Unit Commitment Problem Solution Using Seeker Optimization Algorithm

Arsalan Najafi

Abstract-- Unit commitment (UC), a critical power system operation problem, involves high complexity due to numerous constraints. This paper presents a novel approach using the seeker optimization algorithm (SOA), an evolutionary method inspired by human search behavior. The proposed SOA employs integer coding to efficiently handle minimum up/down constraints and binary coding for spinning reserve requirements. By leveraging population-based search, SOA ensures optimal convergence while addressing UC's combinatorial challenges. Simulations across diverse test systems demonstrate algorithm superiority in convergence speed and solution quality compared to conventional methods. The integer-coded framework enhances computational efficiency, particularly in managing unit minimum uptime/downtime constraints. Results implemented on 10 to 100 generating units validate SOA's effectiveness in optimizing large-scale UC problems, compared to the other existing approaches, offering a reliable tool for power system operators. This approach bridges the gap between metaheuristic adaptability and UC's rigid constraints, paving the way for cost-effective and operationally feasible solutions.

Keywords: Economic load dispatch, seeker optimization algorithm, unit commitment.

I. Introduction

UNIT commitment (UC) has great importance in power system operation concerns. UC is a process that determines the optimum daily schedule of units in economical situation and operation constraint [1, 2]. Its objective is to schedule the generating units online or offline over a scheduling horizon. This problem includes the start up, shut down and fuel cost in 24 hours, seven days or maybe 1 month. First, ON/OFF states of units is determined and then, the amount of unit's generation and fuel cost are calculated using economic dispatch. In the UC problem, the decision variables are the selection of the time for each unit to be on and/or off. UC is a complex problem that has various discrete and continues variables. The power industry has used optimization techniques for solving unit commitment problem for many years. Therefore, millions of dollars were saved in year [3].

Before 1960's, UC problem was limited to economic dispatch. In those times Kuhn-Tucker declared the conditions that characterized the optimal economic status. When these conditions are satisfied, all committed units were loaded according to their fuel, except those efficient units which were loaded on their peak power. Later, the economic dispatch was developed by piecewise approximation of its cost function. Subsequently, a linear optimization method was suggested for UC problem [4]. Dynamic programming could solve the unit commitment efficiently [5]. Lagrange method was another way to solve the UC problem. By adding equal and unequal constraint to the objective function, constrains of the problem were relaxed [6]. Branch-and-bound (B&B) methods [7] are other deterministic methods which have been used to solve UC problem.

Evolutionary methods have also been utilized to solve the UC problem. These methods have attracted much attention because of their potential to reach global solutions. In [8] a method based on genetic algorithm (GA) has been suggested and done on a sample case. An integer coded genetic algorithm (ICGA) has been used in [9]. An evolutionary method based on shuffled frog leaping algorithm (SFLA) is proposed in [10]. This method has used integer coding for UC's solutions and penalty factor in order to satisfy some of constraints. Binary methods have been suggested in [11] and [12]. [13] has proposed a two-level hierarchical approach which combines an expert system with an elite particle swarm optimization to deal with the UC problem. In this method, execution time has been increased linearly. [14] has suggested a mixed-integer linear programming (MILP) reformulation of the UC problem. The proposed formulation is simultaneously tight and compact. This method reduces the search space by tighter search space, and more compact characteristic increases the searching speed by searching in the reduced space. Gravitational Search Algorithm (GSA) has suggested in [15]. GSA is a cooperative agents' approach which is inspired by the observation of the behaviors of all the masses present in the universe due to gravitation force. It has succeeded to solve UC problem by gravitation force significance. Authors in [16] have formulated a UC optimization problem for renewable energy sources distributed in a micro-grid formed by a complex of intelligent

Manuscript received: 16 July 2025, revised: 13 december 2025, accepted 26 may 2026, published online 20 september 2026.
DOI: 10.22077/ijpe.2026.9728.1024
A. N. is with the Department of Architecture and Civil Engineering, Chalmers University of Technology, Gothenburg, Sweden (arsalan.najafi@chalmers.se).

buildings of both office and residential characters, including a wide range of amenities. The related simulation has conducted on a sample micro-grid. A method based on particle swarm optimization (PSO) and concept of time varying coefficients has also been suggested in [17]. It can be pointed at evolutionary programming (EP) [18], simulated annealing (SA) [19], bacterial foraging (BF) [20], imperialistic competition algorithm (ICA) [21], deterministic annular crossover genetic algorithm (DACGA) [22], Harmony Search [23], primary unit commitment-modification process [24], semi definite programming relaxation based technique combined with selective pruning [25], mixed integer quadratically constrained program [26] and Tabu search [27] among the other methods that have been achieved to solve the UC problem. As a combination method, fuzzy dynamic programming has been suggested in UC problem [28]. Recently, the studies have investigated the other aspects of UC problem rather than facing the complexity of solving it. The commitment of distributed generations in order to assist in frequency control has been explored in [29]. High penetration of various types of renewable energy resources have been investigated in [30], [31], and [32]. The impact of other energy carries in UC problem has been studied in [33].

In this paper a new heuristic method based on seeker optimization algorithm (SOA) is proposed. In order to achieve better satisfaction of the UC's constraints, a new integer/binary coding has also been utilized in the proposed method. Simulation results in some sample cases show the ability of the proposed method in comparing with the other methods.

II. Problem Formulation

A. Objective Function

Total production cost in UC problem consists of fuel costs, start up costs and shut down costs that must be minimized in duration of planning. Therefore, the objective function is expressed by [25]:

$$\min \sum_{t=1}^{T} \sum_{i=1}^{NG} (FC_i(P_i^t) \quad u_i^t) + SU_i^t + SD_i^t \tag{1}$$

where P_i^t is the power generation of unit i at hour t, NG is the number of generating units, T is duration time of planning, FC_i is the fuel cost of unit i, SU_i and SD_i are the start up and shut down cost of unit i, respectively. u_i^t also shows the state of unit i at hour t which is equal to zero or one. Fuel cost is the main part of the mentioned equation and it is calculated as follows:

$$FC_i(P_i^t) = a_i (P_i^t)^2 + b_i P_i^t + c_i \tag{2}$$

where a_i, b_i and c_i are fuel cost coefficient of unit i.

Shut down cost is a small term against the other costs and therefore, it is neglected usually. Start up cost mathematically is as below [9]:

$$SU_i = \begin{cases} H_{\text{startup}} & T_i^{OFF} \le MDT_i + (T_{\text{cold}}) \\ C_{\text{startup}} & T_i^{OFF} > MDT_i + (T_{\text{cold}}) \end{cases} \tag{3}$$

where $H_{\text{start up}}$ and $C_{\text{start up}}$ are hot and cold start up cost, respectively. MDT_i, is minimum down time of unit i and T_{cold} determines the time of cold or hot start up.

B. Constraints

The constraints that must be considered in the UC problem are as follows:

- **Minimum up and down constraints**

  Units cannot be turned on or off immediately and needs to be on or off for specific hours. Minimum up / down times are as below:

$$\begin{cases} T_i^{ON} \ge MUT_i \\ T_i^{OFF} \ge MDT_i \end{cases} \quad \text{for } i = 1,2,..,NG \tag{4}$$

  Where T_i^{ON} and T_i^{OFF} are the durations that unit i remains on or off, respectively and MUT_i is the minimum up time of unit i.

- **System power balance constraint**

  Total output generation of units must be equal to demand as below:

$$\sum_{i=1}^{NG} u_i^t P_i^t = P_d^t \quad \text{for } t = 1,2,...,T \tag{5}$$

  where P_d^t is the total system demand at hour t.

- **Generator output limitation**

  Output generation of units must be in their maximum and minimum limitation as below:

$$P_{i\min} \le P_i \le P_{i\max} \quad \text{for } i = 1,2,...,NG \tag{6}$$

  where $P_{i\min}$ and $P_{i\max}$ are minimum and maximum generation limit of unit i, respectively.

- **Spinning reserve constraint**

  In order to minimize the probability of load interruption, spinning reserve must be available in power system. It can be specified in terms of excess megawatt capacity that is expressed by:

$$\sum_{i=1}^{NG} u_i^t \quad P_{i\max}^t \ge P_d^t + SR^t \tag{7}$$

  where SR^t is the amount of spinning reserve at hour t.

- **Ramp up and down rate constraints**

  Because of production limits, increasing or decreasing of power couldn't be higher or lower than specific amount. This amount is one of the generator characters and is a security constraint for generator that prevents from damaging rotor. These constraints determine the maximum change slop of generator output. These constraints are as follows [34]:

$$P_{i\max}(t) = \min\left\{P_{i\max}, P_i^{t-1} + \tau \cdot RU_i\right\}$$
$$P_{i\min}(t) = \max\left\{P_{i\min}, P_i^{t-1} - \tau \cdot RD_i\right\} \quad (8)$$

where RU_i and RD_i are the ramp up and down rates of unit *i*, respectively.

III. SEEKER OPTIMIZATION ALGORITHM

Seeker optimization algorithm is one of the population heuristic based algorithms [35]. SOA operates on a set of solutions called search population. The individual of this population is called seeker. In order to exchange the information, a neighborhood is defined for each seeker. The population is divided into subpopulations randomly. The number of subpopulation is three in this study.

A. Solution Updating

In SOA, a search direction $d_{kj}(n)$ and a step length $\alpha_{kj}(n)$ are computed separately for each seeker on each iteration *n*. D_{kj} has values of -1, 1 or zero. $d_{kj}=1$, means that the seeker goes to positive direction of coordinate axis, $d_{kj}=-1$ shows that it goes toward negative direction and $d_{kj}=0$ depicts that the seeker remains in its position. Positions of seekers are updated as follows:

$$x_{kj}(n+1) = x_{kj}(n) + \alpha_{kj}(n) d_{kj}(n) \quad (9)$$

B. Search Direction

The search space may be seen as a gradient field, and a so-called empirical gradient (EG) can be determined by evaluating the response to the changing of the position [30]. Seekers have two types of cooperative behavior in the population. One, egotistic is entirely pro-self and another is altruistic that is affected from group behavior. Each seeker *k*, as a single sophisticated operator, is uniformly egotistic, with the belief that it should go toward its historical best position $P_{k,best}(n)$ through cognitive learning. This behavior of seekers can be modeled by an experimental vector $d_{k,ego}(t)$ as follows:

$$D_{k,ego}(n) = sign(P_{k,best}(n) - X_k(n)) \quad (10)$$

where *sign* is a sign function.

As it was mentioned before, the other attitude of seekers is altruistic behavior. In this manner seeker exchanges its information in the subpopulations with other seekers. Neighbors' historical best position (g_{best}) and current best position (l_{best}) are attraction signals of the population. Therefore, two equations are introduced as the altruistic directions. The first declares that the seekers go toward the historical best position $D_{k,alt1}(n)$, and the second $D_{k,alt2}(n)$ shows the seekers' tendency to the current best position. These relations are as follows:

$$D_{k,alt1}(n) = sign(G_{best}(n) - X_k(n)) \quad (11)$$

$$D_{k,alt2}(n) = sign(L_{best}(n) - X_k(n)) \quad (12)$$

In addition, future behavior of the seekers can be predicted and leaded by past behavior. As a result, the seeker may be proactive to change its search direction and exhibit a goal-directed behavior according to its past behavior. Hence, each seeker *k* is associated with an empirical direction called pro-activeness direction $D_{k,pro}$. This attitude is expressed by:

$$D_{k,pro}(n) = sign(X_k(n_1) - X_k(n_2)) \quad (13)$$

where $n_1, n_2 \in \{n, n-1, n-2\}$ and it is assumed that $X_k(n_1)$ is better than $X_k(n_2)$. Final direction is obtained based on a compromise among the aforesaid four empirical directions, namely, $D_{k,ego}(n)$, $D_{k,alt1}(n)$, $D_{k,alt2}(n)$ and $D_{k,pro}(n)$. In this study, j^{th} is dimension of $D_k(n)$ and is calculated as follows:

$$d_{kj} = \begin{cases} 0 & \text{if} \quad r_j \le p_j^{(0)} \\ +1 & \text{if} \quad p_j^{(0)} < r_j \le p_j^{(0)} + p_j^{(+1)} \\ -1 & \text{if} \quad p_j^{(0)} + p_j^{(+1)} < r_j \le 1 \end{cases} \quad (14)$$

where r_j is a uniform random number between 0 and 1, $p_j^{(0)}$ is the percent of zero number from the set of $\{D_{k,ego}, D_{k,alt1}, D_{k,alt2}, D_{k,pro}\}$, $p_j^{(+1)}$ is the percent of 1 and $p_j^{(-1)}$ is the percent of -1.

C. Step Length

Based on human searching behavior, it is comprehended that one may find near-optimal solutions in a narrower neighborhood of the point with lower fitness value and in wider neighborhood of the point with higher fitness value. The understanding and linguistic description of human searching makes fuzzy system a good candidate for simulating human focusing searching behavior. The uncertainty reasoning of human searching could be described by natural linguistic variables and a simple control rule as a fuzzy system operates on the principle of control rule as ‘‘If { *fitness value is small* (the conditional part)}, then { *step length is short* (the action part)} [32].

A fuzzy system must be applicable for wide range of optimization problem. For this aim, the fitness values of all the seekers are sorted in descending order (for minimization problems). The sorted fitness is turned into the sequence number from 1 to *S* where *S* is equal to number of population. This sequence is the input of Fuzzy reasoning. The linear membership function is used in the conditional part since the universe of discourse is a given set of numbers, i.e., 1, 2,…,*S*.

The expression is expressed as follows:

$$\mu_k = \mu_{\max} - \frac{S - I_k}{S-1}(\mu_{\max} - \mu_{\min}) \tag{15}$$

Where I_k is the sequence number of seeker k^{th}, $\mu_{\max}$ is the maximum amount of membership function and $\mu_{\min}$ is the minimum amount. $\mu_{\max}$, is a real number near one. In this study it is assumed to be 0.95.

The function $\mu(x) = e^{-x^2/2\delta^2}$ (Bell membership function) is used in the literature to introduce the action part (Fig. 1). For more simplicity, one variable is considered. Thus, the membership degree values of the input variables out of the range $[-3\delta, 3\delta]$ are less than 0.0111 $(\mu(\pm 3\delta) = 0.0111)$, and the elements beyond $[-3\delta, 3\delta]$ can be neglected [32]. Therefore, the value of $\mu_{\min}$ is set on 0.0111. The parameter δ of the Bell membership function is also calculated by (16):

$$\delta = \omega \cdot abs(X_{best} - X_{rand}) \tag{16}$$

where *abs* is the absolute symbol. X_{best} and X_{rand} are the best seeker and randomly selected seeker of the subpopulation, respectively and they must be different. The parameter δ is shared by all the seekers in the same subpopulation. The parameter ω is utilized to decrease the step length. Its amount decreases from $\omega_{\max}$ to $\omega_{\min}$ by varying the iteration number as follows:

$$\omega = \omega_{\max} + (\omega_{\max} - \omega_{\min}) \cdot \text{iter}/\text{iter}_{\max} \tag{17}$$

The values of 0.9 and 0.1 have been suggested in the [27] for $\omega_{\max}$ and $\omega_{\min}$, respectively.

In order to extend the randomness in each variable and to improve local search capability μ_i is converted to vector μ_{kj} by the following equation:

$$\mu_{kj} = Rand(\mu_k, 1) \tag{18}$$

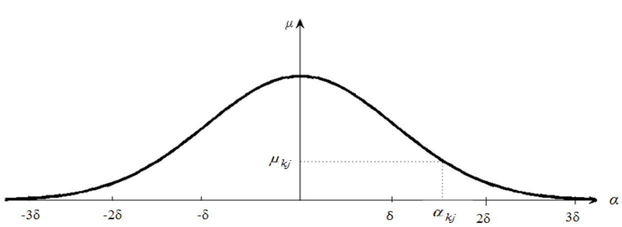

Fig. 1. The action part of the Fuzzy reasoning

where $Rand(\mu_k, 1)$ is a random number between μ_k and 1. Then, the *action* part of the Fuzzy reasoning gives every dimension *j* of *step length* as following relation:

$$\alpha_{kj} = \delta_j \sqrt{-Ln(Rand(\mu_k, 1))} \tag{19}$$

where *Ln* is natural logarithm.

D. Subpopulations Learning From Each Other

Each subpopulation searches for optimum solution by their information. Therefore, it is possible that the algorithm gets stuck in the local solutions. Subpopulations must learn from each other about the optimum information to avoid from premature convergences. Thus, the position of the worst seeker of each subpopulation is combined with the best one in each of the other subpopulations using the following binomial crossover operator as is expressed:

$$x_{k_n j} = \begin{cases} x_{lj,best} & \text{if } rand \le 0.5 \\ x_{k_n j,worst} & \text{elsewhere} \end{cases} \tag{20}$$

where *rand* is a random number within [0,1], $x_{lj,best}$ is the j^{th} dimension of the best seeker of subpopulation *l*, $x_{lj,best}$ is the j^{th} variable of the n^{th} worst position in the k^{th} subpopulation. Flowchart of seeker optimization algorithm has been showed in Fig. 2.

IV. Unit Commitment Solving Based on SOA

Unit commitment is a complex problem because of its nonlinearity. It has also many constraints which complicate the problem more and more. In this paper, SOA has been used for UC problem. This algorithm is very efficient for nonlinear problems. A new heuristic method based on combination of binary and integer coding has also been proposed to accommodate with UC constraints. Number of ON/OFF cycles of units is equal to planning duration (here is considered to be 24) because, by limiting the number of cycle to 5 (which is common in the pervious integer works), it is very difficult for optimization algorithm to commit or de-commit the units. Because it may increase the ON/OFF cycle more than 5 while considering the number of cycles equal to 24, doesn't prevent from committing or de-committing the units. Moreover, considering 5 cycles, ramp rate constraints are satisfied hardly and taking into account the penalty factor, only with 24 cycles in order to code the problem ramp rate, constraints are satisfied without using any penalty coefficients. According to this combination method, initializing the primary population, updating the new solutions and satisfying the minimum up/down time constraints have been solved by the integer coding. The load balancing and the spinning reserve constraints have also been satisfied by the binary coding. Fig. 3 shows flowchart of the proposed method.

A. Initial Population

Initial population has been produced by the integer coding. Active (ON) hours have been showed with positive integer codes and OFF hours by negative integer codes. Zeros also show that the period of planning has been finished. Positive and negative generated codes are produced so that absolute sum of them is equal to 24. A sample string in the initial population has been showed in Fig. 4. Superiority of this method on the others is that many of infeasible solutions (related to the minimum up/down time constraints) are eliminated in population generating. Anyhow, the spinning reserve constraint isn't satisfied by the initial population and therefore, penalty factor is used for it in the dispatch and evaluation stage. According to mentioned contents, population is initialized as below: First, cycle is expressed by [8]:

$$T_i^1 = \begin{cases} rand((MUT_i - T_i^0), T) & \text{if } T_i^0 > 0 \\ -rand((MDT_i - T_i^0), T) & \text{if } T_i^0 < 0 \end{cases} \quad (21)$$

where T_i^0 is the initial state of unit *i*. Other cycles are generated as below:

if $T_i^{c-1} < 0$, next cycle will change as follows:

$$T_i^c = \begin{cases} rand(MUT_i, RT_i^{c-1}) & RT_i^{c-1} > MUT_i \\ RT_i^{c-1} & \text{otherwise} \end{cases} \quad (22)$$

and if $T_i^{c-1} > 0$, next cycle will change as follows:

$$T_i^c = \begin{cases} -rand(MDT_i, RT_i^{c-1}) & RT_i^{c-1} > MDT_i \\ -RT_i^{c-1} & \text{otherwise} \end{cases} \quad (23)$$

where RT_i^{c-1} obtains by:

$$RT_i^{c-1} = T - \sum_{h=1}^{c-1} \left| T_i^h \right| \quad (24)$$

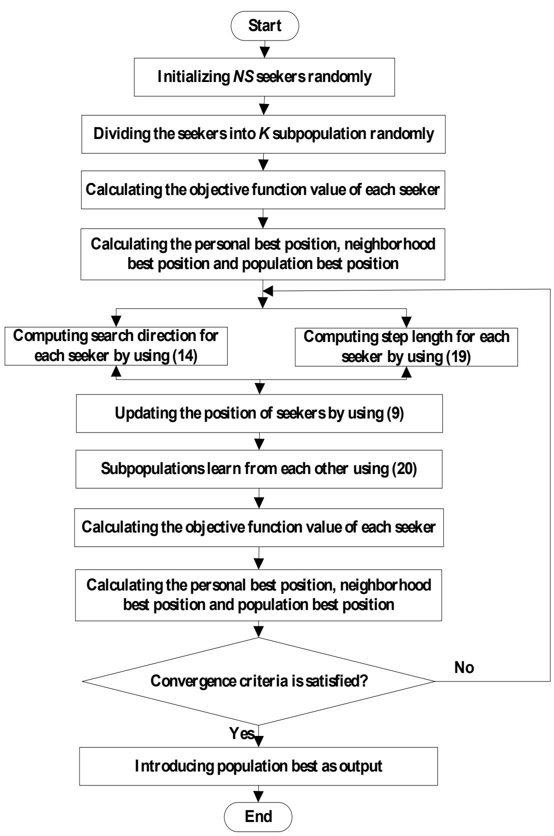

Fig. 2. Flowchart of seeker optimization algorithm

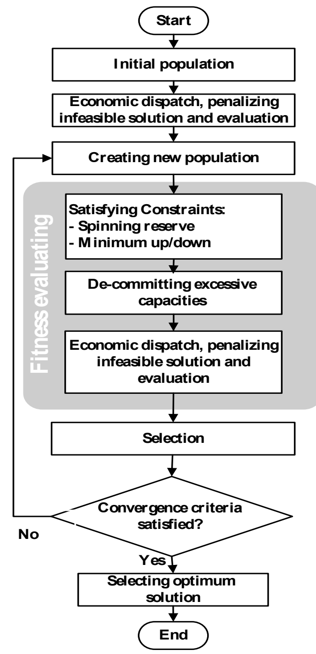

Fig. 3. Flowchart of the proposed method

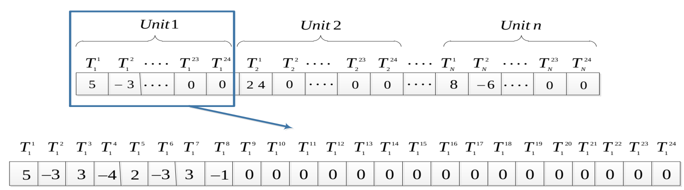

Fig. 4. A sample string in the initial population

B. Creating New Population

New populations are produced by SOA algorithm. As mentioned above, absolute sum of cycles of any solution must be equal with duration of planning. Anyhow, this constraint may not be satisfied by new solutions. So, for values more or less than 24, it must be divided between cycles proportionally.

The *Ln* function is also used in the step length has continuous amounts, so the step length will be continuous. Therefore new population isn't integer while solution of UC problem must be only represented by integer numbers. To solve this problem, round function has been utilized. After rounding, maybe sum of ON/OFF cycles isn't equal to 24. This difference has affected on the last non zero cycle.

C. Priority List

Units are ranked based on priority list. Priority list is based on parameters of units. To create priority list, it is assumed that the units operate with maximum power generation and then, their increasing rate are calculated. At last, priority list is expressed as follows [36]:

$$\alpha_i = 2a_i \cdot P_{i\,\max} + b_i \quad (25)$$

where α_i is the priority of unit *i* for committing. Low α_i means high priority for committing.

D. Spinning Reserve Satisfaction

There are 24 binary digits for each unit during a day in binary solving of UC problem. In this solving pattern digit 0 and 1 show OFF and ON state, respectively. In the proposed method, if spinning reserve constraint is violated, deactivated units will be committed according to priority list based on cheaper units. This process continues until the spinning reserve constraint is satisfied. By using this method to satisfy the spinning reserve, expensive units are replaced with cheaper units. Fig. 5 shows flowchart of proposed heuristic method to handle the spinning reserve constraint.

For example, a sample spinning reserve handling for a system with 5 units along 8 hours (Table I) has been depicted step by step as follows:

a) Start at first hour and consider on units.

Unit 1 is only on in the first hour. Therefore, the system can only produce 500 MW, while the load demand is 800 MW in this hour.

b) Turn on next unit in respect to priority list. In this situation, system can generate 1000 MW that is more than the load demand at this hour. Moreover, the spinning reserve has been met in the first hour, and units on/off schematic are changed as Table II. Therefore, go to the next hour.

c) Repeat step *a* and *b* for all 8 hours.

TABLE I
SAMPLE EXAMPLE FOR CONSIDERING RESERVE CONSTRAINT

P_{max}		Units priority	Hour							
			1	2	3	4	5	6	7	8
500		1	1	1	1	1	1	1	1	1
500		2	0	1	1	1	1	1	1	1
300		3	0	0	1	0	1	1	0	0
150		4	0	0	0	0	1	0	0	0
100		5	0	0	0	0	0	0	0	1
		Load	8	900	1000	1100	1200	1450	1200	900

TABLE II
SPINNING RESERVE HANDLING IN FIRST HOUR

Unit	Hour							
	1	2	3	4	5	6	7	8
1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1
3	0	0	1	0	1	1	0	0
4	0	0	0	0	1	0	0	0
5	0	0	0	0	0	0	0	1

E. Satisfying Minimum Up/Down Constraints

After satisfying the spinning reserve constraint using new solutions, the minimum up/down constraints must be checked for all cycles. If these constraints are violated, they must be repaired. Each unit has a feasible and an infeasible region and the proposed method converts the infeasible region to feasible one (Fig. 6). This technique is expressed as follows:

For satisfying minimum up time, if $T_i^c < MUT_i$ then, the amount of unit minimum up time is added to T_i^c as follows:

$$T_i^c = T_i^c + MUT_i \tag{26}$$

And similarly, in order to satisfy minimum down time, if $-T_i^c < MDT_i$ then, the amount of minimum down time is subtracted from T_i^c as follows:

$$T_i^c = T_i^c - MDT_i \tag{27}$$

Meanwhile, if the remained time was lower than the minimum up/down time in non-zero last cycle, then:

$$T_i^c = RT_i^{c-1} \tag{28}$$

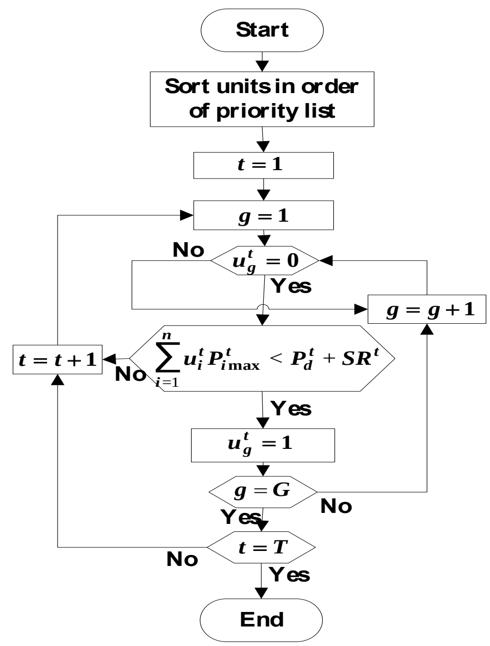

Fig. 5. Flowchart of handling the spinning reserve constraint

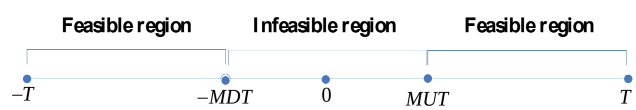

Fig. 6. Positions of feasible and infeasible regions for minimum up/down constraints

F. De- committing of Excessive Capacity

The unit de-commitment has been done after satisfying minimum up/down constraint. As mentioned previously, spinning reserve constraint is satisfied by committing new units. Therefore it may create excessive capacity in the system. Excessive capacity is de-committed from system according to priority list. This process continues until active units can be able to supply the load and spinning reserve and the minimum up/down time constraint doesn't be also violated. Fig.7 illustrates flowchart of the mentioned heuristic process.

Assume previous 5 units system by ON/OFF schematic as Table III.

a) Start from the first hour. Consider on units. It is visible that four units are on and therefore they can generate 1450 MW. The load demand is 800 MW at this hour. Therefore, some units can be turned off. Consider the most expensive on unit in respect to priority list (unit 4). This unit had been activated 4 hours (3 hours in initial condition plus an hour in new cycle), while MUT is 2, and the next hour is also off. Therefore unit 4 will be turned off.

b) Consider next expensive unit (unit 3). This unit had been activated 3 hours and MUT is 3. Therefore, there isn't any problem from MUT point of view. But this unit couldn't be off because of the next hour. If this unit is off then, MDT constraint will be violated. Similarly, units 1 and 2 could not be turned off. Therefore, the schematic will be changed as Table IV.

TABLE III
SAMPLE EXAMPLE FOR CONSIDERING DE-COMMITTING OF EXCESSIVE CAPACITY

T_0	MUT/ MDT	P_{max}		Units priority	Hour 1	2	3	4	5	6	7	8
5	5	500		1	1	1	1	1	1	1	1	1
5	5	500		2	1	1	1	1	1	1	1	1
3	3	300		3	1	1	1	1	1	1	0	0
3	2	150		4	1	0	0	1	1	1	1	0
-1	1	100		5	0	0	1	1	1	0	1	1
				Load	800	900	1000	1100	1200	1450	1200	900

TABLE IV
DE-COMMITTING EXCESSIVE CAPACITY IN FIRST HOUR

Units priority	Hour 1	2	3	4	5	6	7	8
1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	0	0
4	0	0	0	1	1	1	1	0
5	0	0	1	1	1	0	1	1
Load	800	900	1000	1100	1200	1450	1200	900

c) Repeat step a and b for all hours.

G. Economic Load Dispatch and Evaluation

Economic load dispatch is done by the classic lambda iteration method. Producing power of units at each hour is determined in this step. Obtained powers specify production cost by equation (2).

It is also possible that the load balancing and the spinning reserve constraint would be violated again after satisfaction of the minimum up/down constraints. Penalty factor is used for compensation of violated spinning reserve as below [20]:

$$PX = pf * \sum_{t=1}^{T} \frac{1}{P_d^{\,t}} \cdot ((P_d^{\,t} + SR^t) - \sum_{i=1}^{n} u_i^t . P_{i\ max}^t) \qquad (29)$$

where PX is penalty amount and pf is constant penalty factor. Therefore, the fitness function will be as follows:

$$fitness = FC_T + SU_T + PX \qquad (30)$$

where $fitness$, FC_T and SU_T are respectively the values of fitness function, total fuel cost and total start up cost.

H. Satisfying Ramp Rate Constraints

Commitment of required units, de-commitment of excessive units and economic dispatch are done by considering the nominal maximum and minimum output power of units. Anyhow, the ramp rate constraints affect the maximum and minimum available power of units.

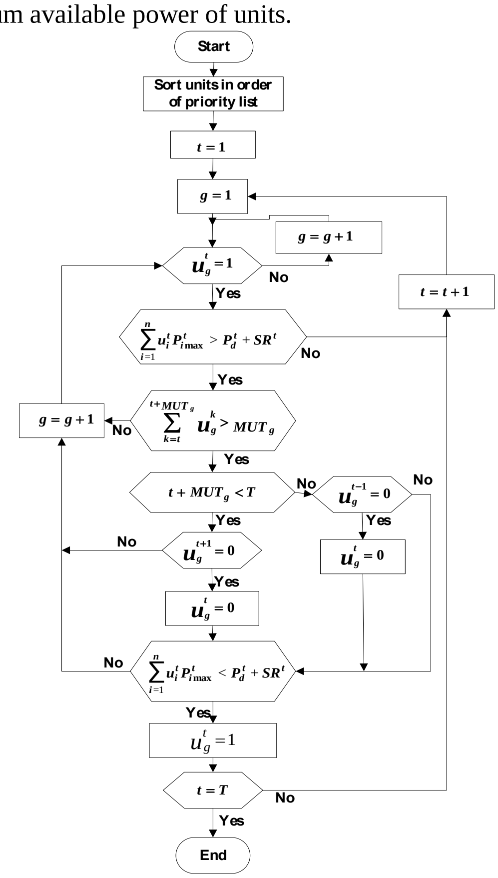

Fig. 7. Flowchart of de-committing the excessive capacity

Therefore, before satisfaction of spinning reserve constraint and de-commitment of excessive capacity, new maximum and minimum output power according to the ramp rate constraints must be determined at each planning hour. Execution of the economic dispatch at each hour determines these bounds.

V. NUMERICAL RESULT

A. Without Considering Ramp Rate Constraint

The proposed method has been tested on the UC problem for the realistic power systems of different sizes which consists of 10, 20, 40, 60, 80 and 100 units over a scheduling period of 24 hours. The load data and details of ten unit systems are given in [9]. For the 20 units, the data of ten unit systems have been duplicated and the load data is also doubled. To construct the other test systems, the same procedure has been applied. The spinning reserve amount has been considered to be 10% of hourly demand. The results of 20 runs of the proposed method with different population have been given in Table V. These results show the best, average, worst and standard deviation of solutions (with respect to $) and also one run time duration (with respect to second). Fig. 8 shows five sequential runs of ten unit systems with different initial population so that all of these runs have been converged at last to optimum cost. Aimed at further comparison, the problem is solved by means of PSO algorithm in similar conditions with SOA, and its convergence diagram is depicted in Fig. 9. All of the productions of units and start up cost in 24 hours are given in Table VI. In order to increase the validation of proposed method, the productions of units, the percent of spinning reserve in all hours and total start up cost have been illustrated in Table VII.

Table VIII lists the comparison results of SOA and the other methods. These results illustrate that SOA has a better convergence rather than other methods on all of the studied systems. The execution times of SOA and available execution times of the other methods are presented in this table. The proposed SOA is carried on an Intel Pentium *IV* 2-GHz CPU with 1 Mbytes of RAM. The execution times of ICGA are obtained from an Intel Pentium IV 1.6-GHz CPU and 512 Mbytes of RAM. ICA has also used a computer Intel Pentium *IV* 2-GHz CPU with 512 Mbytes of RAM. Comparisons show that SOA has achieved to optimum solution at acceptable time.

The methods of ICA and ICGA have considered 5 cycles to code the UC problem. Comparison results between these methods and proposed method (that is considered 24 cycles to code) is declared that better results have been concluded by proposed method in all cases. For example, for 100 generator systems the amount of 5610261$ has obtained by proposed method while ICA and ICGA can reach to 5617913 and 5630838 $, respectively. This matter expresses the efficiency of considering 24 cycles against 5 cycles.

TABLE V

RESULT OF 20 RUNS FOR DIFFERENT POPULATIONS

No. Seeker	Best Solution	Mean solution	Worst solution	std	Time
9	563937.64	564258.47	566879.5	921.5	11.5
21	563937.64	563937.64	563937.64	0.0	18
30	563937.64	563937.64	563937.64	0.0	23

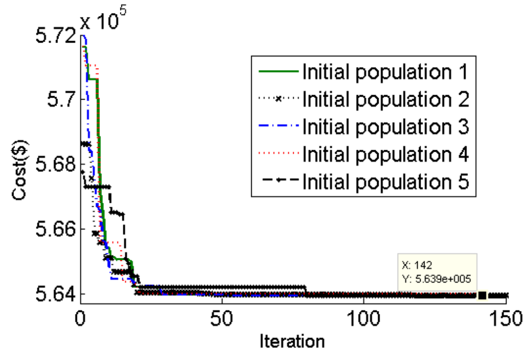

Fig. 8. Five sequential runs of ten units system

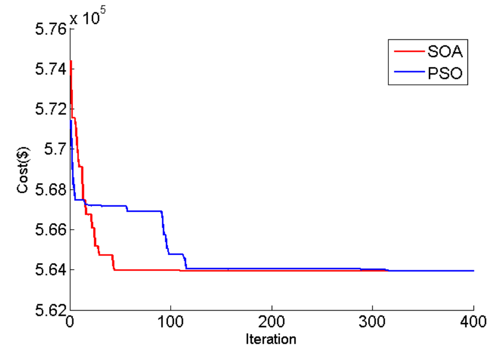

Fig. 9. SOA and PSO convergence in case study A

B. *With Considering Ramp Rate Constraint*

This test case has 26 units in time horizon of 24 hours. Test case data are given in [34] and Tables IX, X. Spinning reserve has been considered to be 5% of hourly demand. Ramp rate constraints have also been considered in it. Of course, start up and shut down costs have been neglected according to the original reference. In order to show efficiency of the proposed method by considering ramp rate limits, SOA has been run 20 times and results compared with PSO. Results show the superiority of SOA rather than PSO. In order to show the difference between considering ramp rate limit or not, SOA and PSO have also been run again for 20 times without ramp rate consideration. Fig. 10 and Table XI display this four mentioned states. SOA is converged to lesser cost in a few iterations in both states. In addition, evaluation indices include the best one, worst and average of solution and standard deviation is better than PSO.

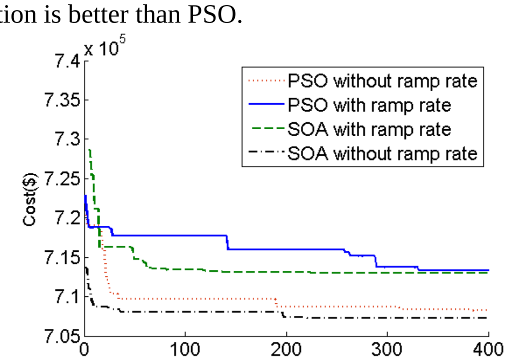

Fig. 10. SOA and PSO convergence in case study B

TABLE VI

CONFIGURATION OF ON/OFF CYCLE, OPERATION COST AND SCHEDULE FOR 24 h IN TEN UNITS SYSTEM

	Unit											
	1	2	3	4	5	6	7	8	9	10		
Duration of ON/OFF cycle	24	24	-5	-4	-2	-8	-8	-9	-10	-11		
			16	17	20	6	6	4	2	1		
			-3	-3	-2	-5	-5	-6	-12	-12		
						4	3	1				
						-1	-2	-4				
Time	Produced power (MW)										Start up cost ($)	Operation cost ($)
1	455	245	0	0	0	0	0	0	0	0	0	13683.12
2	455	295	0	0	0	0	0	0	0	0	0	14554.5
3	455	370	0	0	25	0	0	0	0	0	900	16809.45
4	455	455	0	0	40	0	0	0	0	0	0	18597.68
5	455	390	0	130	25	0	0	0	0	0	560	20020.02
6	455	360	130	130	25	0	0	0	0	0	1100	22387.04
7	455	410	130	130	25	0	0	0	0	0	0	23261.99
8	455	455	130	130	30	0	0	0	0	0	0	24150.32
9	455	455	130	130	85	20	25	0	0	0	860	27251.07
10	455	455	130	130	162	33	25	10	0	0	60	30057.54
11	455	455	130	130	162	73	25	10	10	0	60	31916.07
12	455	455	130	130	162	80	25	43	10	10	60	33890.15
13	455	455	130	130	162	33	25	10	0	0	0	30057.54
14	455	455	130	130	85	20	25	0	0	0	0	27251.07
15	455	455	130	130	30	0	0	0	0	0	0	24150.32
16	455	310	130	130	25	0	0	0	0	0	0	21513.64
17	455	260	130	130	25	0	0	0	0	0	0	20641.83
18	455	360	130	130	25	0	0	0	0	0	0	22387.04
19	455	455	130	130	30	0	0	0	0	0	0	24150.32
20	455	455	130	130	162	33	25	10	0	0	490	30057.54
21	455	455	130	130	85	20	25	0	0	0	0	27251.07
22	455	455	0	0	145	20	25	0	0	0	0	22735.54
23	455	425	0	0	0	20	0	0	0	0	0	17645.36
24	455	345	0	0	0	0	0	0	0	0	0	15427.43
	Total cost : 563937.64										4090	559847.6

TABLE VII

COMMITMENT SCHEDULING, OUTPUT OF GENERATORS AND PERCENT OF SPINNING RESERVE AT EACH HOUR FOR THE 100-UNIT SYSTEM OBTAINED BY SOA

Units	Hours																						
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
2	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
3	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
4	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
5	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
6	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
7	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
8	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
9	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
10	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455	455
11	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
12	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
13	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
14	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
15	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
16	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
17	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
18	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	350
19	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	0
20	245	295	390	455	455	455	455	455	455	455	455	455	455	455	455	375	325	406	455	455	455	455	377	0
21	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
22	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
23	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
24	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
25	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
26	0	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
27	0	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
28	0	0	0	0	0	0	0	130	130	130	130	130	130	130	0	0	0	0	0	130	130	130	130	130
29	0	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
30	0	0	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
31	0	0	0	0	130	130	130	130	130	130	130	130	130	130	0	0	0	0	0	130	130	130	130	130
32	0	0	0	0	130	130	130	130	130	130	130	130	130	130	0	0	0	0	0	130	130	130	130	130
33	0	0	0	0	0	130	130	130	130	130	130	130	130	130	0	0	0	0	0	130	130	130	130	130
34	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0
35	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0
36	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0
37	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
38	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0	0
39	0	0	0	0	0	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	130	0	0
40	0	0	0	0	0	130	130	130	130	130	130	130	130	130	0	0	0	0	0	130	130	130	130	130
41	0	0	0	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	25	0
42	0	0	0	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
43	0	0	25	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
44	0	0	25	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
45	0	0	0	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
46	0	0	0	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
47	0	0	0	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
48	0	0	0	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
49	0	0	0	0	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
50	0	0	0	44.4	51	47	45	30	104	162	162	162	162	104	81	25	25	25	57	162	104	66	0	0
51	0	0	0	0	0	0	0	0	20	35	75	80	35	20	0	0	0	0	20	34	20	0	0	0
52	0	0	0	0	0	0	0	0	20	35	75	80	35	20	0	0	0	0	20	34	20	0	0	0
53	0	0	0	0	0	0	0	0	20	35	75	80	35	20	0	0	0	0	20	34	20	0	0	0
54	0	0	0	0	0	0	0	0	0	35	75	80	35	20	0	0	0	0	20	34	20	0	0	0
55	0	0	0	0	0	0	0	0	20	35	75	80	35	20	0	0	0	0	20	34	20	0	0	0
56	0	0	0	0	0	0	0	0	20	35	75	80	35	20	0	0	0	20	20	34	0	0	0	0
57	0	0	0	0	0	0	0	0	20	35	75	80	35	0	0	0	0	20	20	34	0	0	0	0
58	0	0	0	0	0	0	0	0	0	35	75	80	35	20	20	0	0	0	0	34	20	20	0	0
59	0	0	0	0	0	0	0	0	0	35	75	80	35	20	20	0	0	0	20	34	20	0	0	0
60	0	0	0	0	0	0	0	0	20	35	75	80	35	20	20	0	0	0	20	34	20	0	0	0
61	0	0	0	0	0	0	0	0	0	25	25	25	25	25	25	0	0	0	0	25	25	25	0	0
62	0	0	0	0	0	0	0	0	25	25	25	25	25	25	25	0	0	0	0	25	25	25	0	0
63	0	0	0	0	0	0	0	0	25	25	25	25	25	0	0	0	0	25	25	25	0	0	0	0
64	0	0	0	0	0	0	0	0	0	25	25	25	25	0	0	0	0	25	25	25	0	0	0	0
65	0	0	0	0	0	0	0	0	25	25	25	25	25	0	0	0	0	25	25	25	0	0	0	0
66	0	0	0	0	0	0	0	0	0	25	25	25	25	25	25	0	0	0	25	25	25	0	0	0
67	0	0	0	0	0	0	0	0	25	25	25	25	25	0	0	0	0	25	25	25	0	0	0	0
68	0	0	0	0	0	0	0	0	0	25	25	25	25	0	0	0	0	25	25	25	0	0	0	0
69	0	0	0	0	0	0	0	0	0	25	25	25	25	0	0	0	0	0	25	25	25	0	0	0
70	0	0	0	0	0	0	0	0	25	25	25	25	25	0	0	0	0	25	25	25	0	0	0	0
71	0	0	0	0	0	0	0	0	0	10	10	45	10	0	0	0	0	0	0	10	0	0	0	0
72	0	0	0	0	0	0	0	0	0	10	10	45	10	0	0	0	0	0	0	10	0	0	0	0
73	0	0	0	0	0	0	0	0	0	10	10	45	10	0	0	0	0	0	0	10	0	0	0	0
74	0	0	0	0	0	0	0	0	0	10	10	45	10	0	0	0	0	0	0	10	0	0	0	0
75	0	0	0	0	0	0	0	0	0	10	10	45	10	10	0	0	0	0	0	10	0	0	0	0
76	0	0	0	0	0	0	0	0	0	0	10	45	10	0	0	0	0	0	0	10	0	0	0	0
77	0	0	0	0	0	0	0	0	0	0	10	45	10	0	0	0	0	0	0	10	0	0	0	0
78	0	0	0	0	0	0	0	0	0	10	10	45	10	0	0	0	0	0	0	10	0	0	0	0
79	0	0	0	0	0	0	0	0	0	10	10	45	0	0	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0	0	10	10	45	0	0	0	0	0	0	0	0	0	0	0	0
81	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	0	0
82	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	0	0
83	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	0	0
84	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	0	0
85	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	10	0	0	0	0
86	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	0	0
87	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	0	0
88	0	0	0	0	0	0	0	0	0	0	10	10	0	0	0	0	0	0	0	0	0	0	0	0
89	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	0	0	0	0	10	0	0	10	0	0	0	0	0	0	0	0	0
91	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
92	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
93	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
94	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
95	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
96	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
97	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
98	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0
99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Reserve (%)	30	21.33	10	8.1	7.9	7.8	8.8	11	10	10.1	10.1	10.1	10.1	10.38	9.9	20.4	26.38	21	17	10.5	10	7.7	10.1	10.9

Start up cost: 46300 $

Total operation cost: 5610261

TABLE VIII
COMPARISON OF RESULTS FOR CASE STUDY A

Method	Best Cost ($)	Average Cost ($)	Worst Cost ($)	Time (s)	Method	Best Cost ($)	Average Cost ($)	Worst Cost ($)	Time (s)
10 Units					20 Units				
EP [13]	564551	565532	566231	100	EP	1125494	1127257	1129793	340
GA [7]	565825	---	570032	221	GA	1126243	---	1132059	733
LR [7]	566107	---	---	257	LR	1128362	---	---	514
ICGA [8]	566404	---	---	7.4	ICGA	1127244	---	---	22.4
SA [18]	565828	565988	566260	3	SA	1126251	127955	1129112	17
ICA [20]	563938	564406	---	48	ICA	1124274	---	---	63
HS [22]	565828	---	---	---	HS	1127377	---	---	92
PUC-MP[23]	563938	564040	564320	---	PUC-MP	1124490	1124803	1124995	---
SOA	**563938**	**563938**	**563938**	**18**	**SOA**	**1124274**	**1124429**	**1124840**	**115**
40 Units					60 Units				
EP	249093	2252612	2256085	1176	EP	3371611	3376255	3381012	2267
GA	2251911	---	2259706	2697	GA	3376625	---	3384252	5840
LR	2250223	---	---	1066	LR	3374994	---	---	1594
ICGA	2254123	---	---	58.3	ICGA	3378108	---	---	117.3
SA	2250063	2252125	2254539	88	SA	---	---	---	---
ICA	2247078	---	---	151	ICA	3371722	---	---	366
HS	2250968	---	---	467	HS	3375138	---	---	1021
SOA	**2245862**	**2246836**	**2247715**	**233**	**SOA**	**3366188**	**3367850**	**3368859**	**431**
80 Units					100 Units				
EP	4498479	4505536	4512739	3584	EP	5623885	5633800	5639148	6120
GA	4504933	---	4510129	10036	GA	5627437	---	5637914	15733
LR	4496729	---	---	2122	LR	5620305	---	---	2978
ICGA	4498943	---	---	176	ICGA	5630838	---	---	242
SA	4498076	4501156	4503987	405	SA	5617876	5624301	5628506	316
ICA	4497919	---	---	994	ICA	5617913	---	---	1376
HS	4500745	---	---	2157	HS	5622350	---	---	3710
SOA	**4489137**	**4490488**	**4492070**	**709**	**SOA**	**5610261**	**5612789**	**5615256**	**1014**

TABLE IX
GENERATING UNIT DATA FOR 26 UNIT SYSTEMS

Unit	*Pmin (MW)*	*Pmax (MW)*	*a*	*b*	*c*	*MUT*	*MDT*	*Initial State*	*UR*	*DR*
Unit 1	2.4	12	0.02533	25.5472	24.3891	0	0	-1	48	60
Unit 2	2.4	12	0.02649	25.6753	24.411	0	0	-1	48	60
Unit 3	2.4	12	0.02801	25.8027	24.6382	0	0	-1	48	60
Unit 4	2.4	12	0.02842	25.9318	24.7605	0	0	-1	48	60
Unit 5	2.4	12	0.02855	26.0611	24.8882	0	0	-1	48	60
Unit 6	4	20	0.01199	37.551	117.7551	0	0	-1	30.5	70
Unit 7	4	20	0.01261	37.6637	118.1083	0	0	-1	30.5	70
Unit 8	4	20	0.01359	37.777	118.4576	0	0	-1	30.5	70
Unit 9	4	20	0.01433	37.8896	118.8206	0	0	-1	30.5	70
Unit 10	15.2	76	0.00876	13.3272	81.1364	3	2	3	38.5	80
Unit 11	15.2	76	0.00895	13.3588	81.298	3	2	3	38.5	80
Unit 12	15.2	76	0.0091	13.38085	81.4641	3	2	3	38.5	80
Unit 13	15.2	76	0.00932	13.4073	81.6259	3	2	3	38.5	80
Unit 14	25	100	0.00623	18	217.8952	4	2	-3	51	74
Unit 15	25	100	0.00612	18.1	218.335	4	2	-3	51	74
Unit 16	25	100	0.00598	18.2	218.7752	4	2	-3	51	74
Unit 17	54.25	155	0.00463	10.694	142.7348	5	3	5	55	78
Unit 18	54.25	155	0.00473	10.7154	143.0288	5	3	5	55	78
Unit 19	54.25	155	0.00481	10.7367	143.3179	5	3	5	55	78
Unit 20	54.25	155	0.00487	10.7583	143.5972	5	3	5	55	78
Unit 21	68.95	197	0.00259	23	259.131	5	4	-4	55	99
Unit 22	68.95	197	0.0026	23.1	259.649	5	4	-4	55	99
Unit 23	68.95	197	0.00263	23.2	260.176	5	4	-4	55	99
Unit 24	140	350	0.00153	10.8616	177.0575	8	5	10	70	120
Unit 25	100	400	0.00194	7.4921	310.0021	8	5	10	50.5	100
Unit 26	100	400	0.00195	7.5031	311.9102	8	5	10	50.5	100

TABLE X
LOAD DATA FOR 26 UNIT SYSTEMS

Hour	1	2	3	4	5	6	7	8	9	10	11	12
Load	1700	1730	1690	1700	1750	1850	2000	2430	2540	2600	2670	2590
Hour	13	14	15	16	17	18	19	20	21	22	23	24
Load	2590	2550	2620	2650	2550	2530	2500	2550	2600	2480	2200	1840

TABLE XI
COMPARISON OF RESULTS FOR CASE STUDY B

	Method	Best Solution	Mean solution	Worst solution	Time (s)
Without ramp rate	PSO	708220.2	710106.7	713504.2	346
	SOA	707155.3	707433.7	708019.8	337
With ramp rate	PSO	713356.4	714859.3	717677	377
	SOA	712836.4	713140.1	713643.8	350

C. Sensitivity Analysis

As mentioned above, the parameter ω is utilized to decrease the step length. Its amount is decreases from $\omega_{\max}$ to $\omega_{\min}$ by varying the iteration. In order to comprehend the correct amount of $\omega_{\max}$ and $\omega_{\min}$, sensitivity analysis has been conducted on these parameters.

Simulations have been done on a sample 20 unit systems to show the sensitivity of SOA with respect to the minimum and maximum value of ω. Table XII lists the results of simulation over 20 run of the program. The results show that the better results are obtained in high amount of $\omega_{\max}$ and the worse results are concluded in low amount of $\omega_{\max}$. For example, the amount of 0.25 and 0.1 for $\omega_{\max}$ and $\omega_{\min}$ are concluded to worst result and the value of 0.9 and 0.5 give the best result.

TABLE XII
RESULTS OF SENSITIVITY ANALYSIS ON INITIAL AND FINAL VALUES OF ACCELERATION COEFFICIENTS OF SOA IN 20 UNIT SYSTEM

$\omega_{\max}$	$\omega_{\min}$	Best Result	Average Result	Std
0.9	0.1	1124480	1124651	151
0.75	0.1	1124305	1124597	166
0.5	0.1	1124480	1124761	144
0.25	0.1	1124539	1125403	919
0.9	0.25	1124480	1124584	125
0.9	0.5	1124274	1124429	166
0.9	0.75	1124274	1124439	176
0.75	0.5	1124432	1124722	210
0.75	0.25	1124539	1124769	179
0.5	0.25	1124539	1124801	153

VI. CONCLUSION

This paper has proposed a new heuristic method to solve UC problem using a seeker optimization algorithm. In the heuristic proposed method two integer and binary coding have been used that caused better solution than other popular methods. Switching between integers and binary coding and satisfying related constraints are very simpler than conventional binary or integer coding. Integer coding satisfies the minimum up/down constraints without using the custom penalty factor. Binary coding and penalty coefficients also satisfy the spinning reserve constraint well. The proposed method has been applied to UC problem with several test systems consisting up to 100 units and a special 26 unit systems having the ramp rate constraint. The simulation results show better performance of the proposed method than the others clearly. A case with ramp rate constraint was also solved in which the efficiency of the proposed SOA is revealed from convergence point of view in compare of the standard PSO. Simulation results show the capability of SOA to solve complex and nonlinear problems. Therefore, SOA has very potential to apply on large and complex problems.

VI. REFERENCES

[1] Younesian, M. S. , Samiei. M, Moghaddam. M, Vahedi. M, Davarzani. R., ‘Security-constrained unit commitment in the presence of smart devices using a two-stage robust optimization model.’ Energy Reports, 2025, 13, pp. 5427-5445.

[2] Jiang, S., Gao, S., Pan, G., Zhao, X., Liu, Y., Guo, Y., "A novel robust security constrained unit commitment model considering HVDC regulation." Applied Energy, 2020, 278, pp. 115652.

[3] Yamin, H. Y., 'Review on methods of generation scheduling in electric power systems,' Elec. Power Syst. Res., 2004, 69, (2-3), pp.227-248.

[4] Kuhn, H. W. and Tucker, A. W., 'Nonlinear programming,' Proceedings of the Second Berkeley Symposium on Mathematical Programming Statistics and Probability, University of California Press, Berkeley, Jul./Aug. 1951.

[5] Pang, C. K., Sheble, G. B., and F. Albuyeh, 'Evaluation of dynamic programming based methods and multiple area representation for thermal unit commitments,' IEEE Trans. Power Syst., 1981, 100, (3), pp. 1212-1218

[6] Geoffrion, A. M., 'Lagrangian relaxation for integer programming problems,' Math. Programming Study, 1974, 2, pp. 82-114.

[7] Cohen, A. I., and Yoshimura, M. , 'A branch-and-bound algorithm for unit commitment,' IEEE Trans. Power App. Syst., 1983, PAS-102 , pp. 444–451

[8] Kazarlis, S. A., Bakirtzis, A. G., and Petridis, V., 'A genetic algorithm solution to the unit commitment problem,' IEEE Trans. Power Syst., 1996, 11, (1), pp. 83–92.

[9] Damousis, I. G., Bakirtzis, A. G., and Dokopoulos, P. S., 'A Solution to the unit commitment problem using integer-coded genetic algorithm,' IEEE Trans. Power Syst., 2004, 19, (2), pp. 1165-1172.

[10] Ebrahimi, J., Hosseinian, S. H., and Gharehpetian, G. B., 'Unit commitment problem solution using shuffled frog leaping algorithm,' IEEE Trans. Power Syst., 2011, 26, (2), pp. 573-581

[11] Yuan, X., Nie, H., Su, A., Wanga, L., and Yuan, Y., 'An improved binary particle swarm optimization for unit commitment problem,' Expert Syst. App. 2009, 36, (4), pp. 8049–8055

[12] Yuan, X., Nie, H., Su, A., Wanga, L., and Yuan, Y., 'Application of enhanced discrete differential evolution approach to unit commitment problem,' Energy Convers. Manage., 2009, 50, (9), pp. 2449-2456.

[13] Chen, P., ' Two-level hierarchical approach to unit commitment using expert system and elite PSO'
IEEE Trans. Power Syst., 2012, 27, (2), pp. 780-789.

[14] Morales-España, G., Latorre, J., and Ramos A., 'Tight and compact

MILP formulation for the thermal unit commitment problem', IEEE Trans. Power Syst., 2013, 28, (4), pp. 4897 – 4908.

[15] Roy, P. K., 'Solution of unit commitment problem using gravitational search algorithm', Electric. Pow. Energ. Syst., 2013, 53, (1), pp. 85–94

[16] Garlík, B., and Kˇrivan, M., 'Renewable energy unit commitment, with different acceptance of balanced power solved by simulated annealing', Energ. Buildings, 2013, 67 (1), pp. 392–402.

[17] Najafi, A, Farshad, M, Falaghi, H. A new heuristic method to solve unit commitment by using a time-variant acceleration coefficients particle swarm optimization algorithm. Turkish Journal of Electrical Engineering and Computer Sciences, 2015, 23 (2), pp. 354-369.

[18] Juste, K. A., Kita, H., Tanaka, E., and Hasegawa, J., 'An evolutionary programming solution to the unit commitment problem,' IEEE Trans. Power Syst., 1999, 14, (4), pp. 1452–1459.

[19] Simopoulos, N. D., Kavatza, D. S., and Vournas, D., 'Unit commitment by an enhanced simulated annealing algorithm,' IEEE Trans. Power Syst., 2006, 21, (1), pp. 68-76.

[20] Eslamian, M., Hosseinian, S. H., and Vahidi, B., 'Bacterial foraging based solution to the unit-commitment problem,' IEEE Trans. Power Syst., 2009, 24, (3), pp. 1478–1488.

[21] Hadji, M. M, and Vahidi, B., 'A solution to the unit commitment problem using imperialistic competition algorithm,' IEEE Trans. Power Syst., 2012, 27, (1), pp. 117-124.

[22] Pavez-Lazo, B., and Soto-Cartes, J., 'A deterministic annular crossover genetic algorithm optimisation for the unit commitment problem,' Expert Syst. Appl., 2011, 38, (6), pp. 6523–6529.

[23] Najafi, S., pourjamal, Y, 'A new heuristic algorithm for unit commitment problem,' Energy Procedia, 2012, 14, pp. 2005-2011.

[24] Amiri, M., and Khanmohammadi, S., 'A primary unit commitment approach with a modification process,' Appl. Soft Comput., 2013, 13, (2), pp. 1007–1015.

[25] Mhanna, S.N, and Jabr, R.A., 'Application of semi definite programming relaxation and selective the unit commitment problem,' Electr. Pow. Syst. Res., 2012, 90, pp. 85–92.

[26] Lopez, J.A., Ceciliano-Meza, J. L., Moya, I. G., and Gomez, R. N., 'A MIQCP formulation to solve the unit commitment problem for large-scale power systems, Electric. Pow. Energy Syst., 2012, 36, (1), pp. 68–75.

[27] Rajan, C., Mohan, M., 'An evolutionary programming-based tabu search method for solving the unit commitment problem,' IEEE Trans. Power Syst. 2004, 19, (1), pp. 577–585.

[28] Su, C. C., and Hsu, Y. Y., 'Fuzzy dynamic programming: an application to unit commitment,' IEEE Trans. Power Syst., 1991, 6, (3), pp. 1231–1237.

[29] Zhou, D., Qu, Q., Guo F., "A dual-layer distributed dispatch framework for frequency-secure unit commitment in integrated Transmission–Distribution networks." Electric Power Systems Research, 2025, 248, pp. 111892.

[30] Gong, Y., Liu, T., Liu, P., Tong, X., "Joint unit commitment model for hydro and hydrogen power to adapt to large-scale photovoltaic power, " Energy Conversion and Management, 2024, 317, pp. 118794.

[31] Zhang, Y., Guo, Q., Zhou, Y., Sun, H., "Frequency-constrained unit commitment for power systems with high renewable energy penetration." International Journal of Electrical Power & Energy Systems 153, pp. 109274.

[32] Xu, J., Y. Ma, K. Li and Z. Li, "Unit commitment of power system with large-scale wind power considering multi time scale flexibility contribution of demand response." Energy Reports, 2021, 7, pp. 342-352.

[33] Wu, X., Fang, J., Chen. Z, "Distributionally robust unit commitment of integrated electricity and heat system under bi-directional variable mass flow." Applied Energy, 2022, 326, pp. 119788.

[34] Wang, C., and Shahidehpour, S. M., 'Effects of ramp rate limits on unit commitment and economic dispatch,' IEEE Trans. Power Syst., 1993, 8, (3), pp. 1341-1350.

[35] Shaw, B., Mukherjee, V., Ghoshal, S.P., 'Solution of economic dispatch problems by seeker optimization algorithm,' Expert Syst. Appl., 2012, 39, (1), pp. 508–519.

[36] Wood, A. J., and Wollenberg, B. F., Power generation operation and control. New York: Wiley, 1984.