Parallel Approaches for the Artificial Bee Colony Algorithm

Rafael Stubs Parpinelli[1,2], César Manuel Vargas Benitez[2], and Heitor Silvério Lopes[2]

[1] Santa Catarina State University (UDESC) - Joinville - SC - Brazil
parpinelli@joinville.udesc.br
[2] Federal University of Technology Paraná (UTFPR) - Curitiba - PR - Brazil
cbenitez@cpgei.ct.utfpr.edu.br, hslopes@utfpr.edu.br

Abstract. This work investigates the parallelization of the Artificial Bee Colony Algorithm. Besides a sequential version enhanced with local search, we compare three parallel models: master-slave, multi-hive with migrations, and hybrid hierarchical. Extensive experiments were done using three numerical benchmark functions with a high number of variables. Statistical results indicate that intensive local search improves the quality of solutions found and, thanks to the coevolution effect, the multi-population approaches obtain better quality with less computational effort. A final comparison between models was done analyzing the trade-offs between quality of solution and processing time.

Keywords: Swarm Intelligence, Parallelism, Bee Foraging, ABC Algorithm.

1 Introduction

In the beginning of the research in the area of Swarm Intelligence, the mainstream paradigms were Ant Colony Optimization[1] [9] and Particle Swarm Optimization[2] [16]. ACO is inspired by the foraging behavior of ants and the PSO is motivated by the simulation of fish schools and bird flocks social behavior. Over the years, both methods have been applied successfully in a vast range of problems [8]. Notwithstanding, in recent years several other swarm intelligence algorithms have appeared, and, amongst them, those inspired by specific behaviors of honey bees, such as bees foraging [12, 19] and bees mating [11]. Currently, there are a variety of algorithms inspired by the bee foraging behavior found in literature, such as: Bee System [21], Honey Bee Algorithm [17], BeeHive [29], Virtual Bee Algorithm [30], Bee Colony Optimization [27], Bees Swarm Optimization [10], Bees Algorithm [19], Honey Bee Foraging [2] and the Artificial Bee Colony Algorithm [12].

This work focuses on the Artificial Bee Colony Algorithm (ABC), which was first proposed by Karaboga [12] for solving multi-dimensional and multi-modal

[1] ACO repository: http://iridia.ulb.ac.be/~mdorigo/ACO/
[2] PSO Repository: http://www.particleswarm.info

B.K. Panigrahi, Y. Shi, and M.-H. Lim (Eds.): Handbook of Swarm Intelligence, ALO 8, pp. 329–345.
springerlink.com

optimization problems. The ABC algorithm is a population-based algorithm inspired by the foraging behavior of bees. In this metaphor, bees are the possible solutions to the problem, and they fly within the environment (the search space) to find the best food source location (best solution).

Two remarkable features of such swarm-based systems are self-organization and decentralized control that leads to an emergent behavior. Emergent behavior is a property that emerges through interactions among system components (bees) and it is not possible to be achieved by any of the components of the system acting alone [4]. Bees interactions occur through a waggle dance performed inside the hive. It is used to recruit other bees to exploit a food source. The quality of a food source is proportional to the intensity of the waggle dance performed by a bee. Hence, best food sources lead in a more intense waggle dance, and this can reinforce the exploitation of the best food locations.

To date, the Artificial Bee Colony Algorithm (ABC) is one of the most widely used bee algorithm for problem solving. Some successful applications found in the literature using it include: generalized assignment problem [3], energy distribution network configuration [24], neural networks training [15], multi-objective problems [18], template matching in digital images [7]. A recent work [14] compared the ABC algorithm performance against other evolutionary computation algorithms (Genetic Algorithm, Particle Swarm Optimization, Differential Evolution and Evolution Strategies) upon several benchmark functions. Results showed that the performance of the ABC was better than or similar to those of the other algorithms. Another relevant work concerning the ABC algorithm analyzed the tuning of control parameters [1]. More about the ABC algorithm can be found in the repository[3].

It is known that population-based metaheuristics such as Genetic Algorithms [6][26], PSO [28] and ACO [5][25] can explore efficiently the use of parallel concepts to speed up the search process. In [28] the authors used a master-slave model to compare both synchronous and asynchronous PSO versions in an engineering problem. A PSO master-slave model was also used by [22] to solve a biomechanical problem that involves finding the kinematic structure of an ankle joint model. In [25] both master-slave and the island models of an ACO were compared using several instances of the well-known Traveling Salesman Problem.

To the best of our knowledge, to date, no application that explores parallelization methods for the ABC algorithm was found in the literature. Hence, this work aims at applying to the ABC algorithm concepts of parallel processing widely used in other population-based metaheuristics. This work compares the performance of the standard Karaboga model [14] with an Enhanced Sequential version (ES-ABC) and other three parallel models, namely: Master-Slave ABC (MS-ABC); Multi-Hive ABC with migrations (MH-ABC); and Hybrid Hierarchical ABC (HH-ABC).

Experiments were done using three well-known mathematical functions. An important feature is that the test functions are purposely set with a high number of variables in order to turn the problem solving process extremely difficult to all

[3] ABC Repository: http://mf.erciyes.edu.tr/abc/