{it} + \left(1 - e^{(D_{it} AS_{it})}\right) BS_{it} F_{it} + MS_{it} \text{ \$/hr} \qquad (3)$$

where,

TS_{it} the turbine startup cost;
BS_{it} boiler startup cost;
MS_{it} start-up maintenance cost;
D_{it} number of hours down;
AS_{it} boiler cool down coefficient.

Similarly, the shut-down cost is described by

$$SD_{it} = KP_{it} \text{ \$/hr} \qquad (4)$$

where,

K is the incremental shut-down cost.

Subject to the following constraints

Minimum up-time
$0 < T_{iu} \leq$ number of hours units G_i has been on-line
where T_{iu} is the minimum up-time

Minimum down-time
$0 < T_{id} \leq$ number of hours units G_i has been off-line
where T_{id} is the minimum down-time

Maximum and minimum output limits on generators
$P_{it} \min \leq P_{it} \leq P_{it} \max$

Power rate
$\nabla P_{it} \min \leq \nabla P_{it} \leq \nabla P_{it} \max$
where ∇P_{it} is the power rate of generator i (MW/hr)

Power balance

$$\sum_{i=1}^{N} P_{it} = load(H)$$

where *load (H)* is the system load at hour H

Spinning reserve

A sufficient amount of spinning reserve expressed as a percentage (20%) of total load demand should be maintained. Generally Transmission Losses (TL) is neglected for solving Unit Commitment Problem due to its large computational time and hence in the proposed approach TL is neglected.

Implementation of the Proposed ACS Model

The ACS implementation mainly consists of two phases. In the first phase, all possible S_t states of the t^{th} hour that satisfies the load demand is found and continued for the complete scheduling period of 24 hours. The ant search space, which involves multi decision states, is seen in Fig. 2. Here in the initialization part the forecasted load demands and other relevant problem data from the system is taken for computation. Economic dispatch (ED) using Lagrange multiplier method is used that calculates the generator output and the production cost for each hour. Exhaustive enumeration technique is used to find all possible combinations of the generating units available. Once the search space is identified, then in the second phase the ants are allowed to pass continuously through the ant search space. Each ant starts its journey from the initial condition termed as the starting node and reaches the end stage and finally vanishes. So it's a continuous flow of ants and the ant never returns back. Once an ant reaches the end stage, a tour is completed and it calculates the overall generation cost path. This process is continued until the ants find an optimal solution.

The ant colony search mechanism can be divided into a) initialization, b) transition rule, and c) pheromone trail update rule.

a) Initialization

During initialization the parameters such as the requisite number of ants, the relative importance of the pheromone trail, relative importance of the visibility, initial available pheromone trail, a constant related to the quantity of the trail laid by ants, evaporation factor, tuning factor etc… have to be fixed and taken care.

b) Transition rule

The transition probability for the k^{th} ant from one state i to next state j for an AS model is given by [8].

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}}{\sum_{k \in allowed_k}\left[\tau_{ik}(t)\right]^{\alpha}\left[\eta_{ik}\right]^{\beta}} & if\ \ j \in allowed_k \\ 0 & otherwise \end{cases} \tag{5}$$

where,

τ_{ij} the trail intensity on edge (i, j),
$\eta_{ij} = (1/C_{ij})$ called heuristic function, C_{ij} is the production cost occurred for that particular stage,
α is the relative importance of the trail, $\alpha \geq 0$
β is the relative importance of the visibility, $\beta \geq 0$

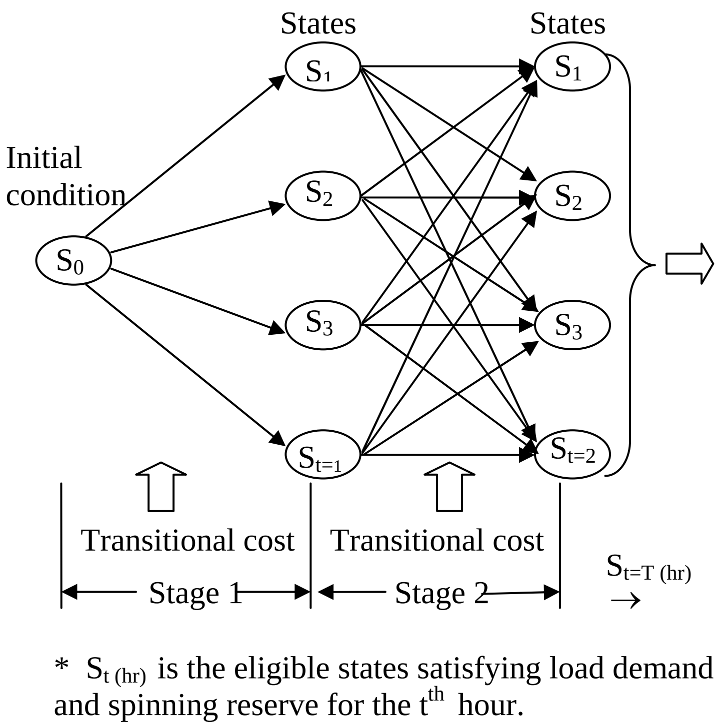

Fig. 2 An example of multi-decision search space

'$allowed_k$' is the available states k^{th} ant can choose from i^{th} state to j^{th} state. The probability transition rule for ACS is modified to allow explicitly for exploration. An ant k on state i choose the next state j to move according to the following rule,

$$j = \begin{cases} \arg\max_{u \in J_i^k} \left\{ [\tau_{iu}(t)].[\eta_{iu}]^{\beta} \right\} & if \quad q \leq q_0 \\ J & if \quad q > q_0 \end{cases} \tag{6}$$

where,

q is a random variable uniformly distributed over [0, 1]
q_0 is a tunable parameter $(0 \leq q_0 \leq 1)$

$J \in J_i^k$ is a state that is randomly selected according to probability

$$P_{iJ}^{k}(t) = \begin{cases} \dfrac{[\tau_{iJ}(t)]\,[\eta_{iJ}]^{\beta}}{\sum_{l \in J_i^k}[\tau_{il}(t)][\eta_{il}]^{\beta}} \end{cases} \tag{7}$$

when $q \leq q_0$ correspond correspond to an exploitation of the knowledge available about the problem, that is the heuristic knowledge about cost between states and the learned knowledge memorized in the form of pheromone trails, whereas $q > q_0$ favors more exploitation. Cutting exploration by tuning q_0 allows the activity of the system to concentrate on the best solutions instead of letting it explores constantly. Here 'l' is the allowable states [9].

c) Pheromone trail update rule

Once the ant's start choosing the minimum cost states pheromone trail update rule has to be implemented. In ACS the global trail updating rule is applied only to the edges belonging to the best tour since the beginning of the trail. The updating rule is:

$$\tau_{ij}(t) \leftarrow (1-\rho)\tau_{ij}(t) + \rho\Delta\tau_{ij}(t) \tag{8}$$

where,

(i, j)'s are the edges belonging to T^+, the best minimum cost tour since the beginning of the trial
ρ is a parameter governing pheromone decay, and

$$\Delta\tau_{ij}(t) = 1/L^+ \tag{9}$$

where,

L^+ is the cost of T^+. This procedure allows only the best tour to be reinforced by a global update. Local updates are however also performed, so that other solutions can emerge.

The local update is performed as follows: when, while performing a tour, ant k is in state i and selects a state $J \in J_i^k$, the pheromone concentration of (i, j) is updated by the following formula:

$$\tau_{ij}(t) \leftarrow (1-\rho)\tau_{ij}(t) + \rho\tau_0 \tag{10}$$

where,

τ_0 is the initial value of the pheromone trail.

Here there is some significant difference between ACS and AS which is given as follows:

a) ACS uses a more aggressive action rule than AS.
b) The pheromone is added only to arcs belonging to the global-best solution [10].

The flow chart of the proposed model is given in Fig. 3.

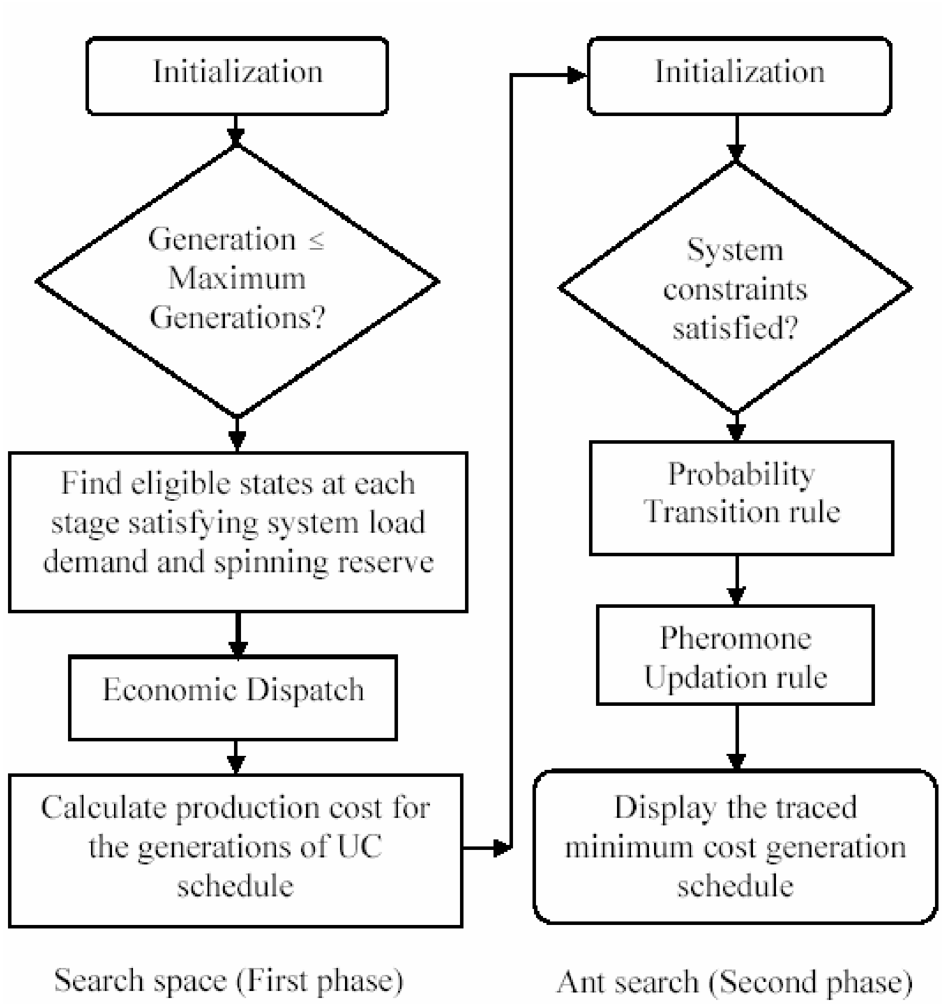

Fig. 3 ACS for UCP flow chart

Results and Discussions

In this proposed ACS model the control parameter are fixed as β = 10, ρ = 0.1 and q_0 = 0.9. These values have been fixed by varying different combinations of control parameters and by analyzing the results. The total number of ants taken is 50. The control parameters are fixed based on the properties of the ACS model, which is given as follows [20]:

a) The tuning factor when set to high will perform well because the model incorporates more exploitation of the knowledge available about the problem.
b) High values of β and/or low values of α make the algorithm very similar to a stochastic multi greedy algorithm.

The commitment schedules and the transitional cost at each stage obtained using DP and ACS for the ten unit system given in Appendix is tabulated in Table 1. The results observed for the ten-unit system shows that the ACS model performed better than the traditional dynamic programming approach. It is observed that the unit commitment schedules have been uniform during 5th to 18th hour because the practical load patterns do not vary significantly during that period. Above and all minimum up and down time of the generated units sometime force the units to occupy its previous state. In the case of unit commitment solutions, the unit state for a given hour is dependent on the previous hour's schedule due to generating unit constraints. So the proposed model cannot guarantee better solution at each and every hour, perhaps it has been concluded that total generation cost so obtained by the proposed model is quite encouraging against conventional technique.

The evolution of the convergence of minimum cost path can be understood from Fig. 4, i.e. after the passing of minimum number of ants through the ant search space and completing their tours, convergence of the cost finally becomes stagnant. Even after the stagnant situation, ants are still trying to explore if any of the available minimum cost paths may exist, which indicates the strength of the ant colony system model. This can be under stood by the oscillations seen in Fig. 5. It is also observed in Fig.6.,that the transitional cost, even though at one stage may be high for ACS approach with respect to DP, the ants can still foresee and pick out the stages afterwards which can be of low cost path and thus maintaining the overall reduction of generation cost. Thus the proposed model searches the optimal path (24 hours schedule) instead of individual solution state. A saving of 0.1393% is achieved with the help of ACS model when compared with DP. So the solutions of the proposed model has been concluded as better compared to the standard conventional Dynamic Programming based on its 24 hours total scheduling. The platform used for the implementation of this proposed approach is on AMD Athlon [TM] XP 1800 + 1.54 GHz, 224 MB of RAM and has been simulated in the MATLAB environment.

Table 1 Unit commitment schedule generated using ACS model for 10 units system

Period (Hr)	Demand (Mw)	Dynamic Programming Unit status	Dynamic Programming TC* ($/hr)	Ant Colony System Unit status	Ant Colony System TC* ($/hr)
1	1160	1111100000	2,467.90	1111100000	2,527.30
2	1265	1111101000	2,780.70	1111110000	2,756.30
3	1380	1111111000	3,024.20	1111111001	3,002.40
4	1555	1111111111	3,561.70	1111111111	3,476.70
5	1700	1111111111	3,603.10	1111111111	3,603.10
6	1825	1111111111	3,919.50	1111111111	3,919.50
7	1900	1111111111	4,119.20	1111111111	4,119.20
8	1945	1111111111	4,243.90	1111111111	4,243.90
9	1990	1111111111	4,378.20	1111111111	4,378.20
10	1990	1111111111	4,378.20	1111111111	4,378.20
11	1980	1111111111	4,347.30	1111111111	4,347.30
12	1935	1111111111	4,215.70	1111111111	4,215.70
13	1900	1111111111	4,119.20	1111111111	4,119.20
14	1845	1111111111	3,972.00	1111111111	3,972.00
15	1870	1111111111	4,038.40	1111111111	4,038.40
16	1850	1111111111	3,985.20	1111111111	3,985.20
17	1700	1111111111	3,603.10	1111111111	3,603.10
18	1515	1111111111	3,172.00	1111111111	3,172.00
19	1410	1111111111	2,946.70	1111111111	2,983.40
20	1315	1111111111	2,745.20	1111111111	2,732.50
21	1260	1111111111	2,633.20	1111011111	2,614.30
22	1205	1111111111	2,523.40	1111011111	2,498.30
23	1160	1111011111	2,432.10	1111011111	2,405.20
24	1135	1111011111	2,351.70	1111011111	2,353.90
Total Cost ($/day)			83,561.57		83,445.16
Time Taken (Secs)			182.57		318.89

**TC-Transitional cost*

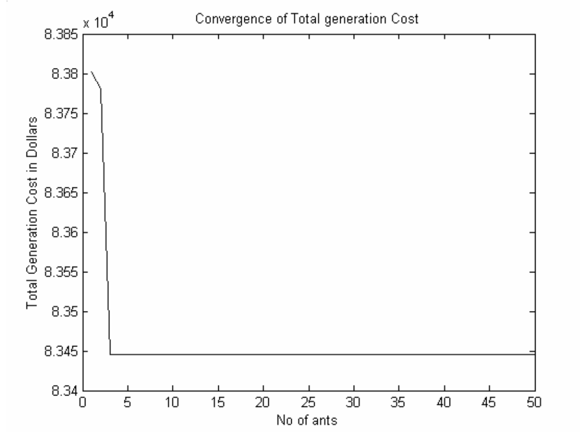

Fig. 4 Convergence of total generation cost

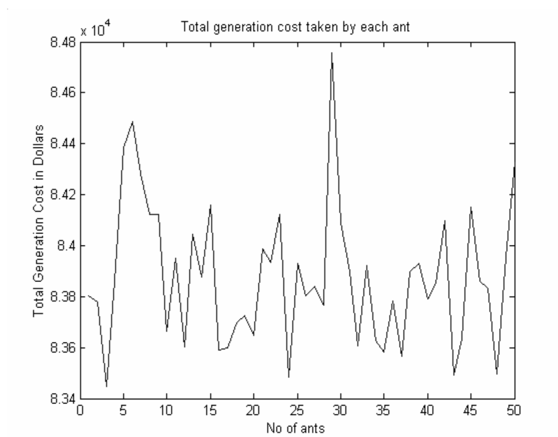

Fig. 5 Total generation cost path taken by each ant

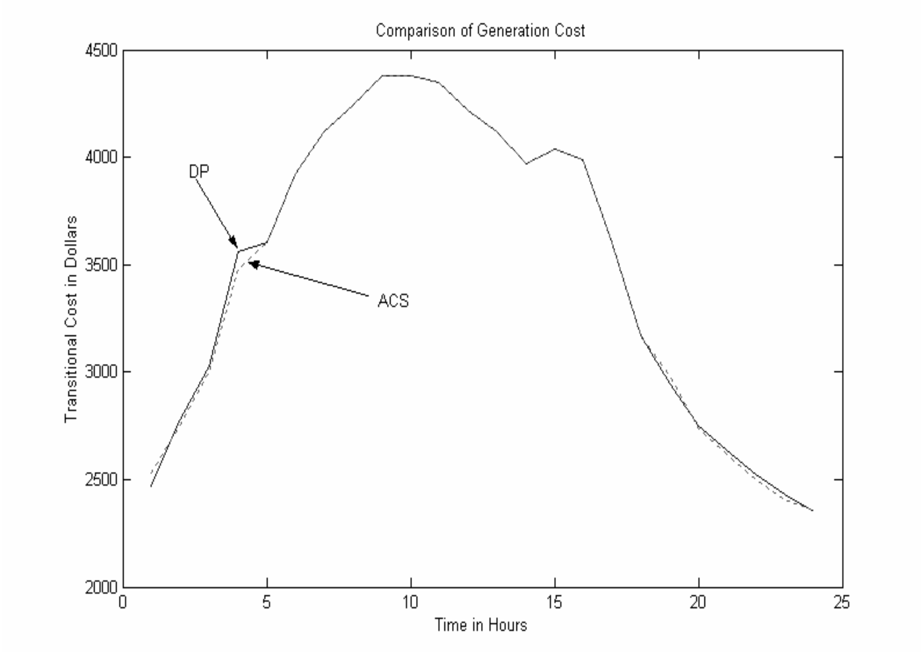

Fig. 6 Comparison of generation cost

Conclusion

This paper presents a new approach for solving the unit commitment problem (UCP) using ant colony system (ACS) by accommodating both the spinning reserve and ramp rate constraint. The new commitment schedule obtained has been compared with the conventional dynamic programming method. Since exhaustive enumeration technique is used, it takes into account the optimality of the solution. Ant algorithms are more suitable for combinatorial optimization problem and they have the potential in finding near global optimum solution. So they are very well suited for unit commitment problem, which is hard combinatorial in nature**.** The effectiveness of this method has been demonstrated on a practical ten-unit test system and may be extended to large systems. The results achieved are quite encouraging and indicate the viability of the proposed technique to deal with future unit commitment problems.

References

[1] A. J. Wood and B. F. Wollenberg, “Power generation, operation and control,” Wiley, New York, 1984.

[2] G. B. Sheble and G. N. Fahd, “Unit commitment literature synopsis,” IEEE Trans. on Power Systems, Vol. 9, No. 1, February 1994.

[3] C. Wang and S .M. Shahidehpour, “Effects of ramp-rate limits on unit commitment and economic dispatch,” IEEE Trans. on Power Systems, Vol. 8, No. 3, August 1993.

[4] N. P. Padhy, “Unit commitment- a bibliographical survey,” IEEE Trans. on Power Systems, Vol. 19, No. 2, May 2004.

[5] N. P. Padhy, “Unit commitment using hybrid models: A comparative study for dynamic programming, expert system, fuzzy system and genetic algorithms,” Electric Power and Energy Systems 23 (2000), February 1994, pp. 827-836.

[6] Shyh-Jier Huang, "Enhancement of hydroelectric generation scheduling using ant colony system based optimization approaches," IEEE Trans. Energy Conversion, Vol. 16, No. 3, September 2001, pp. 296-301.

[7] L. Shi, J. Hao, J. Zhou and Guoyu Xu, "Ant colony optimization algorithm with random perturbation behavior to the problem of optimal unit commitment with probabilistic spinning reserve determination," Electric Power Systems Research 69, May 2004, pp. 295-303.

[8] M. Dorigo, V. Maniezzo, and A.Colorni, "The ant system: optimization by a colony of cooperating agents," IEEE Trans. on Systems, Man, and Cybernetics–Part B, Vol. 26, No. 2, 1996, pp. 29–41.

[9] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman," IEEE Trans. on Evolutionary Computation, Vol. 1, No. 1, 1997, pp. 53-65.

[10] M. Dorigo, "Optimization, learning and natural algorithms," Ph. D thesis, Politecnico di Milano, Italy.

[11] S. P. Simon, N. P. Padhy and R. S. Anand, "Ant colony system: An application to traveling salesman problem," in Proc., National System Conference, IIT Kharagpur, India, December 2003, pp. 215-220.

[12] E. Bonabeau, M. Dorigo, and G. Theraulaz, "Swarm intelligence: From natural to artificial systems," Oxford University Press, New York, NJ, 1999.

[13] M. Dorigo and L. M. Gambardella, "Ant colonies for the traveling salesman problem," Technical Report/IRIDIA/3, Universite' Libre de Bruxelles, Belgium, 1996.

[14] M. Dorgio, G. Di Caro, and L. M. Gambardella, "Ant algorithms for distributed discrete optimization," Artificial Life, Vol. 5, No. 3, 1999, pp. 137–172.

[15] M. Dorigo and T. Stutzle, "ACO algorithms for the traveling salesman problem," Technical Problem Report, IRIDIA, Universite Libre de Bruxelles, Belgium, 1999.

[16] M. Dorigo and G. Di Caro, "The ant colony optimization meta-heuristic," Eds. London, U.K., McGraw-Hill, 1999, pp.11-32.

[17] M. Dorigo and T. Stutzle, "The ant colony optimization meta heuristic: algorithms, applications, and advances," Technical Report, IRIDIA-36, Universite Libre de Bruxelles, Belgium, 2000.

[18] N. Meuleau and M. Dorgio, "ant colony optimisation and stochastic gradient descent," Technical Report, IRIDIA, Universite Libre de Bruxelles, Belgium.

[19] M. Dorigo and T. Stutzle, **"**An Experimental study of the simple ant colony optimization algorithm" in Proc., WSES-International Conference on Evolutionary Computation (EC'01), N. Mastorakis (Ed.), WSES-Press International, 2001, pp.253–258.

[20] M. Dorigo, V. Maniezzo and A. Colorni, "Positive feedback as a search strategy," Technical Report 91-016, Politecnico di Milano, Italy, June 1991.

Appendix

Unit characteristics of a 10-unit system

Unit (No.)	Max. (MW)	Min. (MW)	Ramp Level (MW/Hr)	Minimum Up Time (Hr)	Minimum Down Time (Hr)	Shut down costs ($)	Start up cost ($)	Cold Start (Hrs)	Initial unit status
1	200	80	40	3	2	15	176	3	4
2	320	120	64	4	2	25	187	4	5
3	150	50	30	3	2	40	113	3	5
4	520	250	104	5	3	32	267	5	7
5	280	80	56	4	2	29	180	3	5
6	150	50	30	3	2	42	113	2	-3
7	120	30	24	3	2	70	94	2	-3
8	110	30	22	3	2	72	114	1	-3
9	80	20	16	0	0	75	101	0	-1
10	60	20	12	0	0	80	85	0	-1

Unit Characteristics (Ten unit system)
Initial unit status: Hours off (-) line or on (+) line

Fuel cost equations

$$C_1 = 082.00 + 1.2136 P_1 + 0.00148 P_1^2$$
$$C_2 = 049.00 + 1.2643 P_2 + 0.00289 P_2^2$$
$$C_3 = 100.00 + 1.3285 P_3 + 0.00135 P_3^2$$
$$C_4 = 105.00 + 1.3954 P_4 + 0.00127 P_4^2$$
$$C_5 = 072.00 + 1.3500 P_5 + 0.00261 P_5^2$$
$$C_6 = 029.00 + 1.5400 P_6 + 0.00212 P_6^2$$
$$C_7 = 032.00 + 1.4000 P_7 + 0.00382 P_7^2$$
$$C_8 = 040.00 + 1.3500 P_8 + 0.00393 P_8^2$$
$$C_9 = 025.00 + 1.5000 P_9 + 0.00396 P_9^2$$
$$C_{10} = 015.00 + 1.4000 P_{10} + 0.0051 P_{10}^2$$