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-