



# Integrating the artificial bee colony and bees algorithm to face constrained optimization problems



Hsing-Chih Tsai\*

Department of Construction Engineering, National Taiwan University of Science and Technology, Taiwan

Ecological and Hazard Mitigation Engineering Researching Center, National Taiwan University of Science and Technology, Taiwan

## ARTICLE INFO

### Article history:

Received 18 September 2012

Received in revised form 28 August 2013

Accepted 5 September 2013

Available online 13 September 2013

### Keywords:

Constrained optimization problem

Swarm intelligence

Artificial bee colony

Bees algorithm

## ABSTRACT

Swarm intelligence (SI) has generated growing interest in recent decades as an algorithm replicating biological and other natural systems. Several SI algorithms have been developed that replicate the behavior of honeybees. This study integrates two of these, the artificial bee colony (ABC) and bees algorithms (BA), into a hybrid ABC–BA algorithm. In ABC–BA, an agent can perform as an ABC agent in the ABC sub-swarm and/or a BA agent in the BA sub-swarm. Therefore, the ABC and BA formulations coexist within ABC–BA. Moreover, the population sizes of the ABC and BA sub-swarms vary stochastically based on the current best fitness values obtained by the sub-swarms.

This paper conducts experiments on six constrained optimization problems (COPs) with equality or inequality constraints. In addressing equality constraints, this paper proposes using these constraints to determine function variables rather than directly converting them into inequality constraints, an approach that perfectly satisfies the equality constraints. Experimental results demonstrate that the performance of the ABC–BA approximates or exceeds the winner of either ABC or BA. Therefore, the ABC–BA is recommended as an alternative to ABC and BA for handling COPs.

© 2013 Elsevier Inc. All rights reserved.

## 1. Introduction

Most search and optimization problems used in science and engineering involve inequality and/or equality constraints that the obtained optimal solutions must satisfy. Because constrained optimization problems (COPs) are frequently encountered in real world applications, significant attention and efforts have been invested in their efficient and effective resolution. A COP is usually written as a nonlinear programming problem of the following type. For minimization problems:

$$\begin{aligned}
 & \text{Minimize } f(\vec{x}) \\
 & \text{subject to } g_i(\vec{x}) \leq 0, \quad i = 1, \dots, Ng \\
 & \quad \quad \quad h_j(\vec{x}) = 0, \quad j = 1, \dots, Nh \\
 & \quad \quad \quad x_k^L \leq x_k \leq x_k^U, \quad k = 1, \dots, D
 \end{aligned} \tag{1}$$

where  $f$  is a  $D$ -dimensional function;  $x$  represents function variables;  $g$  denotes less-than-equal-to inequality constraints;  $Ng$  is number of  $g$  constraints;  $h$  means equality constraints; and  $Nh$  is number of  $h$  constraints. All  $x$  variables have lower and upper bounds  $[x^L, x^U]$ .

\* Address: #43, Sec. 4, Keelung Rd., Taipei 106, Taiwan. Tel.: +886 2 27301277/76663; fax: +886 2 27301074.

E-mail address: [tsaihingchih@gmail.com](mailto:tsaihingchih@gmail.com)

Evolutionary algorithms, especially genetic algorithms (GA), have been widely used in recent decades to solve COPs [24]. Powell and Skolnick mapped feasible and infeasible solutions into two different intervals within a GA environment [35]. Deb used GA to propose a constraint handling method that frees settings of penalty numbers [4]. Douglas et al. analyzed COPs with employed niched pareto GA and penalty methods [7]. Coello compiled a comprehensive survey of popular constraint handling techniques used with evolutionary algorithms [3]. Barkat Ullah et al. developed a hybrid evolutionary algorithm based on GA and devised a local search technique that emphasizes equality constraint problems [1].

Swarm intelligence (SI) has generated growing interest as an algorithm that replicates biological and other natural systems. Optimization algorithms have been proposed and employed in various applications, and applications related to SI have demonstrated great promise. SI is the property of a system, in which coherent functional global patterns emerge due to the collective behaviors of agents interacting locally with their environment. SI algorithms include ant colony optimization, particle swarm optimization (PSO), artificial fish swarm, artificial bee colony (ABC) and gravitational search algorithm, among others [6,8–11,21,22,26,27,29,36,37,40,42–45]. Sun et al. applied an improved vector PSO to solve COPs using a constrained-preserving method [39]. Kou et al. proposed a co-evolutionary PSO with infeasible degree to address COPs [19]. Masuda and Kurihara used a multi-objective PSO to minimize both objective functions and total amount of constraint violations [23]. ABC performance on COPs has also been investigated [17,25]. However, as no single algorithm is able to achieve optimal results for all algorithm applications [46], researchers are currently exerting significant efforts to further improve existing SI algorithms and develop new algorithms inspired by natural phenomena.

Bee or bee-colony algorithms are fairly new members of SI. It is widely presumed that only some aspects of the nature/behavior of honeybees can be exploited and that new characteristics can be added to create new classes of algorithms [47]. In recent years, the unique nature of honeybee colonies has inspired numerous researchers to develop various new bee-inspired algorithms such as virtual bee, the bees, BeeAdHoc, the marriage in honeybees, the BeeHive, bee system, bee colony optimization, and artificial bee colony [18]. The bee colony optimization algorithm has since been further developed and applied to transportation problems [41]. Yang initially proposed the virtual bee algorithm and demonstrated how it could solve two-dimensional numerical problems [47]. The bees algorithm (BA) originally proposed by Pham et al. is used to solve unconstrained function optimization problems and train multi-layered perceptron networks to recognize different patterns in control charts [29–34]. Basturk and Karaboga proposed the artificial bee colony (ABC) algorithm and used it to solve unconstrained and constrained function optimization problems [2,12–16].

Among all bee-inspired algorithms, ABC is considered the most effective and currently accounts for over half of bee SI applications [18]. However, BA directly performs bee SI based on perceptions of the natural behavior of bees and its basic concepts are superior to those of other algorithms. Experience has demonstrated that neighborhood search (*ngh*) is a key BA parameter. This study used a stochastic self-adaptive *ngh* to free its setting and improve BA performance as well. While sharing common concepts, both ABC and BA provide unique advantages. This paper thus proposes to integrate ABC and BA (ABC–BA) and apply the resulting ABC–BA algorithm to COPs. Moreover, this paper uses an infeasible degree selection method similar to that of [19] to handle the inequality constraints. This paper tests and compares the efficacy of two methods that use, respectively, tolerance values and functional relationships to convert equality constraints into inequality constraints.

The remainder of this paper is organized as follows: Section 2 presents the ABC and BA algorithms; Section 3 introduces the designs of the ABC–BA algorithm; Section 4 conducts experiments on 6 well-known benchmark functions of COPs; and Section 5 provides a discussion and conclusions.

## 2. Artificial bee colony and bees algorithm

### 2.1. Artificial bee colony

Basturk and Karaboga [2] proposed the artificial bee colony (ABC) algorithm and used ABC in a wide range of applications [18]. The collective intelligence of bee swarms consists of three essential components: food sources, employed bees, and unemployed bees. Unemployed bees are further segregated into onlookers and scouts [12–16]. The three main phases of ABC are the employed phase, onlooker phase, and scout phase. Each employed bee searches for a new candidate location  $cX_s$  from its old location  $x_s$  using the following formula:

$$cX_{sd} = x_{sd} + \varphi_{st}(x_{sd} - x_{td}), \quad t \neq s, \quad t \in \{1, 2, \dots, NE\}, \quad d \in \{1, 2, \dots, D\} \quad (2)$$

where  $s$  denotes the bee index of an employed bee;  $t$  is a random bee index that must be different from  $s$ ;  $NE$  is the number of employed bees;  $d$  is a random dimension index in  $D$ ;  $D$  is the problem dimension; and  $\varphi_{st}$  is a random number in the range of  $[-1, 1]$ . If the food source of  $cX_s$  is better than that of  $x_s$ ,  $cX_s$  replaces  $x_s$ ; otherwise,  $x_s$  is retained. There are  $NO$  onlookers and each onlooker chooses a food source  $x_s$  depending on the probability  $p_s$  associated with its fitness  $fit_s$ .

$$p_s = \frac{fit_s}{\sum fit_s} \quad (3)$$

$$fit_s = \begin{cases} 1/(fv(x_s) + 1), & fv(x_s) \geq 0 \\ 1 + |fv(x_s)|, & fv(x_s) < 0 \end{cases} \quad (4)$$

Download English Version:

<https://daneshyari.com/en/article/391744>

Download Persian Version:

<https://daneshyari.com/article/391744>

[Daneshyari.com](https://daneshyari.com)