Procedia Computer Science
Volume 80, 2016, Pages 804–813
ICCS 2016. The International Conference on Computational Science

Enhancing Particle Swarm Optimization with Socio-cognitive Inspirations

Iwan Bugajski[1], Piotr Listkiewicz[1], Aleksander Byrski[1], Marek Kisiel-Dorohinicki[1], Wojciech Korczynski[1], Tom Lenaerts[2], Dana Samson[3], Bipin Indurkhya[4], and Ann Nowé[5]

[1] AGH University of Science and Technology, Faculty of Computer Science, Electronics and Telecommunications, Department of Computer Science, Al. Mickiewicza 30, 30-059 Krakow, Poland
{iwan.bugajski,piotr.listkiewicz}@gmail.com,{olekb,doroh,wojciech.korczynski}@agh.edu.pl
[2] Université Libre de Bruxelles Campus de la Plaine ULB CP212, Boulevard du Triomphe 1050 Bruxelles, Belgium tlenaert@ulb.ac.be
[3] Université Catholique de Louvain IPSY Pl. Cardinal Mercier 10 bte. L3.05.01 Louvain-la-Neuve, Belgium dana.samson@uclouvain.be
[4] Institute of Philosophy, Jagiellonian University ul. Grodzka 52 31-044 Krakow, Poland bipin@agh.edu.pl
[5] Vrije Universiteit Brussel Boulevard de la Plaine 2 1050 Ixelles, Belgium anowe@como.vub.ac.be

Abstract

We incorporate socio-cognitively inspired metaheuristics, which we have used successfully in the ACO algorithms in our past research, into the classical particle swarm optimization algorithms. The swarm is divided into species and the particles get inspired not only by the global and local optima, but share their knowledge of the optima with neighboring agents belonging to other species. Our experimental research gathered for common benchmark functions tackled in 100 dimensions show that the metaheuristics are effective and perform better than the classic PSO. We experimented with various proportions of different species in the swarm population to find the best mix of population.

Keywords: Metaheuristic computing, Swarm Intelligence, Nature-inspired computing

1 Introduction

Many algorithmic techniques in computer science are inspired by natural and biological processes. For example, decomposition of the population in evolutionary and similar computing techniques popularized such solutions as island model of evolution [2] that effectively enhanced the diversity of the computing metaheuristics. In the same vein, hybrid algorithms incorporating memetic, immunological, and hierarchical processes have been implemented [9]. In our research project, we are following the same approach by bringing different inspirations together

 Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2016

doi:10.1016/j.procs.2016.05.370

to yield interesting algorithms that are always needed (as stated by the famous No Free Lunch Theorem [11]).

In our earlier research, we experimented with socio-cognitive Ant Colony Optimization (ACO) metaheuristics [6, 8] by introducing into classic ACO a number of sub-species along with positing inter-relations among them, and by leveraging stigmergic relations among these artificial ants. These relations incorporated the ants' abilities to change perspective, to get inspired by the solutions found by other species, and so on. [6]. This approach produced efficient results in discrete optimization (solving different instances of TSP).

Our goal here is to extend the proposed socio-cognitive inspirations by applying it (to the possible extent) to Particle Swarm Optimization (PSO). Our motivation for attempting this extension is as follows. In the case of ACO, our goal was to build a common knowledge base about the solutions found by different species and share them. This was relatively easy to implement because the "derivation" of this knowledge was present in the pheromone table. In the case of PSO, this knowledge about global and local current optima is also quite natural, present and easy accessible in the system. Therefore, we plan to introduce various species that will take different inspirations from the current optima of the other species, and to test the resulting algorithms on the common benchmark functions.

In the next section, we provide the brief background, followed by a description of the socio-cognitive inspirations that are the basis of our approach. Then we introduce the classic PSO, and describe the socio-cognitive modification of this algorithm. Finally, we present the results of our experiments to evaluate the resulting algorithms, and then the conclusions.

2 PSO and its multi-species alternatives

Particle swarm optimization [5] is an iterative algorithm commonly used for mathematical optimization of certain problems. PSO belongs to a set of algorthims called metaheuristics - these algorithms do not guarantee finding the most optimum solution, but can yield a solution close to it. This fact makes PSO suitable for solving problems where either there is no known algorithm or execution of an exact algorithm consumes too much time or resources. Moreover, PSO does not require that the function being optimized be differentiable, regular or constant over time.

Particle swarm optimization was originally proposed for simulating social behavior, and was used for simulating the group movement of fish schools, bird flocks, and so on. But the algorithm was also found to be useful for performing mathematical optimization after some simplification.

In the past, there have been other attempts to decompose the PSO into sub-populations and sub-species, and here we would like to mention a few such notable approaches.

In [10], the authors demonstrated a cooperative PSO: they employed many swarms and made them optimize different parts of the solution vector in cooperation.

In another study[7], the authors modified PSO by introducing different swarms sharing information about their best solutions in order to escape the local extrema. Improved results were obtained in continuous optimization using this approach.

In [4], the author extended the original PSO by dividing the particle swarm spatially into multiple clusters called species in a multi-dimensional search space. Each species explored a different area of the search space to find the global or local optima along that dimension.

In [12], the number of particles in the sub-swarms were dynamically adjusted, yielding a competitive performance with respect to the examined benchmarks.