



Numerical optimization using synergetic swarms of foraging bacterial populations

Sotirios P. Chatzis^{a,*}, Spyros Koukas^b

^a Department of Electrical and Electronic Engineering, Imperial College London, United Kingdom

^b Department of Electrical and Computer Engineering, National Technical University of Athens, Greece

ARTICLE INFO

Keyword:

Evolutionary optimization

ABSTRACT

The bacterial foraging optimization (BFO) algorithm is a popular stochastic, population-based optimization technique that can be applied to a wide range of problems. Two are the major issues the BFO algorithm is confronted with: first, the foraging mechanism of BFO might in some cases induce the attraction of bacteria gathered near the global optimum by bacteria gathered to local optima, thus slowing down the whole population convergence. Second, BFO is susceptible to the curse-of-dimensionality, which makes it significantly harder to find the global optimum of a high-dimensional problem, compared to a low-dimensional problem with similar topology. In this paper, we introduce a novel BFO-based optimization algorithm aiming to address these issues, the synergetic bacterial swarming optimization (SBSO) algorithm. Our novel approach consists of: (i) the introduction of the swarming dynamics of the particle swarm optimization algorithm in the context of BFO, in order to ameliorate the convergence issues of the BFO bacteria foraging mechanism; and (ii) the utilization of multiple populations to optimize different components of the solution vector cooperatively, so as to mitigate the curse-of-dimensionality issues of the algorithm. We demonstrate the efficacy of our approach on several benchmark optimization problems.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

In recent years, many researchers have considered imitating food search strategies of living organisms, and concepts from natural selection of species, in an effort to provide highly effective global optimization algorithms. *Evolutionary algorithms* (EAs), including genetic algorithms (GA) (Whitley, 1994), and evolutionary programming (EP) (Fogel, 1995), are among the first algorithms of this kind. EAs have enjoyed wide success due to their simplicity, robustness and flexibility, being applied to various tasks such as robot design (Frutiger, Bongard, & Iida, 2002), power system economic dispatching (Wu & Ma, 1995; Wu, Cao, & Wen, 1998), etc. However, their slow convergence rates, due to the nature of the entailed search procedures, has made their widespread application in large optimization problems rather unpopular. The introduction of hybrid evolutionary algorithms has arisen as a means to remedy this issue; however, hybrid EAs require the objective function to be differentiable, which is rarely the case in real-life applications. Recently, two *swarm intelligence* paradigms, inspired by food search strategies of living organisms, have been developed to tackle these issues of EAs: the first one is ant colony optimization

(ACO) (Coloni, Dorigo, & Maniezzo, 1991; Dorigo & Gambardella, 1997), inspired by ant routing behavior; the second one is particle swarm optimization (PSO) (Kennedy & Eberhart, 1995), simulating animal swarm behavior.

More recently, the foraging mechanism of bacterial populations has functioned as another source of inspiration for designing global optimization algorithms. A foraging organism takes necessary action to maximize the energy utilized per unit time spent for foraging, considering all the constraints presented by its own physiology such as sensing and cognitive capabilities as well as the environment. Based on this observation, several models have been developed to mimic bacterial foraging behaviors and have been applied to solving practical problems. The most renowned such algorithm is the bacterial foraging optimization (BFO) algorithm, presented in the seminal paper of Passino (2002). Since its introduction in 2002, the study of BFO has received great attention from the computational intelligence community. BFO has been successfully applied to a number of problems, such as adaptive control (Kim & Cho, 2005b), harmonic estimation (Mishra, 2005), transmission loss reduction (Tripathy, Mishra, Lai, & Zhang, 2006), machine learning (Kim & Cho, 2005a), optimal power system stabilizers design (Das & Venayagamoorthy, 2006), and optimal power flow scheduling (Tang, Wu, & Saunders, 2006; Tang, Li, He, Wu, & Saunders, 2006).

* Corresponding author.

E-mail address: soterios@mac.com (S.P. Chatzis).

Although BFO has prominent and encouraging performance for global optimization problems, it has also been shown that the convergence of the algorithm can be problematic, leading to long execution times or even getting trapped to local optima. Indeed, in BFO, the chemotactic process is artificially set, imposing that the bacteria swarm together and keep a safe distance from each other. However, such a mechanism suppresses to some extent the algorithm convergence, since it could allow bacteria in nutrient-poor areas to attract bacteria in nutrient-rich areas, thus slowing down the convergence speed of the entire population. The repelling effect among bacteria also prevents the population from gathering together. To resolve these issues, inspired by the swarming mechanism of the PSO algorithm, a novel principle of swarming is introduced in the context of BFO in Tang, Wu, and Saunders (2007), under which the bacterial swarming algorithm (BSA) has been obtained. The BSA algorithm is based on the adjustment of each bacterium position according to the neighborhood environment and the position of the best bacterium, alternatively. Experimental results have shown that the swarming mechanism of BSA allows for faster and more efficient convergence, preventing the bacteria from being trapped into local optima.

Despite these advances, bacterial foraging-based optimization algorithms continue to suffer from the *curse-of-dimensionality* issue, which, in essence, implies a significant performance decrease as the dimensionality of the search space increases. The curse-of-dimensionality is a well-known problem most stochastic optimization algorithms (including PSO and GAs) suffer from, and results from the very definition of the problem being handled. Indeed, an evolutionary search-based optimization algorithm converges upon generation of a solution that falls in the optimality region, that is a small volume of the search space surrounding the global optimum. However, the probability of generating a sample inside the optimality region is the volume of the optimality region divided by the volume of the search space; thus, this probability decreases exponentially as the dimensionality of the search space increases. Consequently, finding the global optimum becomes significantly harder as the dimensionality of the problem increases.

One way to overcome this exponential increase in difficulty is to partition the search space into lower-dimensional subspaces, as long as the optimization algorithm can guarantee that it will be able to search every possible region of the search space. Such an approach has been applied, e.g., in the case of the PSO algorithm, yielding a number of cooperative approaches toward PSO (Asanga, Saman, & Harry, 2004; Jie, Zeng, Han, & Wang, 2008; Luo & Yi, 2008; Niu, Zhu, He, & Wu, 2008; van den Bergh & Engelbrecht, 2004). Inspired by these results, in this paper, we try to address all the aforementioned issues of the BFO algorithm, by introducing a novel optimization scheme, based on synergetic swarms of foraging bacterial populations. The proposed algorithm, dubbed synergetic bacterial swarming optimization (SBSO) algorithm, combines the ideas of introducing a PSO-like swarming mechanism in the context of BSO, and using synergetic evolving populations working in different parts of the search space, so as to both enhance the convergence rates of the BFO algorithm, and make it less susceptible to the curse-of-dimensionality. Our novel approach is evaluated using a number of benchmark mathematical functions which cover a wide range of optimization problems from uni-modal and multi-modal to low and high dimensions.

The remainder of this work is organized as follows: in Section 2, a brief presentation of the BFO and PSO algorithms is provided. In Section 3, the proposed SBSO algorithm is introduced. In Section 4, the experimental evaluation of the proposed algorithm using benchmark mathematical functions is conducted,

and its performance is compared to PSO, BFO, and other rival methodologies. Finally, in the last section, we conclude this paper.

2. Preliminaries

2.1. Bacterial foraging optimization

The original BFO algorithm consists of three principal mechanisms, namely, *chemotaxis*, *reproduction*, and *elimination-dispersal*. *Chemotaxis* is the activity of bacteria gathering to nutrient-rich areas in a spontaneous fashion; in this context, a cell-to-cell communication mechanism is established to simulate the biological behavior of bacterial movement. *Reproduction* comes from the concept of natural selection; under this procedure, only the best adapted bacteria tend to survive and transmit their genetic characters to succeeding generations, while the less adapted ones tend to perish. *Elimination-dispersal* events randomly select parts of the bacteria population to diminish and disperse into random positions in the environment; this way the algorithm ensures the diversity of the species, and prevents getting trapped to local optima. In the following, we provide some details on each one of these processes.

Algorithm 1: BFO Algorithm

Consider a P -dimensional optimization problem. Let S be the number of bacteria in the population. Let also N_s be the number of bacterial movements within one chemotactic cycle, N_c be the number of chemotactic cycles during one reproduction cycle, N_r be the number of reproductive cycles during one elimination-dispersal cycle, and N_{ed} be the number of elimination-dispersal cycles.

[Step 1] Initialize parameters $S, N_s, N_c, N_r, N_{ed}, C(i), P_{ed}$, and the initial bacteria positions.

[Step 2] Elimination-dispersal loop: $l = l + 1$.

[Step 3] Reproduction loop: $k = k + 1$.

[Step 4] Chemotactic loop: $j = j + 1$. For each bacterium $i = 1:S$, take a chemotactic step. For this purpose, conduct a maximum of N_s bacterial movements using Eq. (1), with the first movement being always a tumble, i.e. $\delta(i)$ is randomly selected. Continue until we reach N_s steps or if the total bacterial fitness function (3) deteriorates between two consecutive steps.

[Step 5] If $j < N_c$, go to step 4. In this case, continue chemotaxis since the life of the bacteria is not over.

[Step 6] Reproduction: sort the bacteria in the population based on their total health function accumulated during the previous chemotactic loop, expressed by (4). The $S/2$ bacteria with the highest J_{health} values die, and the remaining $S/2$ bacteria with the best values split, with the copies being placed at the same location as their parent.

[Step 7] If $k < N_{re}$, go to step 3. In this case, we have not reached the number of specified reproduction steps, so we start the next generation of the chemotactic loop.

[Step 8] Elimination-dispersal: eliminate and disperse each bacterium in the population with probability P_{ed} . To do this, if a bacterium is eliminated, simply disperse a replacement bacterium to a random location on the optimization domain. If $l < N_{ed}$, then go to step 2; otherwise, end.

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات