Turkish Journal of Electrical Engineering & Computer Sciences
http://journals.tubitak.gov.tr/elektrik/
Research Article
Turk J Elec Eng & Comp Sci
(2016) 24: 4773 – 4789

doi:10.3906/elk-1411-77

Firefly algorithm with multiple workers for the power system unit commitment problem

Banumalar KOODALSAMY[1], **Manikandan BAIRAVAN VEERAYAN**[1], **Chandrasekaran KOODALSAMY**[2,*], **Sishaj Pulikottil SIMON**[3]
[1]Department of Electrical and Electronics Engineering, Mepco Schlenk Engineering College, Sivakasi, TamilNadu, India
[2]Department of Electrical and Electronics Engineering, National Institute of Technology Puducherry, Karaikal, India
[3]Department of Electrical and Electronics Engineering, National Institute of Technology, Tiruchirappalli, India

Received: 12.11.2014 • **Accepted/Published Online:** 13.09.2015 • **Final Version:** 06.12.2016

Abstract: This paper proposes an improved firefly (FF) algorithm with multiple workers for solving the unit commitment (UC) problem of power systems. The UC problem is a combinatorial optimization problem that can be posed as minimizing a quadratic objective function under system and unit constraints. Nowadays, highly developed computer systems are available in plenty, and proper utilization of these systems will reduce the time and complexity of combinatorial optimization problems with large numbers of generating units. Here, multiple workers are assigned to solve a UC problem as well as the subproblem, namely economic dispatch (ED) in distributed memory models. The proposed method incorporates a group search in a FF algorithm and thereby a global search is attained through the local search performed by the individual workers, which fine tune the search space in achieving the final solution. The execution time taken by the processor and the solution obtained with respect to the number of processors in a cluster are thoroughly discussed for different test systems. The methodology is validated on a 100 unit system, an IEEE 118 bus system, and a practical Taiwan 38 bus power system and the results are compared with the available literature.

Key words: Distributed memory models, economic dispatch problem, firefly algorithm, multiple workers, parallel computing, unit commitment problem

1. Introduction

The unit commitment (UC) problem is one of the most vital problems in power systems and is solved by system operators in regulated and deregulated electricity markets. The UC problem is a binary-variable power system optimization problem that involves finding the operating (on or off) status of thermal generating units. Its subproblem, the "economic dispatch", is a real-variable power system optimization problem that involves determining the economical allocation of continuous power among the committed units to balance the forecasted load demand. Several methods are proposed to find the operating status of the units in the UC problem. The conventional methods such as integer programming (IP) [1], dynamic programming (DP) [2], Bender's decomposition [3], Lagrangian relaxation (LR), [4], and branch and bound [5] have been used to solve the UC problem. These days metaheuristic methods like tabu search [6], genetic algorithm (GA) [7], simulated annealing (SA)[8], particle swarm optimization (PSO)[9], evolutionary algorithm [10], and artificial neural network [11] are

*Correspondence: chansekaran23@gmail.com

able to produce better solutions than the conventional methods like MILP (mixed integer linear programming) used in the load dispatch center.

One of the major limitations of the available algorithms in the literature is to get the best result within a reasonable time. Therefore, it becomes necessary for the researcher to develop a simple algorithm for solving the UC problem independent of number of generating units of the power system. In this article, a new swarm optimization technique known as the firefly (FF) algorithm, developed by Yang, was implemented to solve the UC problem using a FF algorithm with multiple workers in a parallel computing environment. It was successfully implemented on various mathematical functions and power system optimization problems and compared with other heuristic optimization algorithms [12,13] and the results obtained were found to be encouraging.

The aim of this paper is to show the effectiveness of the FF algorithm with multiple workers computing in a parallel platform for solving the UC problem. The proposed algorithm is implemented on a parallel platform called distributed memory models.

2. Problem formulation for the unit commitment problem

The UC problem formulation with the system and unit constraints considered are discussed in this section.

2.1. Objective function: fuel cost

The fuel cost (FC) function of the generating units is formulated as:

Minimize

$$\begin{aligned} FC &= \sum_{t=1}^{T}\sum_{i=1}^{Ng} FC(P_g(i,h)) \times I(i,h) + SC(i,h) \\ &= \sum_{t=1}^{T}\sum_{h=1}^{Ng} \left(\left(a_g(i) + b_g(i)P_g(i,h) + c_g(i).P_g(i,h)^2\right) \times I(i,h)\right) + SC(i,h) \end{aligned} \tag{1}$$

where FC is the total operating cost and $FC(P_g(i,h))$ is the fuel cost ($) of the generating unit i at hour h. The fuel cost coefficients of the i^{th} unit *is* $a_g b_g, c_{g.}$ N_g is the number of generating units, T is the total number of h considered, and $I(i,h)$ is the status of unit i at *hth* h (i.e.) 1 for ON and 0 for OFF, *SC(i,h)* is the startup cost of unit i at *hth* h, and $P_g(i,h)$ is the generation power output of unit i at *hth* h.

2.2. Problem constraints

2.2.1. Equality constraint: generation balance constraints

$$\sum_{i=1}^{Ng} (P(i,h) * I(i,h)) = Load_h\,,\ h \in [1,T], \tag{2}$$

where $Load_h$ is the total system demand at *hth* h.

2.2.2. System reserve constraint

Power system reliability can be improved by keeping certain generation capacity as reserve, i.e.

$$\sum_{i=1}^{Ng} (P(i,\max) * I(i,h)) \geq Load(h) + SR(h),\ \ h \in [1,T], \tag{3}$$

where $P(i, \max)$ is the maximum limit of power output of unit i, and *SR(h)* is the system spinning reserve in MW at *hth* h.

2.2.3. Power limits of thermal unit

Power generation levels of thermal units are within the limit defined by the following equation:

$$\begin{array}{lll} P(i,\min) \leq P(i,h) \leq P(i,\max) & \text{when} & I(i,h)=1 \\ P(i,h)=0 & \text{when} & I(i,h)=0, \end{array} \quad (4)$$

where $P(i, \min)$ is the minimum limit of power output for unit i.

2.2.4. Minimum ON/OFF durations of thermal unit

A unit must be continuously ON/OFF for predefined duration before it can be switched OFF/ON, i.e.

$$X_i^{on}(h) \geq MU(i) \quad (5)$$

$$X_i^{off}(h) \geq MD(i), \quad (6)$$

where *MU(i)* is the minimum time that unit i should kept in *ON* status and *MD(i)* is the minimum time that unit i should kept in *OFF* status. $X_i^{on}(h)$ and $X_i^{off}(h)$ are the time periods for which thermal unit i is ON and OFF status at *hth* h.

2.2.5. Unit ramp constraints

$$P(i,h) - P(i,h-1) \leq RU(i) \quad (7)$$

$$P(i,h-1) - P(i,h) \leq RD(i), \quad (8)$$

where *RU(i)* and *RD(i)* are ramp up and ramp down rate limit of unit i.

3. Review of firefly algorithm

The firefly algorithm is derived from the simulation of firefly group behavior and the main part of the algorithm concentrates on the absolute brightness of fireflies, which represents the value of an objective function. Also the position of the fireflies represents the solution of the problem to be solved. The relative brightness is gained by comparing 2 fireflies and its association is related to attraction. One firefly is attracted by a brighter one and adjusts its position according to the level of attraction. The step by step procedure of the firefly algorithm developed by Yang is discussed in [12]. The flowchart for the FF algorithm is given in Figure 1.

4. Firefly with multiple workers

The growing interest in multithreading programming and the availability of systems supporting multithreading further emphasizes the trend towards shared memory programming models. Parallel computing is achieved based on distributed memory systems with multiple workers of the program running on different nodes as shown in Figure 2. A master node sends its commands to the workers participating in the parallel environment.

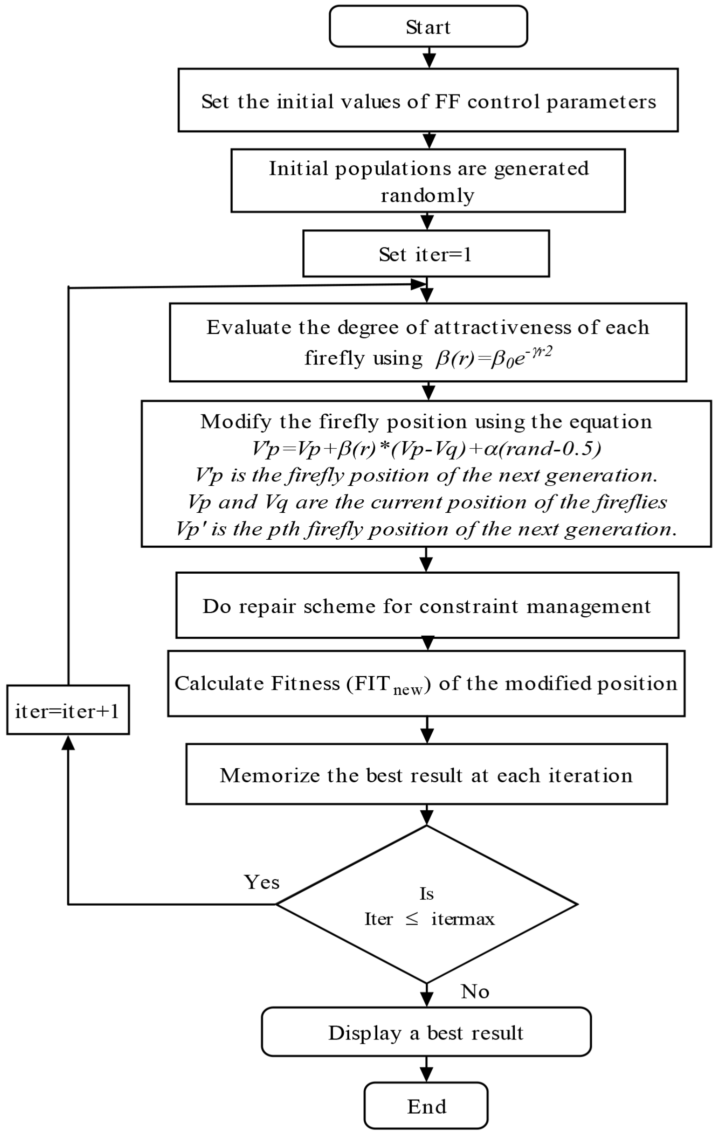

Figure 1. Flow chart for the proposed algorithm.

The main objective of the distributed memory model is to divide the search area into many regions. Every worker owns 1 of the search regions, and they are the responsible for computing the objective function and for holding the optimum states within the region. When a worker computes a new optimum state, first it checks if the obtained best solution is better than the global solution held by the master. If the state is local, the node updates the local optimum solution and goes ahead as usual; otherwise a message containing the best result is sent to the master computer. In this paper, for the distributed memory model the size of the cluster is configured between 10 and 15 nodes. From this cluster, any one of the nodes is assumed as the master node which act like a server. In Figure 2, W_1 is acting as a master node and other nodes are the multiple workers.

4.1. Parallel cluster analysis

To determine the performance of the parallel computing methodology in the FF algorithm, the speedup factor SW_h of the cluster [14–17] is calculated as follows;

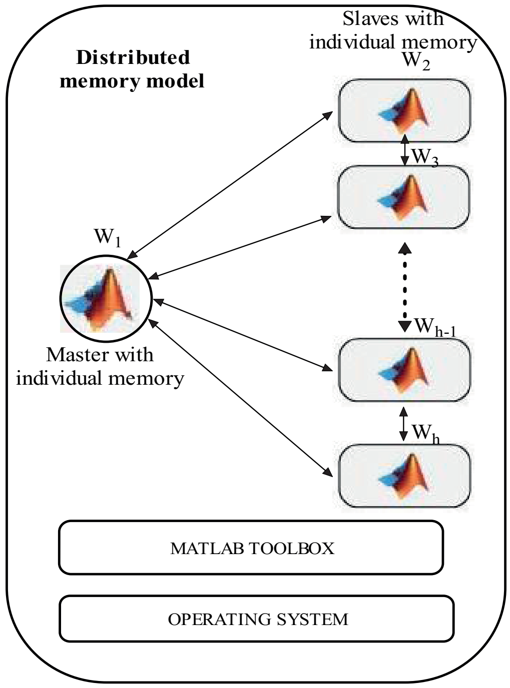

Figure 2. Worker formations in distributed memory model.

$$SW_h = W_t / W_{ht}, \tag{9}$$

where W_t is the execution time of 1 worker and W_{ht} is the execution time of all the workers.

5. Implementation of the firefly algorithm with multiple workers for the unit commitment problem

The sequential steps for the implementation of the FF algorithm with multiple workers for solving the UC problem is described in this section.

Initialize the FF parameter specifications. Specify minimum or maximum generation limits, ramp up/down constraints, and reserve at each h. Initialize the parallel cluster with $W_1, W_2 ... W_h$ as nodes.

Step 1: Initialization of firefly population

The initial solution for the first iteration matrix is obtained by assigning binary digit (1 or 0) for each unit randomly. The dimension of an iteration matrix (IM) depends upon the system data for the given test system. The randomly generated IM is given below,

$$IM = \begin{bmatrix} b_1^1, b_1^2 \cdots \cdots, b_1^d, \cdots b_1^D \\ b_2^1, b_2^2 \cdots \cdots, b_2^d, \cdots b_2^D \\ \cdot \\ \cdot \\ b_m^1, b_m^2 \cdots \cdots, b_m^d, \cdots b_m^D \\ \cdots \\ b_M^1, b_M^2 \cdots \cdots, b_M^d, \cdots b_M^D \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \cdot \\ \cdot \\ B_m \\ \cdot \\ B_M \end{bmatrix} \tag{10}$$

Here D is equal to Ng*T and M represents the size of the firefly population.

Step 2: Initialization of workers

In the distributed memory model, the cluster size is configured between 10 and 15 nodes depending upon the availability of the number of the personal computers (PC). From this cluster, any one node (PC) is taken as the master node and it will act as the cluster file server. For the randomly generated iteration matrix (IM), the worker assigned is shown in Figure 3. Here W_1 is acting as master node. The master node decides the search area for each firefly by firefly sharing policy (FSP). Therefore, the number of fireflies allocated to a worker is given by

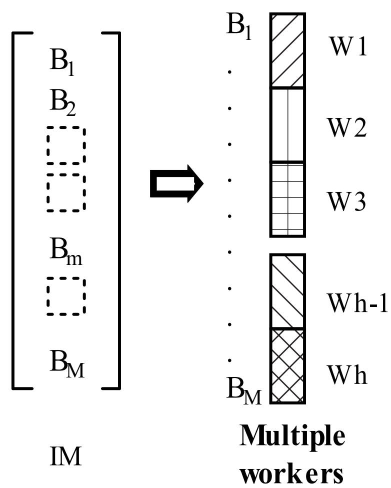

Figure 3. Workers in the distributed memory model.

$$FF_{FSP} = \begin{cases} x, \text{ if } N_{ff} \text{ divisible by } W_h \\ (x+1) \text{ for the first } hx \text{ slaves and } x \text{ for the remaining}, \text{ Otherwise} \end{cases} \tag{11}$$

where

$$x = floor\,(N_{ff} \,/\, W_h) \tag{12}$$

$$hx = N_{ff} - (x * W_h) \tag{13}$$

The master node allocates *(x + 1)* fireflies to the first hx worker in the W_h cluster$(W_1...W_{hx}....W_h)$ and x fireflies to the remaining workers$(W_{hx+1}....W_h)$. Here N_{ff} is the number of fireflies and W_h is the size of the cluster.

Step 3: Constraint management

If the firefly position violates minimum up/down time constraints (5,6), then we randomly commit/decommit the thermal unit based on the minimum up/down times limit to mitigate the violation. Afterwards, the firefly position is checked for spinning reserve (3) constraints. If it violates the limit, then the cheapest units which are in the switch OFF mode are found and the respective unit is committed.

Step 4: Economic dispatch calculation

The economic load dispatch [18,19] is carried out by each worker for the feasible fireflies' positions in the UC problem and constraints (2), (4), and equations (7–8) are checked for violation.

If the firefly position violates the constraints, then the constraint management as shown in Figure 4 is implemented to mitigate the violation. To calculate the power output of each generator, the FF algorithm with multiple workers as mentioned in section 5.1 is solved.

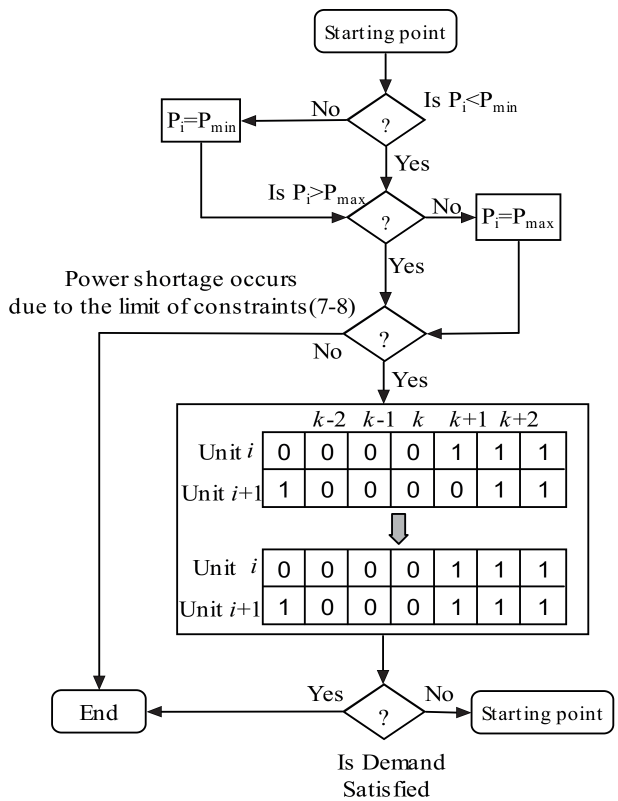

Figure 4. Flowchart for constraint management.

Step 5: Fitness function

Calculate the objective function fuel cost for each firefly position. Then evaluate the fitness value of the firefly position corresponding to the brightness (fuel cost value) of the firefly. The fitness value is determined using the equation (14).

$$\begin{aligned} FIT_p &= 1/Cost_p, && \text{if,} \quad Cost_p > 0 \\ &= 1 + abs(Cost_p), && \text{if} \quad Cost_p < 0 \end{aligned} \tag{14}$$

Step 6: Memorization of the best solution

Whenever a global best solution is selected by the master node, and if the total operating cost is found to be less than the minimum total operating cost computed by the master, then the present solution is captured, or else the previous minimum cost solution is retained.

Step 7: Modification of firefly position by each worker

Based on the brightness of each firefly, the firefly modifies its position using the equation given in Figure 1. Each worker modifies their firefly position simultaneously by considering the brightness in the total number of firefly population as given in Figure 5.

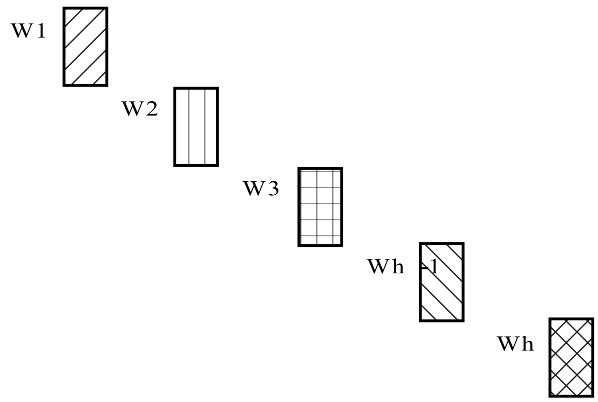

Figure 5. Modification of firefly position by the worker.

Step 8: Increment the iteration count.

Step 9: Continue Step 3 to Step 8 until the number of desired iterations and desired time schedule is completed and output the result.

5.1. Pseudo code to solve economic dispatch problem

The stepwise procedure for solving economic dispatch problem using real coded FF algorithm is given below.

1. *Set the FF algorithm control parameters and read the necessary system data.*
2. *Initial populations are generated randomly as explained in step 1 of section V. It should be noted that the Bm are real values. Also the real values are initialized only for thermal unit that has an ON status obtained by the UC problem.*
3. *Iteration starts: Iter = 1.*
4. *Check for constraints (2, 4, 7–8). If the firefly position violates the constrains, then constraint management is implemented as given in step 4 of section V, and the fitness value is calculated using (14).*
5. *Memorize the best result at the end of the iteration achieved so far.*
6. *For each firefly position produce a new solution.*
7. *Iter = Iter + 1.*
8. *Until Iter = maximum number of iteration, go to step 4.*
9. *Output the best result.*
10. *End.*

The flowchart of the proposed method is shown in Figure 6.

6. Case study

The parallel computing for the multiple workers is implemented using a distributed memory model with a distribution using MATLAB computing toolbox. In a distributed memory model, the maximum size of the cluster is considered as 15 workers/processors (Pentium – IV 3.40 GHz, 1 GB RAM) to evaluate the optimum solution. To validate the proposed algorithm, 3 different test systems such as a practical Taiwan 38 bus power system, an IEEE 118 bus system, and a 100 unit system were considered for solving UC problem for 24 h. The obtained best solutions were validated and compared with the best solutions published in the existing literature.

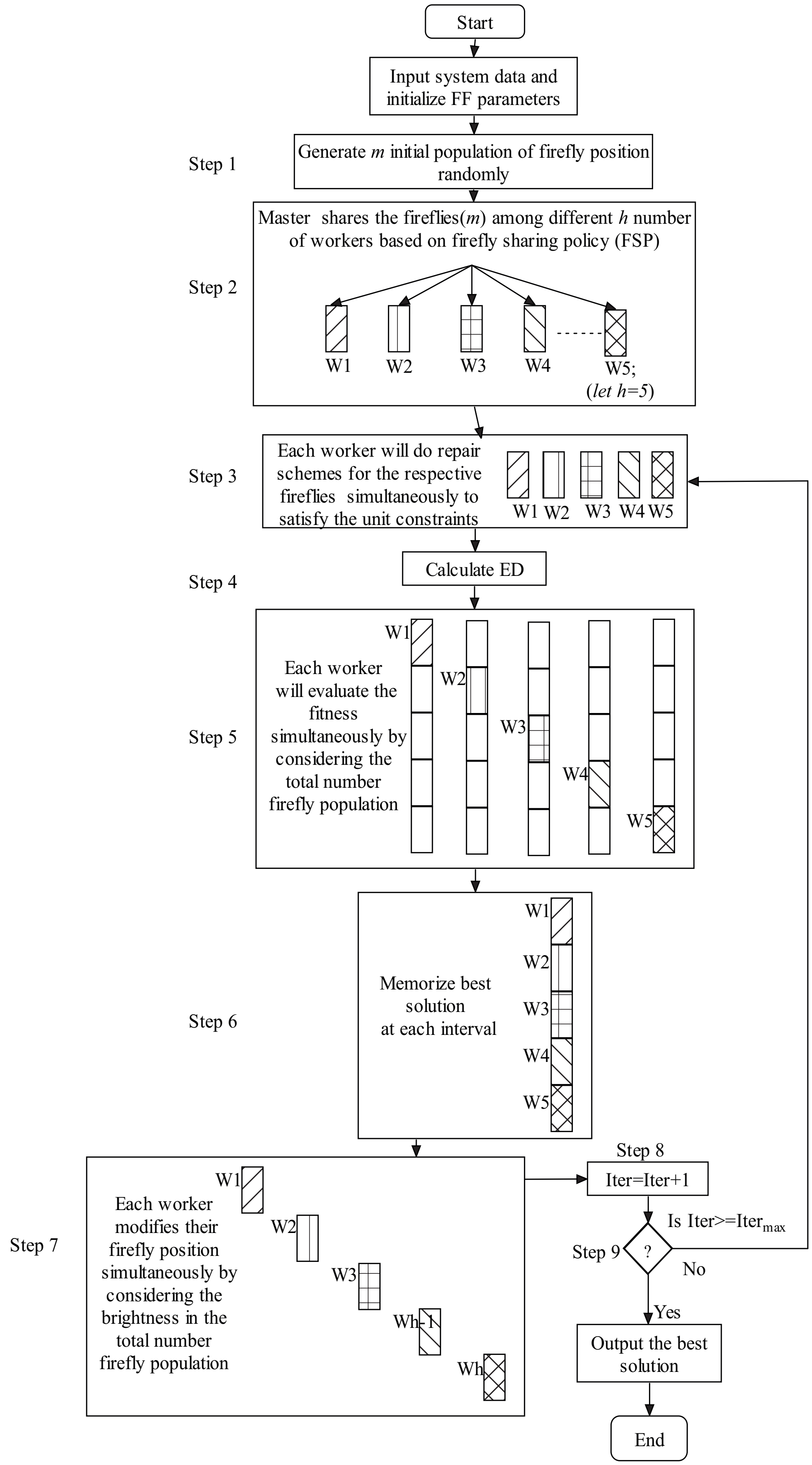

Figure 6. Flowchart for the proposed methodology.

6.1. Test System 1: 100 unit system

A power system with 100 units was presented to evaluate the performance of the proposed methodology. The system input data are available in [20]. In Table 1, the best solution of the unit status and hourly cost as obtained from 20 trials runs of the proposed FF algorithm with multiple workers for the 100 unit system are given.

Table 1. Unit commitment solution - Test system 1.

Hour	Unit 1–100	Fuel cost $
1	1111111111111111111100000000000000000000000000000000000000000000000000000000000000000000000000000000	1,36,830.00
2	1111111111111111111100000000000000000000010000000000000000000000000000000000000000000000000000000000	1,469,50.00
3	1111111111111111111100000000000000000000110000000000000000000000000000000000000000000000000000000000	164,930.00
4	1111111111111111111100000000000000000000111111100100000000000000000000000000000000000000000000000000	191,250.00
5	1111111111111111111100000000000000000011111111111100000000000000000000000000000000000000000000000000	199,860.00
6	1111111111111111111100000000001111111111111111111100000000000000000000000000000000000000000000000000	227,980.00
7	1111111111111111111111001110001111111111111111111100000000000000000000000000000000000000000000000000	235,540.00
8	1111111111111111111111111111111111111111111111111100000000000000000000000000000000000000000000000000	247,000.00
9	1111111111111111111111111111111111111111111111111101011111111110000000000000000000000000000000000000	272,520.00
10	1111111111111111111111111111111111111111111111111111111111111111111111111111100000000000000000000000	303,300.00
11	1111111111111111111111111111111111111111111111111111111111111111111111111111111111111110000000000000	318,350.00
12	1111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111100	338,130.00
13	1111111111111111111111111111111111111111111111111111111111111111111111001111111100000000000000000000	299,190.00
14	1111111111111111111111111111111111111111111111111111111111111110000000000000000000000000000000000000	267,870.00
15	1111111111111111111111100000001111111111111111111111111111110000000000000000000000000000000000000000	243,800.00
16	1111111111111111111111100000001111111111111111111000000000000000000000000000000000000000000000000000	210,800.00
17	1111111111111111111111100000001111111111111111111000000000000000000000000000000000000000000000000000	202,050.00
18	1111111111111111111111100000001111111111111111111000000000000000000000000000000000000000000000000000	219,560.00
19	1111111111111111111111100000001111111111111111111111111100000000111000000000000000000000000000000000	246,530.00
20	1111111111111111111111111111111111111111111111111111111111110000111000111111111111111111000000000000	308,240.00
21	1111111111111111111111111111111111111111111111111111111111110000111000000000000000000000000000000000	267,870.00
22	1111111111111111111100011111100000001100111111111100000011100000000000000000000000000000000000000000	219,830.00
23	1111111111111111111100011111100000000000000000000000000000000000000000000000000000000000000000000000	176,090.00
24	1111111111111111110000011111100000000000000000000000000000000000000000000000000000000000000000000000	156,750.00
	Total ($)	5,601,220.00

In Table 2, the proposed results are compared with the previously reported results both in cost and execution time. From Table 2, it is clear that the minimum cost produced by the proposed algorithm is significantly less than that of other methods in the literature. It should also be observed that the program execution time of the proposed techniques is less than that of the other convention methods (ALR [21], DPLR [21], ALR [21], ELR [21], MILP [21]) and a population based approach. This shows the advantage of the proposed algorithm.

6.1.1. Parameter setting analysis

The optimal result and the performance of the heuristic algorithm usually vary based on the initial value of the control parameter set by the different user. Hence, the 100 unit system was solved here 10 times with different parameter values for analyzing average performance of the proposed method. In each run of all the systems a firefly population of 200 solutions is allowed to evolve for 1000 iterations. The 3 control parameters of the FF algorithm, β_0, γ, n, were determined before its implementation. Feasible boundary limits of control parameters for the FF algorithm were adopted from the reference papers [12] and are given in Table 3. The FF parameter that is finally set to obtain the best result for a 100 unit system is $\gamma = 2, \beta_0 = 0.45$, and $n = 0.6$.

Table 2. Comparison of result - 100 unit system.

Sl. no	Solution techniques	Minimum cost ($)	Average computation time (s)
1	Dynamic programming Lagrangian relaxation [21]	5,640,488	12,437.00
2	Binary coded genetic algorithm [21]	5,637,930	397.00
3	Evolutionary programming [21]	5,623,885	6120.00
4	Integer coded genetic algorithm [21]	5,630,838	242.5.00
5	Genetic algorithm [21]	5,627,437	15,733.00
6	GA based on unit characteristic [21]	5,626,514	3547.00
7	Particle swarm optimization [21]	5,625,376	6250.00
8	Particle swarm optimization - Lagrangian relaxation [21]	5,623,607	856.00
9	Improved particle swarm optimization [21]	5,619,284	5750.00
10	Augumented Lagrangian relaxation [21]	5,615,893	167.00
11	Stochastic priority list [21]	5,615,530	374.03
12	Lagrangian relaxation- genetic algorithm [21]	5,613,127	4045.00
13	Matrix real-coded genetic algorithm [21]	5,610,031	260.50
14	Extended priority list [21]	5,608,440	-
15	Priority list-based evolutionary algorithm [21]	5,607,904	-
16	Enhanced adaptive Lagrangian relaxation [21]	5,605,678	345.00
17	Mixed integer linear programming [21]	5,605,189	123.00
18	Firefly algorithm tuned fuzzy design [21]	5,601,298	945
19	**Proposed method**	**5,601,220**	**113**

Table 3. Feasible boundary limits of control parameters.

Control parameter	Limits
-Absorption coefficient, γ	0–10
Initial attractiveness, β_0	0–1
Constant, n	0.1–3

6.1.2. Parallel cluster analysis

In a parallel cluster environment, the solution quality and execution time depend on the number of workers assigning to do the work. Hence it is necessary to determine the optimal number of workers to complete the particular task. Therefore, an analysis was carried out to identify the optimal workers for the 100 unit system to obtain the best result. Once the number of workers was identified, then the same number of workers can be used for any change in the system load and reserve data. The variation in total cost, execution time, and speedup factor with respect to the number of workers is given in Table 4.

Table 4. Parallel cluster analysis - 100 unit system.

No. of workers	Total cost $	Exe time, s	Speedup factor
2	5,602,785	401	1.97
3	5,602,711	253	3.13
4	5,602,045	201	3.94
5	5,601,277	171	4.63
6	5,601,261	143	5.53
7	**5,601,220**	**113**	**7**
8	5,601,221	102	7.76
9	5,601,230	87	9.1
10	5,601,231	76	10.42

From Table 4, it is evident that, for the 100 unit system with 7 workers, better optimal total cost was obtained with the least execution time. On further increment of workers, there is no significant improvement in the total cost, but the execution time and speedup factor are improved.

For the 100 unit system with 7 workers, the balance between exploration and exploitation is found to be good in the FF algorithm. On further incrementing the workers, the balance is not maintained and the result moves toward a suboptimal value.

The convergence graph for the 100 unit system with different workers is shown in Figure 7. Due to the better balance between exploration and exploitation, the convergence graph for the 100 unit system with 7 workers is found to be best when compared with others.

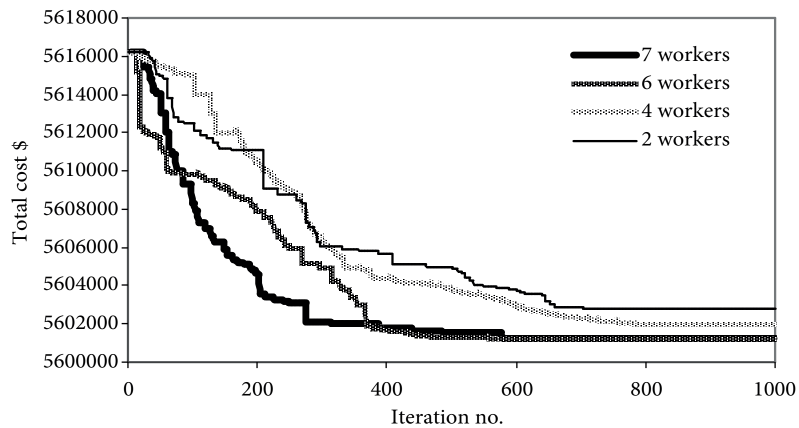

Figure 7. Convergence graph of the firefly algorithm - 100 unit system.

6.1.3. Performance analysis

The effectiveness of the proposed method was validated by conducting and statistically observing over 20 trials, the average cost, standard deviation, and rate of recurrence of attaining minimum operating cost. The statistical result shows that the obtained result is better than average cost, which is given in Table 5. This shows the robustness of the proposed method.

Table 5. Performance analysis - 100 unit system.

Average cost $	5,601,221.20
Standard deviation	2.14
Rate of recurrence of attaining minimum operating cost is better than the average cost	18

Also, for the 100 unit system the nonparametric test was implemented and the results produced. This test was implemented to verify the significance of the proposed methodology for the UC problem with respect to different workers.

In order to verify the significant improvements of the cost achieved between 6 and the 7 workers for solving UC problem in the 100 unit system, a Kruskal–Wallis one-way nonparametric analysis ($one-way$ ANOVA *statistical* test) that is available in the MATLAB was implemented. The results are presented in Table 6. The P value obtained is 6.57862×10^{-6}. Since the P values are very small, it can be concluded that the cost obtained by the 7 workers is significant compared to 6 workers.

Also the performance analysis was carried out based on the solution quality and computational efficiency. The UC problem was solved for 100 units test system using standard GA, PSO, and FF algorithms. It will be genuine if the initial random population generated by all the heuristics techniques is the same during comparison analysis. Therefore, the initial randomly generated population is kept the same for all the 4 techniques (GA, PSO, FF, and FF algorithms with multiple workers). The comparison of result is given in Table 7. Here, the proposed multiple worker FF techniques for the UC problem is capable of achieving better quality of solution with less computational time than standard GA, PSO, and FF.

Table 6. ANOVA results.

Source	Sums of squares (SS)	Degrees-of-freedom (df)	Mean squares (MS)	Statistics (F)	P values. Prob>F
Columns	1.79928×10^{10}	1	1.79928×10^{10}	25.6	6.57862×10^{-6}
Error	3.37371×10^{10}	48	7.02856×10^{8}		
Total	5.17299×10^{10}	49			

Table 7. Comparison of results - 100 unit system.

Solution techniques	Minimum operating cost ($)	Execution time (s)
Genetic algorithm	5,601,843.00	982
Particle swarm optimization	5,601,520.00	1041
Firefly algorithm	5,601,298.00	945
Proposed firefly methodology	5,601,220.00	113

6.2. Test system 2: IEEE 118 bus system

The test system 2 consisted of 54 generating units. From the reference [21], the generator data and load profile for IEEE 118 bus system is adapted. Out of 20 trials, the total cost and comparison of results with existing methods are given in Table 8. Similar to the section 6.1.1, the parameter setting was carried out and the optimal parameters to obtain the best result was given as $\gamma = 2.5$, $\beta_0 = 0.6$, and n = 1.3. Also for the IEEE 118 bus system, the balance between exploration and exploitation is found to be good in the FF algorithm with 6 workers. From Table 8, it is clear that the proposed method is capable of giving a better solution compared with the solutions available in the literature. The generating unit status and hourly total cost of 54 units using the proposed method are given in Table 9.

Table 8. Comparison of results - Test system 2.

Solution techniques	Total cost $
Semi-definite programming [21]	1,645,445.00
Artificial bee colony- Lagrangian relaxation [21]	1,644,269.00
BRCFF [21]	1,644,141.00
Proposed method	1,644,134.00

6.3. Test system 3: 38 unit Taiwan power system

The Taiwan Power (Taipower) 38-unit system is adapted from [21]. The UC problem was executed under the same conditions as in the reference [21]. Table 10 provides the comparison of the total operating cost obtained using proposed method with respect to other techniques available in the literature.

Similar to the section 6.1.1, the parameter setting was carried out and the optimal parameters to obtain the best result was given as $\gamma = 4$, $\beta_0 = 0.65$, and n = 1.8. Also for the 38-unit system, the optimal number of workers was found to be 6 and the best cost obtained was $195,914,900.00. The generation dispatch and hourly total cost of the Taiwan power system are given in Table 11.

Table 9. UC solution - Test system 2.

Hour	Unit no.	Fuel cost $
1	111111111111111110000000000000000000000000000000000000	58,179
2	111111111111111110000000000000000000000000000000000000	54,094
3	111111111111111110000000000000000000000000000000000000	46,270
4	111111111111111110000000000000000000000000000000000000	30,727
5	111111111111111110000000000000000000000000000000000000	38,978
6	111111111111111111000000000000000000000000000000000000	48,292
7	111111111111111111000010000000000000000000000000000000	58,296
8	111111111111111111010010000000000000000000000000000000	66,834
9	111111111111111111010010001000000000000000000000000000	71,297
10	111111111111111111010110011001110000000000000000000000	78,241
11	111111111111111111010110011001110000000000000000000000	79,137
12	111111111111111111010100011001110000000000000000000000	73,469
13	111111111111111111010100011001110000000000000000000000	69,011
14	111111111111111111010100010001110000000000000000000000	64,664
15	111111111111111111010000010111111000000000000000000000	78,160
16	111111111111111111010001110111111000000000000000000000	80,363
17	111111111111111111010001110111111000000000000000000000	74,597
18	111111111111111111011011110111111000000000000000000000	79,200
19	111111111111111111011011111111111000000000000000000000	84,835
20	111111111111111111111111111111111000000000000000000000	89,445
21	111111111111111111111111111111111000000000000000000000	91,674
22	111111111111111111011110111001010000000000000000000000	80,263
23	111111111111111111100110011100100000000000000000000000	76,868
24	111111111111111111010010001010010000000000000000000000	71,240
	Total cost $	1,644,134

Table 10. Comparison of results – Taipower system.

Solution techniques	Total cost (M $)
Dynamic programming [21]	210.50
Lagrangian relaxation [21]	209.00
Constraint logic programming [21]	208.10
Simulated annealing [21]	207.80
Fuzzy optimization [21]	207.80
Matrix real-coded genetic algorithm [21]	204.60
Memory-bounded ant colony optimization [21]	200.46
Twofold simulated annealing [21]	197.98
Enhanced merit order-ALHN [21]	197.50
Heuristics and absolutely stochastic SA [21]	196.96
Fuzzy adaptive particle swarm optimization [21]	196.73
Absolutely stochastic simulated annealing [21]	196.70
BRCFF [21]	195.916304
Proposed method	**195.914900**

Table 11. UC solution -Test system 3.

Hour	Unit no.																													Fuel cost $
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27-36	37	38	
1	550	550	500	500	500	500	500	500	0	0	0	0	0	0	0	0	184.1	65	65	272	272	260	190	0	125	110	0	29.27	27.61	6,882,400.00
2	490.3	490.3	493.5	493.5	493.5	493.5	493.5	493.5	0	0	0	0	0	0	0	0	167.3	65	0	272	272	260	151.9	0	125	97.91	0	24.14	23.12	6,248,400.00
3	469	469	472.2	472.2	472.2	472.2	472.2	472.2	0	0	0	0	0	0	0	0	164.7	0	0	272	272	260	144.8	0	125	94.62	0	23.35	22.43	5,668,900.00
4	453.7	453.7	456.9	456.9	456.9	456.9	456.9	456.9	0	0	0	0	0	0	0	0	0	0	0	272	272	260	139.7	0	120.8	92.26	0	22.78	21.94	5,221,600.00
5	446.7	446.7	449.8	449.8	449.8	449.8	449.8	449.8	0	0	0	0	0	0	0	0	162	0	0	272	272	260	137.3	0	118.8	91.18	0	22.53	21.71	5,475,100.00
6	448.1	448.1	451.2	451.2	451.2	451.2	451.2	451.2	0	0	0	0	0	0	0	0	0	0	0	272	272	260	137.8	0	119.2	91.39	0	22.58	21.76	5,167,600.00
7	453.7	453.7	456.9	456.9	456.9	456.9	456.9	456.9	0	0	0	0	0	0	0	0	0	0	0	272	272	260	139.7	0	120.8	92.26	0	22.78	21.94	5,221,600.00
8	482.9	482.9	486	486	486	486	486	486	0	129.9	0	0	0	0	0	0	166.4	0	0	272	272	260	149.4	0	125	96.76	0	23.86	22.88	6,529,600.00
9	550	550	500	500	500	500	500	500	0	310.6	313.4	332.5	124.8	0	0	0	215.8	0	0	272	272	260	190	0	125	110	0	38	36.07	9,430,100.00
10	550	550	500	500	500	500	500	500	396.7	396.7	394.4	418.5	269.4	173.4	156.7	0	239.3	0	0	272	272	260	190	0	125	110	0	38	38	11,436,000.00
11	550	550	500	500	500	500	500	500	414.3	414.3	411	436.1	299	197.6	178.6	0	244.2	0	0	272	272	260	190	0	125	110	0	38	38	10,107,000.00
12	550	550	500	500	500	500	500	500	424.4	424.4	420.5	446.1	315.9	211.4	191.2	0	246.9	0	0	272	272	260	190	14.22	125	110	0	38	38	10,376,000.00
13	550	550	500	500	500	500	500	500	249.1	249.1	255.5	271	110	90	82	0	199	0	0	272	272	260	190	0	125	110	0	33.8	31.58	8,564,700.00
14	550	550	500	500	500	500	500	500	424.3	424.3	420.4	446	315.8	211.3	191.1	0	246.9	65	0	272	272	260	190	0	125	110	0	38	38	10,635,000.00
15	550	550	500	500	500	500	500	500	436	436	431.5	457.8	335.5	227.5	205.7	0	250.1	65	0	272	272	260	190	0	125	110	0	38	38	10,764,000.00
16	550	550	500	500	500	500	500	500	414.3	414.3	411	436.1	299	197.6	178.6	0	244.2	0	0	272	272	260	190	0	125	110	0	38	38	10,107,000.00
17	550	550	500	500	500	500	500	500	390.8	390.8	388.9	412.6	259.5	165.3	149.4	0	237.7	0	0	272	272	260	190	0	125	110	0	38	38	9,810,100.00
18	550	550	500	500	500	500	500	500	359.9	359.9	359.8	381.7	0	122.8	111	0	0	0	0	272	272	260	190	0	125	110	0	38	38	8,678,900.00
19	550	550	500	500	500	500	500	500	300.1	300.1	303.5	322	0	90	82	0	0	0	0	272	272	260	190	0	125	110	0	38	35.31	8,262,500.00
20	550	550	500	500	500	500	500	500	390.4	390.4	388.5	412.2	0	164.7	148.9	0	0	0	0	272	272	260	190	0	125	110	0	38	38	8,968,200.00
21	550	550	500	500	500	500	500	500	359.9	359.9	359.8	381.7	0	122.8	111	0	0	0	0	272	272	260	190	0	125	110	0	38	38	8,678,900.00
22	550	550	500	500	500	500	500	500	290.7	290.7	294.7	312.6	0	0	0	0	210.4	0	0	272	272	260	190	0	125	110	0	37.28	34.62	8,229,600.00
23	550	550	500	500	500	500	500	500	281.1	281.1	285.6	303	0	0	0	0	0	0	0	272	272	260	190	0	125	110	0	36.47	33.92	7,798,800.00
24	550	550	500	500	500	500	500	500	273.1	0	278.1	295.1	0	0	0	0	205.6	0	0	272	272	260	190	0	125	110	0	35.81	33.34	7,652,900.00
	Total cost $																													195,914,900.00

7. Conclusion

This paper proposed a parallel computing firefly algorithm using a message passing interface protocol in a distributed memory model. The algorithm has been successfully implemented to a UC problem for benchmark test systems and a practical Taiwan power system. From the analysis, it is noted that the multiple worker model significantly reduced the program execution time and also improved the quality of solution. In the present scenario, parallel computing with multiple workers with distributed memory models are found to be an alternative viable option. Further, ANOVA is carried out to show the significant improvements in the cost achieved by different clusters. From the results and discussion, it can be concluded that the proposed method paves the way for future fast computing technologies and helps the power system operators to take efficient decision in lesser time for effective operation of the system.

References

[1] Frangioni A, Gentile C, Lacalandra F. Sequential Lagrangian-MILP approaches for unit commitment problems. Int J Elec Power 2011; 33: 585-593.

[2] Siu TK, Nash GA, Shawwash ZK. A practical hydro, dynamic unit commitment and loading model. IEEE T Power Syst 2001; 16: 301-306.

[3] Marcovecchio MG, Novais AQ, Grossmann IE. Deterministic optimization of the thermal unit commitment problem: a branch and cut search. Comput Chem Eng 2014; 67: 53-68.

[4] Cheng CP, Liu CW, Liu CC. Unit commitment by Lagrangian relaxation and genetic algorithms. IEEE T Power Syst 2000; 15: 707-714.

[5] Cohen AI, Yoshimura M. A branch-and-bound algorithm for unit commitment. IEEE T Power Ap Syst 1983; PAS-102: 444-451.

[6] Mantawy AH, Abdel-Magid YL, Selim SZ. Integrating genetic algorithms, tabu search, and simulated annealing for the unit commitment problem. Power Systems, IEEE T Power Syst 1999; 14: 829-836.

[7] Dudek G. Genetic algorithm with binary representation of generating unit start-up and shut-down times for the unit commitment problem. Expert Syst Appl 2013; 40: 6080-6086.

[8] Saber AY, Senjyu T, Miyagi T, Urasaki N, Funabashi T. Fuzzy unit commitment scheduling using absolutely stochastic simulated annealing. IEEE T Power Syst 2006; 21: 955-964.

[9] Yuan X, Nie H, Su A, Wang L, Yuan Y. An improved binary particle swarm optimization for unit commitment problem. Expert Syst Appl 2009; 36: 8049-8055.

[10] Duo H, Sasaki H, Nagata T, Fujita H. Solution for unit commitment using Lagrangian relaxation combined with evolutionary programming. Electr Pow Syst Res 1999; 51: 71-77.

[11] Jahromi MZ, Mehdi M, Bioki H. Solution to the unit commitment problem using an artificial neural network. Turk J Electr Eng Co 2013; 21: 198-212.

[12] Yang XS. Firefly algorithms for multimodal optimization. Lect Notes Comput Sc 2009; 5792: 169-178.

[13] Mohammadi S, Mozafari B, Soleymani S. Optimal operation management of microgrids using the point estimate method and firefly algorithm while considering uncertainty. Turk J Electr Eng & Comp Sci 2014; 22: 735-753.

[14] Bosilca G, Bouteiller A, Danalis A, Herault T, Lemarinier P, Dongarra J. DAGuE: A generic distributed DAG engine for high performance computing. Parallel Comput 2012; 38: 37-51.

[15] Chakroun I, Melab N, Mezmaz M, Tuyttens D. Combining multi-core and GPU computing for solving combinatorial optimization problems. J Parallel Distr Com 2013; 73: 1563-1577.

[16] Tuominen J, Naghdy F, Tabrizi N. Generation of a parallel computing platform for a distributed manipulation environment. Control Eng Pract 1993; 1: 873-880.

[17] Christopher Columbus C, Simon SP. Profit based unit commitment: A parallel ABC approach using a workstation cluster. Comput Electr Eng 2012; 38: 724-745.

[18] Koodalsamy C, Simon SP. Fuzzified artificial bee colony algorithm for nonsmooth and nonconvex multiobjective economic dispatch problem. Turk J Electr Eng & Comp Sci 2013; 21: 1995-2014.

[19] Slimani L, Bouktir T. Economic power dispatch of power systems with pollution control using artificial bee colony optimization. Turk J Electr Eng & Comp Sci 2013; 21: 1515-1527.

[20] Carrión M, Arroyo JM. A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. IEEE T Power Syst 2006; 21: 1371-1378.

[21] Chandrasekaran K, Simon SP. Optimal deviation based firefly algorithm tuned fuzzy design for multi-objective UCP. IEEE T Power Syst 2013; 28: 460-471.