Electrical Power and Energy Systems 55 (2014) 21–28

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Unit commitment problem solution using invasive weed optimization algorithm

B. Saravanan [a,*], E.R. Vasudevan [a], D.P. Kothari [b,1]

[a] Vellore Institute of Technology University, Vellore, India
[b] MVSR Engineering College, Hyderabad, India

ARTICLE INFO

Article history:
Received 20 March 2013
Received in revised form 29 July 2013
Accepted 23 August 2013

Keywords:
Unit commitment
Evolutionary programming
Economic dispatch
IWO
Seeds
Fitness

ABSTRACT

The evolutionary algorithm of invasive weed optimization algorithm popularly known as the IWO has been used in this paper, to solve the unit commitment (UC) problem. This integer coded algorithm is based on the colonizing behavior of weed plants and has been developed to minimize the total generation cost over a scheduled time period while adhering to several constraints such as generation limits, meeting load demand, spinning reserves and minimum up and down time. The minimum up/down time constraints have been coded in a direct manner without using the penalty function method. The proposed algorithm was tested and validated using 10 units and 24 h system. The most important merit of the proposed methodology is high accuracy and good convergence speed as it is a derivative free algorithm. The simulation results of the proposed algorithm have been compared with the results of other tested algorithms for UC such as shuffled frog leaping, particle swarm optimization, genetic algorithm and Lagrangian relaxation and bacterial foraging algorithm.

1. Introduction

Unit commitment (UC) is of high priority in power generation. UC may be defined as the determination of the units that need to be committed in order to satisfy load demand. In order to satisfy the load demand in a cost efficient manner it is important that unit commitment is performed. The continuous and increasing demand for power and the ever reducing fossil fuels make it impossible for continuous power supply to the people without any interruptions. Even though there is enough stress laid on the use of renewable sources of energy and a wide-scale implementation of different sources of energy like solar, wind, tidal, and biomass, can be seen it has not helped in putting a check on the power deficit that we face. The growing demand and the unreliability of renewable sources of energy forces the electricity generation to be extremely cost efficient. The optimal generation of power is necessary to meet the load demand and also to avoid any wastage of power [1]. The problem of unit commitment involves two processes; determining which units are meant to be ON/OFF; and to feed the unit commitment output into economic dispatch for determining the generation. The complications of the unit commitment problem increases with the increase in the system size, i.e. increase in the number of generating units. Certain problems like execution time and sub optimal solutions stresses the importance of developing new algorithms for unit commitment which can effectively overcome these problems. Moreover, the economic dispatch output must satisfy certain constraints pertaining to individual units or the system as a whole.

There are several methods to determine the unit commitment outputs but there are certain drawbacks of such methods. The conventional methods involve complete enumeration techniques and Priority List (PL) which might be easy to generate but take time for convergence. Priority list methods simply look out for cheapest units to switch on and make a priority list of the generating units in order to meet the load demand. Other evolutionary algorithms such as Genetic Algorithm (GA), Bacterial Foraging (BFOA), Lagrangian Relaxation (LR) [2] and Shuffled Frog Leaping Algorithms (SFLAs) [3] have their own drawbacks. All these evolutionary algorithms have been tested for 10 unit 24 h systems and are based on events happening in nature. Bacterial foraging algorithm solves the problem of UC based on the foraging techniques adopted by e-coli bacteria for convergence towards optimal solution. Genetic algorithm adopts the concept of combinations of DNA molecules, thereby forming all possible combinations of a unit commitment problem and deciding on the best combination. Frog Leaping algorithms adopt the mechanism of leaping of frogs towards food, wherein with each iterative step the converging variable moves towards the optimum point. Similarly Particle swarm optimization algorithm for UC problem is based on the animal flocking behavior. While PL method might be fast in execution speeds but it does not give a cost effective solution. The LR method is suitable for solving problems with large systems but it only

* Corresponding author.
E-mail addresses: bsaravanan@vit.ac.in (B. Saravanan), dpk0710@yahoo.com (D.P. Kothari).
[1] Director-Research

0142-0615/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ijepes.2013.08.020

Nomenclature

$F^{(i)}$	fuel cost objective function
$P(i)$	power output of the generation unit
P_d	load requirement
P_{max}	maximum amount of power unit can produce once it is turned on
P_{min}	minimum amount of power unit can produce once it is turned on
$P_{k,t}$	power produced by unit k at time period t
S_{cr}	cold state start-up cost
S_{hr}	hot state start-up cost
$S_{k,t}$	cost of starting up unit k at time t
$t_{coldstart}$	time a generator is in a hot state after it is turned off
t_{up}	minimum number of hours required for a generator to stay up once it is on
t_{down}	minimum number of hours required for a generator to stay down once it is off
σ	standard deviation
F_i	fitness of the ith plant
F_{worst}	fitness value of the worst plant
F_{best}	fitness value of the best plant
$iter_{max}$	maximum value of the iteration assigned by the user
$iter$	current iteration value
σ_0 and σ_f	initial and final values of standard deviations pre-assigned by the user
N_{max}	maximum number of seeds
N_{min}	minimum number of seeds
n	non-linear modulation index
UT_i	up time of unit i
DT_i	down time of unit i
SU_i	start-up cost of unit i
SD_i	shut down cost of unit i
PT_i	pending time in hourly schedule
$I_i(t)$	unit i ON/OFF status

generates sub optimal solutions due to lack of an iterative process in it. The evolutionary algorithms are stochastic search models. All evolutionary algorithm techniques are inspired by different processes occurring in nature. GA [4,5] is a binary coded algorithm which works on the binary values of 1 and 0. But its disadvantage is that it has a high execution time without any guarantee for an optimal solution. Although PSO [5] is more efficient than GA, it has a high dependency on initial conditions and parameter values. The SFLA approach although fast provides no guarantee that the obtained solution is optimal as it gives solutions with higher cost. Mixed Integer Linear Programming (MILP) has become giving more importance to solve UC problem because of many MILP solvers are introduced [18]. To reduce the search space and to increase the searching speed tight and compact MILP was developed and tested with 28–11,870 generators [18].

2. Invasive weed optimization algorithm

2.1. Terms used

I. Seeds – All units in the optimization problem that are assigned a value pertaining to the limiting conditions.
II. Plants – Seeds that grow into plants before being evaluated.
III. Fitness value – A value that determines how good the plant is, i.e. how much optimized the solution is.
IV. Field – The probable solution area/search area.

The technique of IWO was inspired from the biological growth of weed plants. It was first used by Mehrabian and Lucas in solving control system designing [6]. This technique is based on the colonizing behavior of invasive weed plants [7]. Weed plants are called invasive because the growth of weed plants is extensively invading in the growth area. IWO is known to be highly converging in nature since it a derivative free algorithm. It also converges to the optimal solution thereby eliminating any possibilities of sub optimal solutions. This integer coded algorithm also involves simple coding. IWO has been so far implemented for several applications such as DNA computing, antenna system design [8], optimal arrangement of piezoelectric actuators on smart structures.

In this algorithm, the number of decision variables are taken in the form of seeds and then randomly distributed in a definite search space [9]. These seeds are then allowed to grow into plants and the fitness of each individual plant is determined. Depending upon these fitness values, new seeds are generated by each plant in accordance with a normalized standard deviation σ. The importance of this σ is that it helps in converging to the optimal solution faster as it determines exactly where to distribute the new seeds so that the seeds always approach the optimal solution. In the next step the combined fitness values of seeds and plants is calculated until the fitness value converges to an optimal solution. The objective function of this algorithm depends upon the type of application the algorithm is used for. The objective function is utilized as the fitness function to achieve the optimized results using convergence technique. The step by step procedure of conventional IWO algorithm has been explained below.

2.2. Steps involved in conventional IWO

Step 1: The seeds are initialized depending upon the number of selected variables involved in the process over the probable search boundary. The initialization of seeds is random which means that the seeds are dispersed in a random manner in the solution space.

Step 2: The fitness of the seeds initialized is evaluated depending upon the fitness function (or) the objective function chosen for the optimization problem. These seeds then evolve into weed plants capable of producing new units.

Step 3: The evolved plants are arranged in a definite order (increasing or decreasing) and new seeds are produced by these plants depending upon its position in the sorted list of plants, starting with the maximum number of seeds produced by the best fit plant.

Step 4: The number of seeds to be produced by the plants varies linearly from N_{max} to N_{min} which is decided by the formula,

$$\text{Number of seeds} = \frac{F_i - F_{worst}}{F_{best} - F_{worst}}(N_{max} - N_{min}) + N_{min} \quad (1)$$

Step 5: The generated seeds are distributed normally over the search space with zero mean and a standard deviation that is varying σ_{iter} which is given by,

$$\sigma_{iter} = \left(\frac{iter_{max} - iter}{iter_{max}}\right)^n (\sigma_0 - \sigma_f) + \sigma_f \quad (2)$$

The non-linear modulation, n, index is used to traverse around the search space more efficiently and is generally assumed to be between 2 and 3.

Step 6: The fitness of each seed produced in the above steps is calculated along with the parent weeds and by means of com-

petitive exclusion, the seed-parent combinations that are least in fitness are eliminated and the number of weed plants is limited to the maximum of number of weeds allowed.
Step 7: The above steps are carried out until maximum iterations and the plant with the best fitness value at the end of it is the optimized solution.

2.3. Flowchart of conventional IWO algorithm

See Figs. 1–3.

3. UC problem formulation

The objective function in a UC problem is to minimize the fuel cost,

$$Min\sum_{t=1}^{T}\sum_{i=1}^{N}[C_i(p(i,t))I(i,t)+SU(i,t)+SD(i,t)] \quad (3)$$

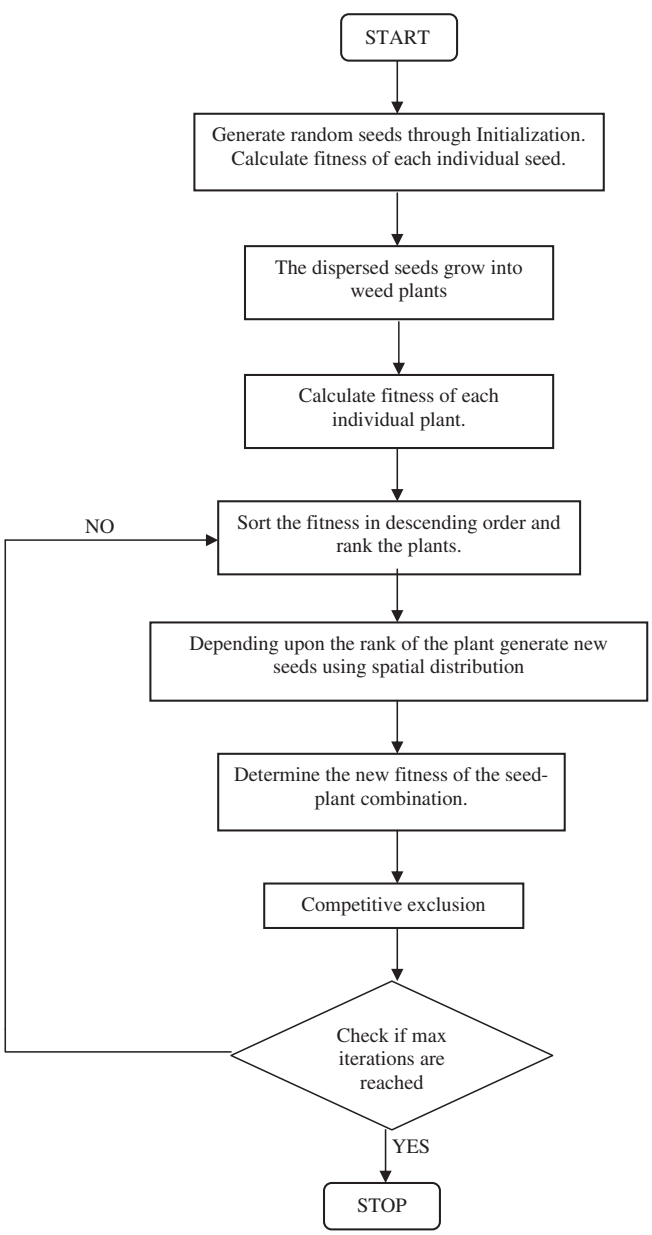

Fig. 1. Conventional IWO technique.

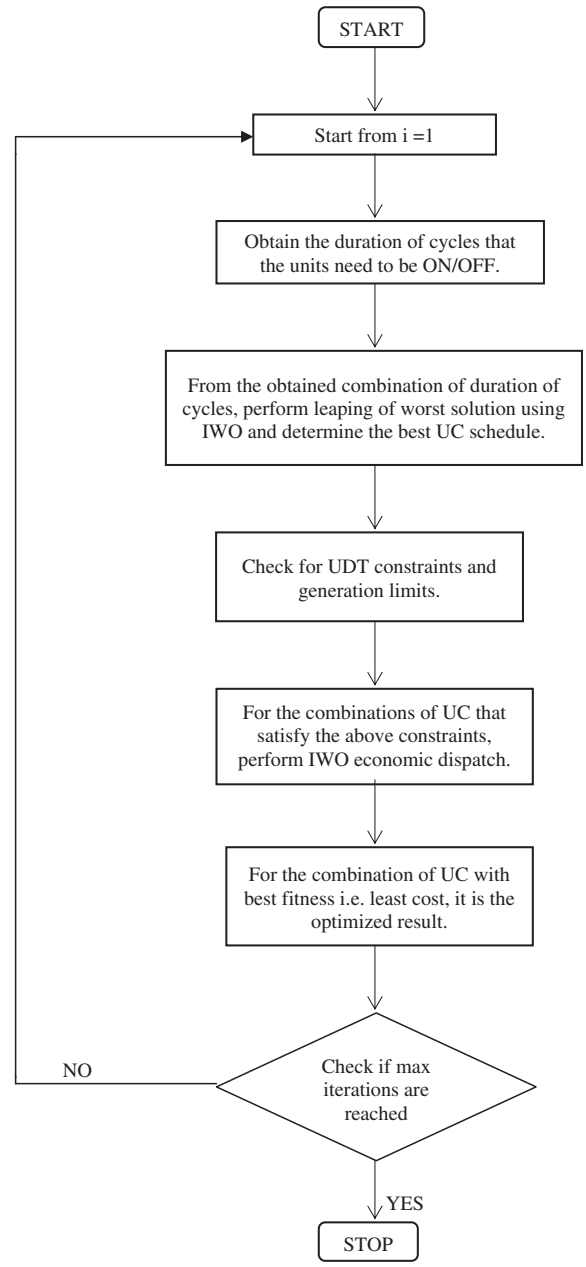

Fig. 2. IWO implementation on UC problem.

Fuel cost constitutes bulk of the operational cost. A generating unit is also subjected to several other constraints like [10–12],

$$FC(i) = a_iP_i^2 + b_iP_i + c_i \quad (4)$$

- Operational status of the unit
  The status of the unit whether it is ON/OFF before the scheduling of a new day begins. It is otherwise called as the initial status of the unit.

- Minimum up time and down time constraints

Up time refers to the minimum time that the unit must stay ON once switched ON before it can be switched OFF.
Down time refers to the minimum time that the unit must stay OFF once switched OFF before it can be switched ON.
A unit cannot be switched ON/OFF when it violates the up/down time constraints.

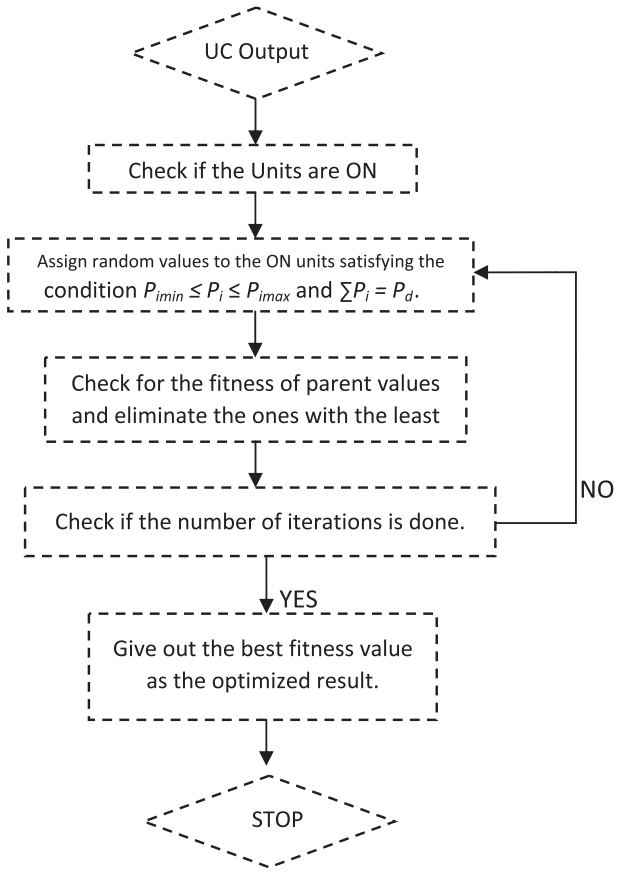

Fig. 3. Flowchart of proposed methodology.

$$(T_{i,\mathrm{ON}}(t-1) - UT_i)(I_i(t-1) - I_i(t)) \geqslant 0 \tag{5}$$

$$(DT_i - T_{i,\mathrm{OFF}}(t-1))(I_i(t-1) - I_i(t)) \geqslant 0 \tag{6}$$

- Satisfying demand requirements for a particular hour

$$\sum_{i=1}^{N} P_i(t) = P_d(t) \tag{7}$$

- Generation limit constraints

A generation unit must adhere to its minimum and maximum generation limits.

$$P_{i\min}(t) \leqslant P_i(t) \leqslant P_{i\max}(t) \tag{8}$$

- Sum of duration of cycles, i.e. the ON/OFF status of each unit should be equal to the scheduling horizon
- Spinning reserve constraints

It is the sum total of maximum possible generation from committed units in an hour minus the load demand in that particular hour.

$$\sum_{i=1}^{N} P_{i,\max} \cdot I_{i,t} = (1+a)\ell_t, \quad \text{where } i = 1,2,\ldots,N \quad \text{and} \quad t = 1,2,\ldots,T \tag{9}$$

4. Application of IWO on UC problem

4.1. Seed definition

In the problem of UC, the position of the seeds in the search space represents an integer ranging between {−*T*, +*T*} where *T* represents the scheduling horizon. These set of integers obtained can be split into a number of cycles which represent the sequence of ON/OFF of a unit. A positive number signifies that the unit is ON for as many hours as the cycle duration and a negative number signifies that the unit is OFF for the cycle duration. The number of cycles is determined from number of load peaks in the scheduling horizon and the sum of the minimum up and down times of the unit. In this paper for the given load and system input data, we take the number of cycles to be equal to 5. Increasing the number of cycles may lead to more complexity in the execution and reducing the number of cycles could significantly reduce the search space leading to sub optimal solutions.

4.2. Initial population of seeds

The initial, i.e. the first cycle input values are determined randomly from a set of equations. Initialize a cycle matrix depending upon the load data and the number of generating units for the given system. Randomize the values of first cycle duration keeping in mind the up/down time constraints using equation These equations make sure that the generated number automatically adheres to the minimum up time limit if the unit is to be switched OFF or minimum down time if the unit is to be switched ON. The value of the first cycle depends on the initial status of the unit. The value of the first cycle makes sure that the unit stays in the same state as the initial status for as long as required to satisfy the UDT constraints.

$$T_i^1 = \begin{cases} +Rand\left(\max\left(0, UT_i - T_i^0\right), T\right), & \text{if } T_i^0 > 0 \\ -Rand\left(\max\left(0, DT_i + T_i^0\right), T\right), & \text{if } T_i^0 < 0 \end{cases} \tag{10}$$

where T_i^0 is equal to the duration of the last cycle in the previous scheduling day. It is the initial status of a unit.

Initialize the values of other cycle durations using Eqs. (11) and (12) considering whether the unit is ON or OFF respectively. Generate several such matrixes using a definite number of iterations. For $k < K$, where k = number of the current cycle and K = total number of cycles which in this case is 5, durations of cycles are determined as per the following equations.

If the unit was OFF in the previous cycle, i.e. the duration of the previous cycle is a negative number then,

$$T_i^k = \begin{cases} +Rand\left(UT_i, PT_i^{k-1}\right), & \text{if } \left(PT_i^{k-1} > UT_i\right) \\ +PT_i^{k-1}, & \text{otherwise} \end{cases} \tag{11}$$

If the unit was ON in the previous cycle, then

$$T_i^k = \begin{cases} -Rand\left(DT_i, PT_i^{k-1}\right), & \text{if } \left(PT_i^{k-1} > DT_i\right) \\ -PT_i^{k-1}, & \text{otherwise} \end{cases} \tag{12}$$

where PT_i^{k-1} represents the number of hours pending after the last cycle.

$$PT_i^{k-1} = T - \sum_{p=1}^{k-1} T_i^p \tag{13}$$

Since the initial population is randomly determined, it is possible that the scheduling hours are met within $k < K$ cycles itself. If the sum of *c* cycles is equal to the scheduling horizon then the values from $k + 1$ to K are given as zero. This initial population of seeds or cycle durations are found to be in conformity with the unit up/down time constraints.

From the generated iterations, choose the matrix that is closest to the optimized solution. Optimized solution is one in which the load demand is met out using the generator units that has the cheapest operational cost.

Table 1
System input data for 10 units, 24 h.

Unit	P_{max} (MW)	P_{min} (MW)	a_0	a_1	a_2	t_{up} (h)	t_{down} (h)	S_{hr} ($)	S_{cr} ($)	t_{cold} (h)	Init. st
1	455	150	1000	16.19	0.00048	8	8	4500	9000	5	8
2	455	150	970	17.26	0.00031	8	8	5000	10,000	5	8
3	130	20	700	16.6	0.00200	5	5	550	1100	4	−5
4	130	20	680	16.5	0.00211	5	5	560	1120	4	−5
5	162	25	450	19.7	0.00398	6	6	900	1800	4	−6
6	80	20	370	22.26	0.00712	3	3	170	340	2	−3
7	85	25	480	27.74	0.00079	3	3	260	520	2	−3
8	55	10	660	25.92	0.00413	1	1	30	60	0	−1
9	55	10	665	27.27	0.00222	1	1	30	60	0	−1
10	55	10	670	27.79	0.00173	1	1	30	60	0	−1

Table 2
Load data for 10 units, 24 h.

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

4.3. Leaping of worst solution

After the first iteration, i.e. the initial population determination, it is not necessary that the obtained population of seeds or cycles duration are optimal. Therefore in order to determine the optimal solution, we incorporate an iterative approach with the values approaching towards the best optimal solution after each iteration.

From the obtained matrix, update the values of cycle durations so that they converge to the optimum solution using standard deviations. At the end of each iteration, the solution with the best and the worst fitness is identified. The worst fitness is updated towards the best fitness by using a normalized standard deviation given by the equation in 2. Now that the duration of cycles are changed, there will automatically be a change in the summation of these cycle duration values as they will no longer be adding up to the scheduling horizon. Therefore in order to correct this problem, we scale the obtained values after leaping so that the total sum of the duration is again equal to the total number of hours.

$$\left(T_i^1,\ldots,T_i^k\right)=\frac{T}{\sum_{g=1}^{K}\left|T_i^g\right|}\left(T_i^1,\ldots,T_i^K\right)\quad \text{where } i=1,2,3,\ldots,N \tag{14}$$

Since rand generates a number between 0 and 1 and since the duration of cycles can only be integers, we round off the obtained values to the nearest integer.

$$X_w^1=Round(NewX_w) \tag{15}$$

Since rounding off values would change the sum of the cycle durations, we take the last non-zero integer in the cycle values and change it into the number of pending hours after the summation till the previous cycle durations.

$$T_i^l=T-\sum_{i=1}^{l-1}T_i^g\quad \text{where } i=1,2,\ldots,N \tag{16}$$

4.4. Satisfying minimum up/down time constraints

After the leaping of the worst solution, it is possible that for certain duration cycles, the values violate the minimum up/down time constraints. Therefore an UDT check needs to be performed for each cycle duration value for each individual unit.

For $T_i^2>0$ and if $T_i^2<\max\left(0,UT_i-T_i^0\right)$, then the duration of cycles 1 and 2 of unit i are changed as follows,

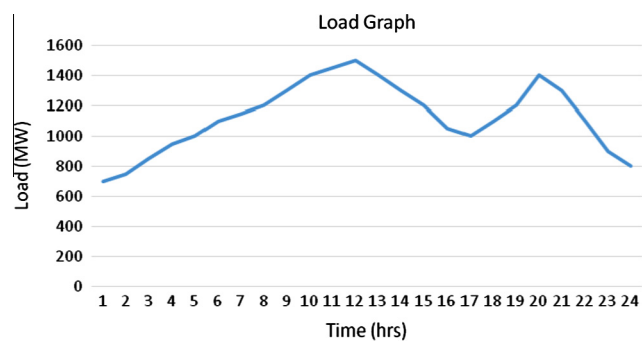

Fig. 4. Load demand for 24 h.

Table 3
Initial cycle values.

Units	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
1	12	−9	3	0	0
2	13	−10	1	0	0
3	−16	7	−1	0	0
4	−5	14	−5	0	0
5	−8	7	−6	3	0
6	−23	1	0	0	0
7	−18	3	−3	0	0
8	−6	6	−5	2	−5
9	−16	2	−3	3	0
10	−14	2	−3	3	−2

Table 4
Cycle vaues after updation.

Units	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
1	24	0	0	0	0
2	24	0	0	0	0
3	−2	21	−1	0	0
4	−3	18	−3	0	0
5	−5	10	−4	4	−1
6	−8	6	−5	*1*	−4
7	−18	3	−3	0	0
8	−6	6	−5	2	−5
9	−16	2	−3	2	−1
10	−14	2	−3	3	−2

$$\begin{cases} T_i^2 = T_i^2 - T_i^1 + \max\left(0, UT_i - T_i^0\right) \\ T_i^1 = \max\left(0, UT_i - T_i^0\right) \end{cases} \tag{17}$$

For $T_i^1 < 0$ and if $-T_i^1 < \max\left(0, DT_i + T_i^0\right)$, then the duration of cycles 1 and 2 of unit i are changed as follows,

$$\begin{cases} T_i^2 = T_i^2 - T_i^1 - \max\left(0, DT_i + T_i^0\right) \\ T_i^1 = \max\left(0, DT_i + T_i^0\right) \end{cases} \tag{18}$$

The duration of rest of the cycles that contain non-zero values are then checked for UDT.

If the unit is ON in the previous cycle (positive number) then,

$$\begin{cases} T_i^{k+1} = T_i^{k+1} - T_i^k + UT_i \\ T_i^k = UT_i \end{cases} \tag{19}$$

If the unit is OFF in the previous cycle (negative number) then,

$$\begin{cases} T_i^{k+1} = T_i^{k+1} - T_i^k + DT_i \\ T_i^k = DT_i \end{cases} \tag{20}$$

The updated matrix of cycle durations is converted into binary number combinations of 1s and 0s to get unit commitment schedule. This is done so because naturally the solution to UC problems is in terms of binary. This table of unit commitment schedule is then fed into economic dispatch programming.

4.5. Fitness function evaluation

Once the cycle durations are found out to be the closest to optimal, ED is performed with those combinations. In this paper, the economic dispatch is also performed using IWO algorithm and convergence of best solution is achieved using the same standard deviation technique obtained using Eq. (2). During generation of power for the units that are ON, the generation limits and load demand for the particular hour are to be kept in mind. The number of iterations carried out to determine the best fit combination of power is about 1000. The generation of power by the committed units in that particular hour is done in conformity with the individual unit's minimum and maximum generation limits [13] and considering the load demand balancing factor so that only those combinations that adhere to these two constraints will be considered for fitness function evaluation. Using this process about a 1000 fitness values i.e. operational costs are generated and the combination with the least operational cost is given as the optimal solution [14].

The objective function for economic dispatch is the fuel cost polynomial equation which is taken as the fitness function to evaluate the fitness of each hourly combination.

$$FC(P_i) = a_i P_i^2 + b_i P_i + c_i$$

The function FC gives the fuel cost in terms of $ for each hourly combination of units that satisfy the load demand. Apart from this the start up and the shut down costs are included in the objective function [15]. Start up costs are included when the unit is switched ON and shut down costs are included when the unit is switched OFF. Hence, the total objective function or the fitness function becomes the combination of fuel cost and start-up/shut-down cost which constitute the operational costs of an hour for a certain demand. Here we consider the shutdown cost of all generator units to be equal to 0 [16].

Table 5
Hourly schedule.

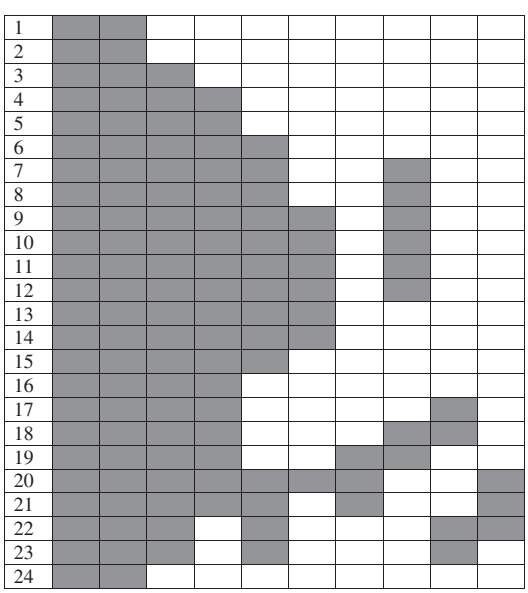

Table 6
Unit commitment schedule for 10 units, 24 h system.

Units/h	1	2	3	4	5	6	7	8	9	10
1	1	1	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0
3	1	1	1	0	0	0	0	0	0	0
4	1	1	1	1	0	0	0	0	0	0
5	1	1	1	1	0	0	0	0	0	1
6	1	1	1	1	1	0	0	0	0	0
7	1	1	1	1	1	0	0	1	0	0
8	1	1	1	1	1	0	0	1	0	0
9	1	1	1	1	1	1	0	1	0	0
10	1	1	1	1	1	1	0	1	0	0
11	1	1	1	1	1	1	0	1	0	0
12	1	1	1	1	1	1	0	1	0	0
13	1	1	1	1	1	1	0	0	0	0
14	1	1	1	1	1	1	0	0	0	0
15	1	1	1	1	1	0	0	0	0	1
16	1	1	1	1	0	0	0	0	0	1
17	1	1	1	1	0	0	0	0	1	0
18	1	1	1	1	0	0	0	1	1	0
19	1	1	1	1	0	0	1	1	0	0
20	1	1	1	1	1	1	1	0	0	1
21	1	1	1	1	1	0	1	0	0	1
22	1	1	1	0	1	0	0	0	1	1
23	1	1	1	0	1	0	0	0	1	0
24	1	1	0	0	0	0	0	0	0	0

5. Flowchart for implementation of IWO for UC

5.1. Flowchart for implementation of IWO for ed

6. Simulation, results and analysis

In this paper, UC analysis using IWO was performed on a 10 unit – 24 h system. In Table 1 the system input data is given [17]. Load data for the scheduled time period is given in Table 2. Fig. 4 consists of a graphical representation of load data. It can be seen that there are 5 points or rather 5 edges in the graph which determine the number of cycle durations. The initial cycle values and the values after updating are given in Tables 3 and 4 respectively. The unit commitment scheduling of all the 10 generators for 24 h is given in

Table 7
Generation dispatch for 10 units, 24 h system.

Units/h	1	2	3	4	5	6	7	8	9	10	Total gen (Mw)	Fuel cost ($)	Start-up cost ($)	Reserve %
1	440	260	0	0	0	0	0	0	0	0	700	13,684	0	30
2	413	337	0	0	0	0	0	0	0	0	750	14,555	0	21.33
3	390	353	107	0	0	0	0	0	0	0	850	16,926	550	22.35
4	331	422	106	91	0	0	0	0	0	0	950	19,268	560	23.16
5	448	374	93	85	0	0	0	0	0	0	1000	20,142	0	17
6	428	444	74	30	124	0	0	0	0	0	1100	22,414	900	21.09
7	397	385	83	74	162	0	0	49	0	0	1150	23,771	60	20.6
8	428	446	130	54	97	0	0	45	0	0	1200	24,666	0	15.58
9	449	432	115	85	130	41	0	48	0	0	1300	27,348	340	12.84
10	455	455	130	130	157	51	0	22	0	0	1400	28,933	0	4.78
11	455	455	130	130	162	78	0	40	0	0	1450	30,448	0	1.17
12	455	455	130	130	162	80	33	55	0	0	1500	31,142	520	3.47
13	455	455	130	130	162	68	0	0	0	0	1400	28,653	0	0.857
14	455	451	113	99	158	24	0	0	0	0	1300	26,435	0	8.61
15	455	455	125	122	32	0	0	0	0	11	1200	24,585	60	15.58
16	393	431	108	73	0	0	0	0	0	45	1050	21,490	0	21.42
17	361	384	95	68	0	0	0	0	0	0	1000	20,622	0	17
18	396	442	89	110	0	0	0	0	38	25	1100	22,808	90	16.36
19	437	434	127	90	0	0	75	37	0	0	1200	24,593	580	9.17
20	428	437	127	105	161	50	43	0	0	49	1400	30,229	1070	10.85
21	419	409	125	121	126	0	69	0	0	31	1300	27,348	0	12.85
22	392	410	117	0	131	0	0	0	31	19	1100	22,815	60	19.27
23	387	311	22	0	162	0	0	0	18	0	900	18,880	0	39.6
24	440	360	0	0	0	0	0	0	0	0	800	15,740	0	50.875
											Total	557,495	4790	

Table 8
Comparison of results obtained by different methods.

Method	Total start up cost ($)	Total production cost ($)	Total operational cost ($)
GA	–	–	565825.00
PSO	2095	562899.00	565804.00
HPSO	4090	559852.30	563942.30
SFLA	4090	559847.70	563937.70
IWO	4790	557495.00	562285.00

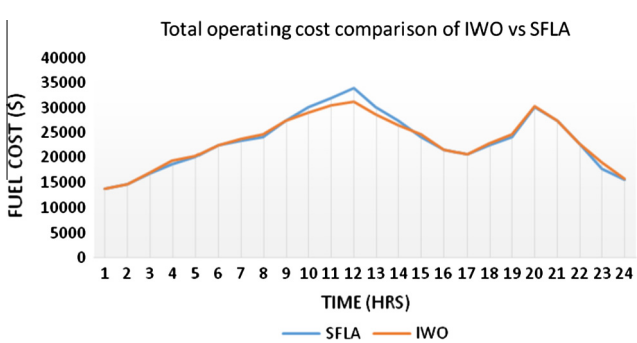

Fig. 5. Comparison of fuel cost (IWO vs. SFLA).

Tables 5 and 6. In Table 7 the generation dispatch using IWO is listed for each hour. The total cost of IWO is reasonably less when compared to SFLA. A comparison between the operating costs using different algorithms has been done in Table 8 and it is seen that the cost obtained through IWO is the cheapest for the taken system [3,4].

The load demand graph shown in Fig. 4 has 5 sharp points including the first and the last hour values. Hence the number of cycles is chosen to be 5 for this system. This number of cycle durations might vary depending upon the system and the load data (see Fig. 5).

In Table 3, the initial values obtained through randomization of the best matrix from a pool of matrix generated is given. As one can see that the values of the first cycle of the first two units that carry the chunk of the load is only around 12. This is therefore updated through the process of convergence step by step using standard deviation σ. The final cycle values after updation is given in Table 4.

The difference between the cycle values obtained through the proposed IWO methodology and that through other evolutionary algorithms like PSO [5], SFLA [3], and BFOA [13], is that the values obtained are better suited for cost reduction in IWO than in other algorithms. This is because several iterations of cycle values are carried out before choosing the best and the search field in IWO is maximized for each unit irrespective of how costly or cheap its generation is which makes it possible to achieve optimal solutions for every unit during every hour.

Fig. 4 shows a comparison of production cost at each hour between the proposed IWO algorithm and the Shuffled Frog Leaping Algorithm. It is seen that at most hours, the production cost obtained is much lesser using IWO when compared to that of SFLA. Even though the start up cost using the proposed methodology is higher compared to the results obtained through other algorithms, the production cost more than makes up for it thereby resulting in decreased total operational cost. This is shown in Table 8.

7. Conclusion

This paper has proposed the application of invasive weed optimization algorithm for solving the problem of UC. The problem of unit commitment being a challenging problem requires algorithms that could effectively produce best results in terms of operational cost. The fast convergence and optimal solution properties of the proposed methodology yields better UC results when compared to the UC results obtained through already tested evolutionary algorithms [19,20]. The results of UC have been obtained for 10 units and 24 h system. It is seen that the total operational cost obtained through IWO is much lesser when compared to the results obtained through other evolutionary algorithms. This technique has the flexibility to achieve optimized results and give least operating costs for much larger systems as well [21]. This algorithm

could provide better solutions to the problem of unit commitment in deregulated power systems [22].

References

[1] Wood AJ, Wollenberg F. In: Power generation, operation and Control. John Wiley Publisher; 1984.
[2] Virmani S, Adrian EC, Imhof K, Muhherjee S. Implementation of a Lagrangian based unit commitment problem. IEEE Trans Power Syst 1989;4(4):1373–80.
[3] Ebrahimi Javed, Hosseinian Seyed Hossein, Ghahrepetian Gevorg B. Unit commitment problem solution using shuffled frog leaping algorithm. IEEE Trans Power Syst 2011;26(2):573–81.
[4] Kazarlis SA, Bakirtzis AG, Petridis V. A genetic algorithm solution to the unit commitment problem. IEEE Trans Power Syst 1996;11(1):83–92.
[5] Xing WG, Wu FF. Genetic algorithm based unit commitment with energy contracts. Int J Electr Power Energy Syst 2002;24(5):329–36.
[6] Ting TO, Rao MVC, Loo CK. A novel approach for unit commitment problem via an effective hybrid particle swarm optimization. IEEE Trans Power Syst 2006;21(1):411–8.
[7] Sepehri Rad H, Lucas C. A recommender system based on invasive weed optimization algorithm. In: IEEE congress on evolutionary computation. CEC; 2007.
[8] Ahmadi Mohamadreza, Mojallali Hamed, Izadi-Zamanabadi Roozbeh. State estimation of nonlinear stochastic systems using a novel meta-heuristic particle filter. Swarm Evol Comput 2012;4(6):44–53.
[9] Karimkashi S, Kishk Ahmed A. Invasive Weed Optimization and its Features in Electromagnetics. IEEE Trans Antenn Propag 2010;58(4):1269–78.
[10] Mehrabian AR, Lucas C. A novel numerical optimization algorithm inspired from weed colonization. Ecol Inform 2006;1:355–66.
[11] Sen S, Kothari DP. Optimal thermal generating unit commitment – a review. Int J Electr Power Energy Syst 1998;20(7).
[12] Padhy Narayana Prasadh. Unit commitment – a bibliographical survey. IEEE Trans Power Syst 2004;19(2):1196–205.
[13] Eslamian Morteza, Hosseinian Seyed Hossein, Vahidi Behro. Bacterial foraging based solution to the unit commitment problem. IEEE Trans Power Syst 2009;24(3):1478–88.
[14] Sathish Kumar K, Rajaram R, Tamilselvan V, Shanmugasundaran V, Naveen S, Mohamed Nowfal IG, et al. Economic dispatch with valve point effect using various PSO Techniques. Int J Recent Trends Eng 2009;2(6):130–5.
[15] Sharma R, Nayak Niranjan, Krishnanand KR, Rout PK. Modified invasive weed optimization with dual mutation technique for dynamic economic dispatch. In: IEEE international conference on energy, automation and signals (ICEAS 2011), 28–30 December, 2011. p. 1–6.
[16] Wong KP, Wong YW. Genetic and genetic/simulated-annealing approaches to economic dispatch. IEEE Proc Gener Transm Distrib 1994;141(5):507–13.
[17] Saravanan B, Vasudevan ER, Kothari DP. A solution to unit commitment problem using invasive weed optimization algorithm. In: Proc ICPEC, IEEE international conference, February 2013.
[18] Morales-Espana German, Latorre Jesús M, Ramos Andrés. Tight and compact MILP formulation for the thermal unit commitment problem. IEEE Trans Power Syst 2013. http://dx.doi.org/10.1109/TPWRS.2013.2251373.
[19] Elyas Seyyed Hamid, Foroud Asghar Akbari, Chitsaz Hamed. A novel method for maintenance scheduling of generating units considering the demand side. Int J Electr Power Energy Syst 2013;51(10):201–12.
[20] Hosseinnezhad Vahid, Babaei Ebrahim. Economic load dispatch using θ-PSO. Int J Electr Power Energy Syst 2013;49(7):160–9.
[21] Yuang-Kang Wu, Chin-Cheng Huang, Chun-Liang Lin. Resolution of unit commitment problem by using the hybrid Taguchi-ant colony system algorithm. Int J Electr Power Energy Syst 2013;49(7):188–98.
[22] Lopez Juan Alvarez et al. A heuristic algorithm to solve for a real life large scale power systems. Int J Electr Power Energy Syst 2013;49(7):287–95.