Applied Soft Computing 43 (2016) 131–142

Contents lists available at ScienceDirect

Applied Soft Computing

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

Optimization of fuzzy controller design using a new bee colony algorithm with fuzzy dynamic parameter adaptation

Camilo Caraveo, Fevrier Valdez, Oscar Castillo*

Tijuana Institute of Technology, Tijuana Mexico

ARTICLE INFO

Article history:
Received 5 August 2015
Received in revised form
15 December 2015
Accepted 20 February 2016
Available online 27 February 2016

Keywords:
Fuzzy logic
Fuzzy controller
Linguistic variables
BCO
FBCO
Dynamic parameter adjustment

ABSTRACT

In this paper we are presenting a modification of a bio-inspired algorithm based on the bee behavior (BCO, bee colony optimization) for optimizing fuzzy controllers. BCO is a metaheuristic technique inspired by the behavior presented by bees in nature, which can be used for solving optimization problems. First, the traditional BCO is tested with the optimization of fuzzy controllers. Second, a modification of the original method is presented by including fuzzy logic to dynamically change the main parameter values of the algorithm during execution. Third, the proposed modification of the BCO algorithm with the fuzzy approach is used to optimize benchmark control problems. The comparison of results show that the proposed fuzzy BCO method outperforms the traditional BCO in the optimal design of fuzzy controllers.

1. Introduction

In optimization problems, the main objective is to find the best alternative among a set of possible solutions. In some cases the space of solutions for the problems is just too big and this can cause that the time to find the very best solution is prohibitive large. On the other hand, there are different areas of computational intelligence that provide a set of techniques for solving search and optimization problems [6,29,36]. Such techniques can provide highly competitive results, but not the best solutions. In addition there are alternative methods using heuristic algorithms, but these do not guarantee to find the best solution, although they are able to find a good solution in a reasonable time.

Population based algorithms constitute a new paradigm of collective intelligence and they are able to find good solutions to optimization problems with a reasonable cost and time [10,13,36]. These algorithms have become a research topic of recent interest to many scientists in the field of artificial intelligence [2,25,33]. Collective intelligence can be defined as a set of metaheuristic techniques of artificial intelligence based on the study of collective behavior systems present in nature, generally in a decentralized and self-organizing fashion [2].

There are many works on search and optimization algorithms that have been applied to a plethora problems, for example; in [1] an Artificial Bee Colony (ABC) algorithm to tune optimal rule-base of a Fuzzy Power System Stabilizer (FPSS) is presented, which leads to damp low frequency oscillation following disturbances in power systems. In [3] the optimization of the type-1 and type-2 fuzzy controller design for the water tank using the bee colony optimization method is presented. In [11] the design of an optimal fuzzy logic-PID controller using bee colony optimization for frequency control in an isolated wind-diesel system is presented. In [12] a bee colony optimization based-fuzzy logic-pid control design of electrolyzer for microgrid stabilization is presented. In [39] an optimal fuzzy load frequency controller with simultaneous auto-tuned membership functions and fuzzy control rules is presented. In [8] the optimization of fuzzy controllers design using the bee colony algorithm is presented and the BCO algorithm is used to optimize the parameters of membership functions of a fuzzy logic controller.

The main contribution of this paper is the modification of the bee colony optimization algorithm (BCO), by adding the fuzzy approach to dynamically change its parameters. In the literature there are similar works using the bee colony algorithm to solve different problems, for example [1,11], where the main difference with respect to our proposal is the algorithm that was used. We use the BCO bee colony optimization (created by Teodorovic in 2002) [30–32] unlike to the algorithm used in [39], which is the artificial bee colony (created by Karaboga [19]), both bio-inspired algorithms are based on the behavior of bees, but with different features that are described below. The ABC algorithm has: follower bees, scout bees, and working bees, and the selection metric is different. The BCO algorithm has: follower bees, scout bees, and the process in the algorithm is different. In this case, the bees use a step forward and a step back, in step forward the bees explores new solutions, and in the step back the bees share information of new solutions with the bees of the hive, and this defines the role of follower bees or recruiter bees, therefore these are different algorithms.

There exists a previous publication by the authors using the same BCO algorithm with the difference that in the present paper, fuzzy logic for dynamic adjustment of parameters is now applied to different benchmark control problems. In the previous work, the algorithm is only used to adjust the parameters of the membership functions of a simple fuzzy controller.

The paper is organized as follows. The methodology is explained in Section 3, the theory and operation of the algorithm are described in Section 2, the cases used in this work are described in Section 4, the experimental results with the traditional method and with dynamic parameter adaptation using fuzzy logic and the statistical comparison between the two algorithms are described in Section 5. Finally, in Section 6 the conclusions are presented.

* Corresponding author. Tel.: +52 6646236318.
E-mail address: ocastillo@tectijuana.mx (O. Castillo).

http://dx.doi.org/10.1016/j.asoc.2016.02.033
1568-4946/

2. Bee colony optimization (BCO)

The proposed algorithm is based on the artificial bee colony (BCO, bee colony optimization), which is one of the recently proposed algorithms in the area of collective intelligence [30,33,34]. The present paper focuses on the model proposed by Lucic and Teodorović in 2001, motivated by observing the intelligent behavior of honey bee's swarms [19,20,31,32].

Bee colony optimization (BCO) is a meta-heuristic algorithm [13,20] that belongs to the class of nature-inspired algorithms. These algorithms are inspired by various biological and natural processes [20]. Natural systems have become an important source of ideas and models for the development of many artificial systems [6].

This method (BCO) uses an analogy based on the way the bees do their foraging in nature, and the way in which they apply their search optimization methods to find optimal routes between the hive and the source food [20]. The basic idea behind BCO is to build the multi-agent system (artificial bee colony) [21] efficiently able to solve difficult combinatorial optimization problems [20,23].

2.1. Elements and behavior of BCO

The model defines three main components as shown below [20,21]:

1) Source of food: the value of a food source depends on many factors, including its proximity to the hive, wealth or food concentration and ease of food extraction.
2) Working bees: they are associated with a current or exploited food source. They carry with them information about that particular source, its distance, location and return to share, with a certain probability.
3) Scout bees: they are in constant search of a food source.

2.1.1. Construction of a path by an artificial bee

In this model, a bee is allowed to explore and find a full path travel when leaving the hive the bees observe random dances performed by other bees. Then these bees are equipped with an array of movements of the observed dances.

This set of moves, referred to as "preferred path" is denoted as θ, and it will serve as a guide in the process of foraging. θ contains a complete path that was previously explored by its partner who will lead the bee to the destination.

During foraging, a bee goes from a node to another node until the destination is reached. In the model bee a heuristic rule is used to help transition the bee in its decision making about what node to visit next.

This rule is composed of two factors: the arc fitness and the distance heuristic. The arc fitness is calculated by Eq. (1) for all possible paths that can be visited by a bee on a particular node at a particular moment.

A bee is aided by a transition rule in decision making of the next node to visit, as is shown in Eq. (1). The state transition probability, $\boldsymbol{P_{ij,n}}$, gives the probability of moving from node $\boldsymbol{i}$ to node $\boldsymbol{j}$ after $\mathbf{N}$ transitions. This is formally defined in Eq. (1) [37]

$$p_{ij,n} = \frac{[\rho_{ij,n}]^{\alpha} \cdot \left[\frac{1}{d_{ij}}\right]^{\beta}}{\sum\limits_{j \in A_{i,n}} [\rho_{ij,n}]^{\alpha} \cdot \left[\frac{1}{d_{ij}}\right]^{\beta}} \tag{1}$$

where $\boldsymbol{\rho_{ij,n}}$ is the arc fitness from node $\boldsymbol{i}$ to node $\boldsymbol{j}$ after n transitions and $\boldsymbol{d_{ij}}$ represents the distance between node $\boldsymbol{i}$ and node $\boldsymbol{j}$. Note that the $\boldsymbol{P_{ij,n}}$ is inversely proportional to the node distance. In other words, the shorter the distance, the higher is the likelihood of that node to be selected. $\boldsymbol{\alpha}$ is a binary variable that turns on or off the arc fitness influence in the model. $\boldsymbol{\beta}$ is used to control the significance level of the heuristic distance [37].

The main references in the literature do not provide values for the alpha and beta parameters, and we consider using for beta the range from 2 to 7, and for alpha between 0 and 1 because of its similarity with the ant colony algorithm. The alpha parameter has more relevance than beta in the equation, because if alpha is null, the bee decides to use the heuristic distance, makes an assessment of the nearby nodes and determines which nodes to visit depending on the value of the fitness.

The arc fitness, $\boldsymbol{\rho_{ij,n}}$, is defined as in Eq. (2). Where $|\boldsymbol{A_{i,n}} \cap \boldsymbol{F_{i,n}}|$ is 1 when there is a common instance in both $\boldsymbol{A_{i,n}}$ and $\boldsymbol{F_{i,n}}$, or 0 otherwise. $\boldsymbol{A_{i,n}} - \boldsymbol{F_{i,n}}$ denotes the difference between sets $\boldsymbol{A_{i,n}}$ and $\boldsymbol{F_{i,n}}$. It contains all elements of $\boldsymbol{A_{i,n}}$ that are not present in $\boldsymbol{F_{i,n}}$. When there is only one node left in $\boldsymbol{A_{i,n}}$, $\boldsymbol{\rho_{ij,n}}$ is set to 1 to indicate that the node is the only remaining choice.

This happens at the last transition before a bee revisits the start node in order to complete the tour [20]

$$\rho_{ij,n} = \left\{ \begin{array}{ll} \boldsymbol{\lambda}, & j \in F_{i,n},\ |A_{i,n}| > 1 \\ \dfrac{1-\lambda |A_{i,n} \cap F_{i,n}|}{|A_{i,n} - F_{i,n}|}, & j \notin F_{i,n},\ |A_{i,n}| > 1 \\ 1, & |A_{i,n}| = 1 \end{array} \right\} \begin{array}{l} \forall j \in \boldsymbol{A}_{i,n}, \\ \mathbf{0} \le \lambda \le 1 \end{array} \tag{2}$$

3. Algorithm with dynamic adjustment using Fuzzy Logic

This section is dedicated to describing the modification of the BCO algorithm with a fuzzy logic approach for dynamically adjusting the values of the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ parameters. The proposed methodology is described below [15,18,26,28,37]. The particular objective in choosing the BCO algorithm is because there are few research works published of this algorithm, there also exist other variants and we consider a good idea to use it and analyze if good results can be achieved when compared with respect to other algorithms.

3.1. Method for adaptation of parameters

In Fig. 1 we graphically illustrate the sequence of steps of the proposed algorithm:

The pseudocode of the proposed algorithm is presented as follows:

Step 1: Randomly generate the initial population of **n** scout bees for the MF parameters.

To start the algorithm a set of random source foods is generated, where each row (vector) represents a bee and the bee has in memory a route and that route is a possible solution to the problem, which in this case represents the values of the membership function parameters of the fuzzy controller.

This initial population must have possible candidate solutions that satisfy the constraints. Set NC = 0, and evaluate the fitness value of the initial populations by (6).

Step 2: Select the **m** best sites for the neighborhood search.

The bee selected memory contains the best path found so far, and to define the fitness of each bee the equation of the mean square error is used, which is Eq. (9).

Step 3: Recruit bees are used for selected sites (more bees for the best sites).

Step 4: Represent the new value of MF from each working bee.

Step 5: Select the fittest bees from each path.

Step 6: Check the stopping criterion. If satisfied, terminate the search, else NC = N C + 1, show and save the best values found and return to evaluate new solutions.