A Parallel Ant Colony Optimization Algorithm Based on Crossover Operation

Adem Kalinli¹ and Fatih Sarikoc²

¹ Department of Computer Technologies, Erciyes University, Kayseri Vocational High School, Kayseri, Turkey. kalinlia@erciyes.edu.tr

² Department of Computer Engineering, Erciyes University, Institute of Science and Technology, Kayseri, Turkey. fsarikoc@yahoo.com

Abstract

In this work, we introduce a new parallel ant colony optimization algorithm based on an ant metaphor and the crossover operator from genetic algorithms. The performance of the proposed model is evaluated using well-known numerical test problems and then it is applied to train recurrent neural networks to identify linear and nonlinear dynamic plants. The simulation results are compared with results using other algorithms.

Keywords: Parallel Ant Colony Optimization, Hybrid Algorithms, Continuous Optimization, Recurrent Neural Network, System Identification

1 Introduction

There are many combinatorial optimization problems of the NP-hard type, and they cannot be solved by deterministic methods within a reasonable amount of time. The great difficulty of optimization problems encountered in practical areas such as production, control, communication and transportation has motivated researchers to develop new powerful algorithms. Therefore, several heuristics have been employed to find acceptable solutions for difficult real-world problems. The most popular of these new algorithms include genetic algorithms (GAs), simulated annealing (SA), ant colony optimization (ACO), tabu search (TS), artificial immune system (AIS), and artificial neural networks (ANNs) [1–4]. Although all of these algorithms converge to a global optimum, they cannot always guarantee optimum solutions to the problem. Therefore, they are called approximate or heuristic algorithms.

The ACO algorithm is an artificial version of the natural optimization process carried out by real ant colonies. The first ACO algorithm was proposed by Dorigo et al. in 1991, and was called an ant system (AS) [5, 6]. Real ants communicate with
each other by leaving a pheromone substance in their path, and this chemical substance leads other ants. Thus, stimergy is provided and swarm intelligence emerges in the colony behaviour. The main features of the algorithm are distributed computation, positive feedback and constructive greedy search. Since 1991, several studies have been carried out on new models of the ACO algorithm and their application to difficult optimization problems. Some of these algorithms are known as AS with elitist strategy (AS$^{\text{elit}}$), rank based version of AS (AS$^{\text{rank}}$), MAX-MIN AS and ant colony system (ACS) [7–10]. In most application areas, these algorithms are mainly used for optimization in discrete space [9–13]. In addition, different kinds of ant algorithms, such as continuous ant colony optimization (CACO), API, continuous interacting ant colony (CIAC) and touring ant colony optimization (TACO), have been introduced for optimization in the continuous field [14–17].

It is known that there is a premature convergence (stagnation) problem in the nature of ant algorithms [6]. Therefore, as the problem size grows, the ability of the algorithm to discover the optimum solution becomes weaker. On the other hand, when the problem size and number of parameters increase, parallel implementation of the algorithm could give more successful results [18,19]. Furthermore, ant colony optimization approaches are population based and they are naturally suited to parallel implementation [18–22]. So, these advantages lead us to consider a parallel version of the ant algorithm.

In this work a parallel ant colony optimization (PACO) algorithm based on the ant metaphor and the crossover operator of GAs is described. Our aim is to avoid premature convergence behaviour of the ant algorithm and to benefit from the advantages of a parallel structure. The performance of the proposed PACO algorithm is compared to that of the basic TS, parallel TS (PTS), GA and TACO algorithms for several well-known numerical test problems. Then, it is employed to train a recurrent neural network to identify linear and nonlinear dynamic plants. The second section of the chapter presents information about parallel ant colony algorithms in the literature. In the third section, the basic principles of TACO algorithms are introduced and the proposed model is described. Simulation results obtained from the test functions optimization and an application of PACO to training recurrent neural network are given in the fourth section. The work is concluded in the fifth section.

2 Parallel Ant Colony Algorithms

There are a few parallel implementations of ant algorithms in the literature. The first of these studies is that of Bolondi and Bondanza. They used fine-grained parallelism and assigned each ant to a single processor. Due to the high overhead for communication, this approach did not increase performance with an increased number of processors. Better results have been obtained with a more course-grained model [20,23].

Bullnheimer et al. propose two parallelization strategies, synchronous and partially asynchronous implementations of the ant system [18]. In simulations made on some TSP instances, it is shown that the synchronization and communication overhead slows down the performance. For this reason, the asynchronous parallel version outperforms the synchronous version as it is reduces the communication frequency.