Expert Systems with Applications 38 (2011) 9729–9735

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Generation maintenance scheduling in power systems using ant colony optimization for continuous domains based 0–1 integer programming

Abdolvahhab Fetanat [a], Gholamreza Shafipour [b,*]

[a] Department of Electrical Engineering, Behbahan Branch, Islamic Azad University - Behbahan-Iran
[b] Department of Computer Engineering, Behbahan Branch, Islamic Azad University - Behbahan-Iran

ARTICLE INFO

Keywords:
Ant colony optimization for continuous domains
Integer programming
Maintenance scheduling
Optimization

ABSTRACT

In this paper, we present a formulation that enables Ant Colony Optimization for Continuous Domains (ACO_R) to seek the optimal solution of the unit maintenance scheduling problem. ACO_R is a direct extension of Ant Colony Optimization (ACO). Also it is the significant ant-based algorithm for continuous optimization. For the maintenance scheduling, cost reduction is as important as reliability. The objective function of this algorithm considers the effect of economy as well as reliability. Various constraints such as spinning reserve, duration of maintenance crew are being taken into account. The ACO_R formulation developed is applied on a power system with six generating units. The simulation result of this technique is compared with those reported in literature. The outcome is very encouraging and proves that the authors' proposed approach outperforms them in terms of reaching a better optimal solution and speed.

1. Introduction

The power station maintenance department exists to help the production function to maximize plant reliability, availability and efficiency by determining both short and long term maintenance requirements and by carrying out the work accordingly. This includes work to comply with statutory and mandatory requirements and investigations into plant problems. The department has to make the most economic use of its available resources; this is achieved, in part, by having a level of staff (engineering, supervisory, craft) to deal with the general day-to-day steady workload and by making alternative arrangements to cater for work load peaks (Mohammadi Tabari, Pirmoradian, & Hassanpour, 2008).

To achieve the above goal, periodic servicing must take place and normally falls under the following items (Mohammadi Tabari et al., 2008):

(1) Planned maintenance: overhaul, preventive maintenance.
(2) Unplanned maintenance: emergency maintenance.

Preventive maintenance is expensive. It requires shop facilities, skilled labor, keeping records and stocking of replacement parts. However, the cost of downtime resulting from avoidable outages may amount to ten or more times the actual cost of repair. The high cost of downtime makes it imperative to economic operation that maintenance be scheduled into the operating schedule (Mohammadi Tabari et al., 2008).

The maintenance scheduling problem is to determine the period for which generating units of an electric power utility should be taken off line for planned preventive maintenance over the course of a 1 or 2 year planning horizon, in order to minimize the total operating cost while system energy, reliability requirements and a number of other constraints are satisfied (Marwali & Shahidehpour, 1998a,b).

2. Solution approaches

In the recent decade, many efforts which are categorized as follows have been done in the maintenance scheduling field:

(1) Harmony search algorithm (Fetanat & Shafipour, 2009).
(2) Linear programming (Chattopadhyay, 1998; Guntsch & Middendorf, 2002).
(3) Mixed integer programming (Da ilva, Schilling, & Rafael, 2000).
(4) Decomposition methods (Marwali & Shahidehpour, 2000).
(5) Goal programming (Moro & Ramos, 1994).
(6) Tabu search (El-Amin, Salih, & Mohammed, 2000).
(7) Simulated annealing (Mohammadi Tabari et al., 2008).
(8) Genetic algorithm (Huang, 1997).
(9) Fuzzy logic (Leou, 2001).
(10) Neural networks (Mohammadi Tabari et al., 2008).
(11) Expert systems (El-Sharkh & El-Keib, 2003).
(12) meta heuristic-based hybrid approaches (Dahal & Chakpitak, 2007).

* Corresponding author.
E-mail addresses: av-fetanat@behbahaniau.ac.ir (A. Fetanat), pp_shafipour@yahoo.com (G. Shafipour).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.02.027

(13) Particle swarm optimization (Siahkali & Vakilian, 2009).
(14) Ant colony optimization (Foong, 2007).
(15) Deterministic approaches (Marwali & Shahidehpour, 1998a; b).

3. ACO for continuous domain – ACO_R (Socha & Dorigo, 2008)

Similarly to a combinatorial optimization problem (COP), also a model for continuous optimization problem (CnOP) may be formally defined:

A model $Q = (S, \Omega, f)$ of a CnOP consists of:

- a search space **S** defined over a finite set of continuous decision variables and a set Ω of constraints among the variables;
- an objective function $f = S \rightarrow \mathbb{R}_0^+$ to be minimized.

The search space **S** is defined as follows: Given is a set of continuous variables X_i, $i = 1, \ldots, n$, with values $v_i \in D_i \subseteq \mathbb{R}$. A variable instantiation, that is, the assignment of a value v_i to a variable X_i, is denoted by $X_i \leftarrow v_i$. A solution $s \in S$ – i.e., a complete assignment, in which each decision variable has a value assigned – that satisfies all the constraints in the set Ω, is a feasible solution of the given CnOP. If the set Ω is empty, Q is called an unconstrained problem model, otherwise is called a constrained one. A solution $s^* \in S$ is called a global optimum if and only if: $f(s^*) \leqslant f(s) \forall\, s \in S$. The set of all globally optimal solutions is denoted by $S^* \subseteq S$. Solving a CnOP requires finding at least one $s^* \in S$.

The idea that is central to the way ACO works is the incremental construction of solutions based on the biased (by pheromone) probabilistic choice of solution components. In ACO applied to combinatorial optimization problems, the set of available solution components is defined by the problem formulation. At each construction step, ants make a probabilistic choice of the solution component c_i from the set $N(s^p)$ of available components according to Eq. (1).

$$p(c_{ij}|s^p) = \frac{\tau_{ij}^{\alpha}.\eta(c_{ij})^{\beta}}{\sum_{c_{il} \in N(s^p)} \tau_{il}^{\alpha} \cdot \eta(c_{ij})^{\beta}}, \quad \forall c_{ij} \in N(s^p) \tag{1}$$

The probabilities associated with the elements of the set $N(s^p)$ make up a discrete probability distribution (Fig. 1a) that an ant samples in order to choose a component to be added to the current partial solution s^p.

The fundamental idea underlying ACO_R is the shift from using a discrete probability distribution to using a continuous one, that is, a probability density function (PDF) (Fig. 1b). In ACO_R, instead of choosing a component $c_{ij} \in N(s^p)$ according to Eq. (1), an ant samples a PDF. In the following sections we explain how this is accomplished. In Section 3.1, we present briefly the idea of using a PDF, and in Section 3.2 we outline the pheromone representation used in ACO_R. Finally, in Section 3.3 we give a detailed description of the ACO_R algorithm itself.

3.1. Probability density function (PDF)

Before going into in-depth description of the ACO_R algorithm, we must discuss certain characteristics of PDFs, and select the one that we will use. In principle, a probability density function may be any function $P(x) \geqslant 0 \forall\, x$ such that:

$$\int_{-\infty}^{\infty} P(x)dx = 1. \tag{2}$$

For a given probability density function $P(x)$, an associated cumulative distribution function (CDF) $D(x)$ may be defined, which is often useful when sampling the corresponding PDF. The CDF $D(x)$ associated with PDF $P(x)$ is defined as follows:

$$\int_{-\infty}^{x} P(t)dt. \tag{3}$$

The general approach to sampling PDF $P(x)$ is to use the inverse of its CDF, $D^{-1}(x)$. When using the inverse of the CDF, it is sufficient to have a pseudo-random number generator that produces uniformly distributed real numbers. However, it is important to note that for an arbitrarily chosen PDF $P(x)$, it is not always straightforward to find $D^{-1}(x)$.

One of the most popular functions that are used as a PDF is the Gaussian function. It has some clear advantages, such as a reasonably easy way of sampling – e.g., the Box–Muller method (Box & Muller, 1958) – but it also has some disadvantages. A single Gaussian function is not able to describe a situation where two disjoint areas of the search space are promising, as it only has one maximum. Due to this fact, we use a PDF based on Gaussian functions, but slightly enhanced – a Gaussian kernel PDF. Similar constructs have been used before (Bosman & Thierens, 2002), but not exactly in the same way. We define a Gaussian kernel as a weighted sum of several one-dimensional Gaussian functions $g_l^i(x)$ and denote it as $G^i(x)$:

$$G^i(x) = \sum_{l=1}^{k} \omega_l g_l^i(x) = \sum_{l=1}^{k} \omega_l \frac{1}{\sigma_l^i \sqrt{2\pi}} e^{-\frac{\left(x-\mu_l^i\right)^2}{2\sigma_l^{i2}}}. \tag{4}$$

Since we use as many Gaussian kernel PDFs as the number of dimensions of the problem, $i = 1, \ldots, n$ identifies a single such PDF. The Gaussian kernel $G^i(x)$ is parameterized with three vectors of parameters: ω is the vector of weights associated with the individual Gaussian functions, μ^i is the vector of means, and σ^i is the vector of standard deviations. The cardinality of all these vectors is equal to the number of Gaussian functions constituting the

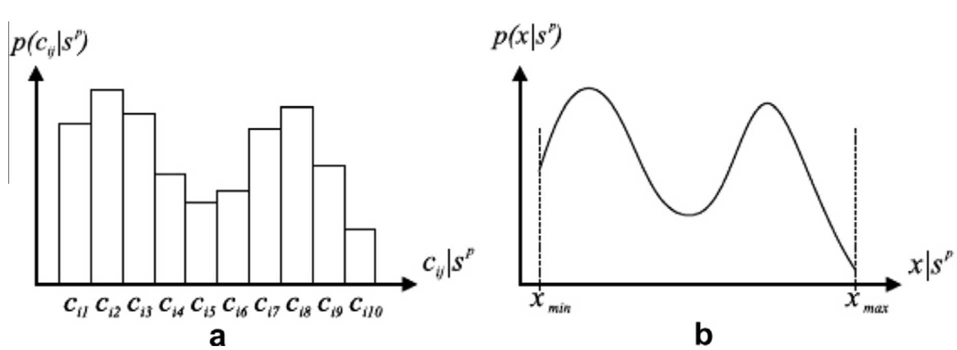

Fig. 1. (a) Discrete probability distribution $P_d(c_{ij}|s^p)$ of a finite set $P_d(c_{ij}|s^p) \in N(s^p)$ of available components. (b) Continuous probability density function $P_c(x|s^p)$ with possible range $x \in [x_{\min}, x_{\max}]$. The y-axis on both plots indicates the probability p. Note that $\sum_{j=1}^{10} P(c_{ij}|s^p) = \int_{x_{\min}}^{x_{\max}} P(x|s^p)dx = 1$.