Applied Soft Computing 26 (2015) 143–148

Contents lists available at ScienceDirect

Applied Soft Computing

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

The Bisection–Artificial Bee Colony algorithm to solve Fixed point problems

P. Mansouri [a,b,*], B. Asady [a], N. Gupta [b]

[a] Department of Mathematics, Arak Branch, Islamic Azad University, Arak, Iran
[b] Department of Computer Science, Delhi University, Delhi, India

ARTICLE INFO

Article history:
Received 29 September 2012
Received in revised form 18 May 2014
Accepted 3 September 2014
Available online 28 September 2014

Keywords:
Bisection method
Fixed point problems
Artificial Bee Colony algorithm

ABSTRACT

In this paper, we introduce a novel iterative method to finding the fixed point of a nonlinear function. Therefore, we combine ideas proposed in Artificial Bee Colony algorithm (Karaboga and Basturk, 2007) and Bisection method (Burden and Douglas, 1985). This method is new and very efficient for solving a non-linear equation. We illustrate this method with four benchmark functions and compare results with others methods, such as ABC, PSO, GA and Firefly algorithms.

1. Introduction

Solving equations is one of the most important problem in engineering and science. The *bisection method* in mathematics, is a root-finding method that repeatedly bisects an interval and then selects a subinterval in which a root must lie for further processing so that the range of possible solutions is halved in each iteration. This is a very simple and robust method, but it is also relatively slow. Thus, it is often used to obtain a rough approximation to a solution which is then used as a starting point for more rapidly converging methods [1]. The *bisection method* is also called the *binary search method* because of its similarity to the binary search algorithm [1] in computer science.

Bio-inspired algorithms are amongst the most powerful algorithms for the optimization problems [2–7,18], especially for the NP-hard problems like the *traveling salesman problem* and others. Particle Swarm Optimization (PSO) algorithm was developed by Kennedy and Eberhart in 1995 [8] based on the swarm behavior(/intelligence) such as that of fishes and birds schooling in nature. Though particle swarm optimization has many similarities with genetic algorithms, it is much simpler because it does not use mutation/crossover operators. Instead, it uses the real-number randomness and the global communication among the swarming particles. In this sense, it is also easier to implement. The Firefly Algorithm(FA) was introduced by X.S. Yang in 2009 [9] for multimodal optimization applications. He compared FA algorithm with other metaheuristic algorithms such as Particle swarm optimization (PSO).

Motivated by the intelligent behavior of honey bees, Dervis Karaboga gave Artificial Bee Colony algorithm(ABC) [10,11] in 2005. It is as simple as Particle Swarm Optimization (PSO) and Differential evolution (DE) algorithms, Genetic algorithm (GA) [12], biogeography based optimization (BBO) algorithm, and uses only common control parameters such as colony size and maximum cycle number. ABC as an optimization tool, provides a population-based search procedure in which food positions are modified by the artificial bees with time and the bee's aim is to discover the places of food sources with high nectar amount and finally the one with the highest nectar. In ABC system, artificial bees fly around in a multidimensional search space and some (employed and onlooker bees) choose food. Development of an ABC algorithm for solving generalized assignment problem, which is known to be NP-hard, is presented in detail along with some comparisons in [13]. Sources, depending on their experience and the experience of their nest mates, adjust their positions. Some applications of the ABC algorithm to solve hard problems are presented in [14–17]. In this paper, we introduce a novel iterative method that combines the advantages of both the Bisection method and the Artificial Bee Colony algorithm to solve the hard fixed point problem.

In Section 2, fixed point problem is defined and brief overview of the ABC algorithm and the Bisection method are described. In Section 3, we explain and discuss about our method. In Section 4, we

* Corresponding author at: Department of Mathematics, Arak Branch, Islamic Azad University, Arak, Iran. Tel.: +98 9187590319.
E-mail addresses: pmansouri393@yahoo.com, p-mansouri@iau-arak.ac.ir, pmansouri393@gmail.com (P. Mansouri).

http://dx.doi.org/10.1016/j.asoc.2014.09.001
1568-4946/

Repeat

1. Give the transcendental equation $x = g(x)$,
2. Give initial guess in interval $[a, \quad b]$,
3. Do $x_{i+1} = g(x_i)$

until (D_1 or D_2 is true)

Fig. 1. Fixed Point iteration scheme.

Input: A continuous function $f(x)$, interval [a,b], tolerance: TOL; maximum number of iterations: N_0

1. If $(b-a)/2 < TOL$ then exit
2. Set $i = 1$
3. while $i \leq N_o$ and $(b-a)/2 \geq TOL$ do
   Set $c = a + (b-a)/2$
   Compute $f(c)$
   If $f(c) = 0$ then output c and exit
   If $f(a).f(c) > 0$ then set $a = c$ and $f(a) = f(c)$
   else set $b = c$ and $f(b) = f(c)$
   Set $i = i + 1$.
4. If $(b-a)/2 < TOL$ then output c, 'tolerance limit exceeded '
5. If $i > N_o$ then output c, 'number of iterations reached N_o'

Fig. 2. Bisection method to compute the roots of a function.

compare accuracy and complexity of proposed method with other algorithms on four benchmark functions. In Section 5, we discuss about the ability of the proposed method.

2. Preliminaries

In this section, we define the fixed point problem and briefly explain the ABC algorithm and the Bisection method.

2.1. Fixed point of a function

In mathematics, a fixed point(invariant point) of a function is a point that is mapped to itself by the function. In other word, a number c is a fixed point for a given function g if $g(c) = c$. A set of fixed points is sometimes called a fixed set. For example, benchmark function $g_3 = 20 + e - 20e^{-0.2\sqrt{1/x^2}} - e^{1/\cos(2\pi x)}$, $x \in [1, 21]$ has a fixed point (see in Section 3), but not all functions have fixed points. Function $g(x) = x + 1$, $x \in R$, has no fixed points, since x is never equal to $x + 1$ for any real number.

An iterative method for solving equation $g(x) = x$ is the recursive relation $x_{i+1} = g(x_i)$, $i = 0, 1, 2, \ldots$, with some initial guess x_0. The algorithm stops when one of the following stopping criterion is met:

- D_1: total number of iterations is N, for some N, fixed apriori.
- D_2: $|x_{i+1} - x_i| < \epsilon$ for some ϵ, fixed apriori (Fig. 1).

1. Initial food sources are produced for all employed bees.
2. Repeat UNTIL (requirements are met)
   (a) Each employed bee goes to a food source in her memory and determines a neighbor source, then evaluates its nectar amount and dances in the hive.
   (b) Each onlooker watches the dance of employed bees and chooses one of their sources depending on the dances, and then goes to that source. After choosing a neighbor around that, she evaluates its nectar amount.
   (c) Abandoned food sources are determined and are replaced with the new food sources discovered by scouts. item The best food source found so far is registered.

Fig. 3. Artificial Bee Colony algorithm.