Information Sciences 433–434 (2018) 142–162

Contents lists available at ScienceDirect

Information Sciences

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

A best-path-updating information-guided ant colony optimization algorithm

Jiaxu Ning[1], Qin Zhang[1], Changsheng Zhang*, Bin Zhang*

College of Computer Science & Engineering, Northeastern University, Shenyang 110819, PR China

ARTICLE INFO

Article history:
Received 17 May 2017
Revised 17 December 2017
Accepted 25 December 2017
Available online 26 December 2017

Keywords:
Ant colony optimization
Swarm intelligence
Pheromone update mechanism
Pheromone smoothing mechanism
Constraint satisfaction problem
Traveling salesman problem

ABSTRACT

The ant colony optimization (ACO) algorithm is a type of classical swarm intelligence algorithm that is especially suitable for combinatorial optimization problems. To further improve the convergence speed without affecting the solution quality, in this paper, a novel strengthened pheromone update mechanism is designed that strengthens the pheromone on the edges, which had never been done before, utilizing dynamic information to perform path optimization. In addition, to enhance the global search capability, a novel pheromone-smoothing mechanism is designed to reinitialize the pheromone matrix when the ACO algorithm's search process approaches a defined stagnation state. The improved algorithm is analyzed and tested on a set of benchmark test cases. The experimental results show that the improved ant colony optimization algorithm performs better than compared algorithms in terms of both the diversity of the solutions obtained and convergence speed.

1. Introduction

In the intelligent optimization field, many problems can be abstracted into path optimization problems, such as the constraint satisfaction problem (CSP) [26,27] or the traveling salesman problem (TSP) [39]. The ant colony optimization (ACO) algorithm, a classical bionic algorithm for determining the optimal path, has several advantages: it is easy to integrate with other algorithms, is amenable to distributed parallel computing, includes an intelligent search, and has good global optimization and strong robustness when compared with other swarm intelligence algorithms. Furthermore, the ant colony optimization algorithm is one of the representative incomplete algorithms that not only has the advantage of high speed and high accuracy but can also find a quasi-optimal solution quickly. The ant colony optimization algorithm may not have a significant advantage compared with other algorithms when solving simple problems, but on average, it shows a large efficiency when applied to comparatively complex problems. ACO's efficiency becomes quite important when we do not know the problem characteristics or what kind of algorithm to use in advance. Furthermore, the ACO reduces the costs of obtaining solutions to large-scale combinatorial optimization problems.

The ACO algorithm has been widely applied, from the TSP to problems such as data mining [16,30], telecommunications routing optimization [12,14], robot path planning [3], deep learning [21], and image processing [15,37].Compared with other meta-heuristic algorithms such as the genetic algorithm (GA) [35] and particle swarm optimization (PSO) [22], the ACO is not only easy to implement [13] but also highly suitable for such problems, because they are easily transformed into path

* Corresponding authors.
E-mail addresses: 1571053@stu.neu.edu.cn (Q. Zhang), zhangchangsheng@cse.neu.edu.cn (C. Zhang), zhangbin@cse.neu.edu.cn (B. Zhang).

[1] These authors contributed equally.

https://doi.org/10.1016/j.ins.2017.12.047
0020-0255/

optimization problems, and the optimization model can be directly built by the ACO's constructed graph. In contrast, when the GA or PSO are used to solve such problems, the optimization model is difficult because operations such as crossover, mutation, the position updating and velocity updating strategies and other components must all be redefined. Recently, many improved ant colony optimization algorithms have been proposed. Liu et al. [17] proposed a novel ant colony optimization algorithm with a dynamically weighted pheromone update mechanism (DWACA) that updates the pheromone dynamically and adaptively based on the pheromone concentration and the iterative optimal solution. Under a precondition that guarantees the quality of the solution, the efficiency of the algorithm is effectively improved. Qin et al. [24] presented an improved ant colony optimization algorithm with a dynamic local search (DLACA). The dynamic local search technique improves the quality of the constructed solution. The dynamic strategy is used to update the pheromone, and different paths use different pheromone update strategies that are proportional to both the time to find the optimal path and the optimal path length. Wang et al. [31] proposed an improved ant colony optimization algorithm (NCRM-ACO) to solve the problem of minimizing network coding resources. The algorithm uses a multidimensional pheromone mechanism to solve the problem of pheromone overlap. A local pheromone update mechanism is put forward to guide the ants to construct appropriate and potential paths. To avoid premature convergence, a method of solution reconstruction is presented that effectively improves the global optimization capability of the proposed algorithm. Goradia et al. [10] proposed an ant colony optimization algorithm (LMAS) that uses finite memory to preserve information concerning the optimal path. When ants explore a new edge and construct a new path, they will use their limited memories from the last iteration to guide the construction of the complete path. Bellaachia et al. [2] proposed an improved ACO algorithm with local pheromone initialization. In contrast to the traditional method, which initializes the pheromone on each edge directly using a constant or pre-calculated value, in the improved ACO, the ants initialize the edges with local information when they explore the edges for the first time. However, the algorithm does not initialize the pheromone on all edges. Thus, this local pheromone initialization technique can successfully achieve a balance between the quality of the solution and the time required to construct the solution. Yang et al. [34] presented a novel ant colony optimization algorithm (IACO) to solve mobile-agent-routing problems by introducing a genetic operator into the algorithm, and modifying the global pheromone update rule. This approach has some advantages in terms of robustness, parallelism, adaptation, positive feedback mechanisms, and so on. The experimental results showed that the IACO algorithm not only avoids local-minima but also converges faster than the GA, the simulated annealing algorithm, and the basic ACO. Seckiner et al. [25] proposed an ACO algorithm based on a novel pheromone updating mechanism. After each iteration, the pheromone update is performed based on the percentage of all ants that have searched for the optimal solution. The experimental results show that this ACO algorithm with the novel pheromone update mechanism has better performance in convergence and diversity than do other stochastic search algorithms. Deng et al. [4] proposed an ant colony optimization algorithm (PM-ACO) with a node-based pheromone updating strategy. In addition to the optimal path update rule, the algorithm includes two other update rules. One of them is based on node ordering, in which the nodes are sorted by their search times, and the pheromone of the highest-ranked nodes is updated. The other update rule is called the relevant node storage rule, and it updates the pheromone of the k nearest neighbor (KNN) nodes of a highly ranked node.

However, in the existing literature, some aspects of ant colony optimization algorithms could be further optimized. First, compared with other intelligent search algorithms, although the ACO algorithm globally searches the high-quality solution set, it requires a longer search time. Second, the optimizing processes of the existing ant colony algorithms easily fall into some local optimal state. This occurs mainly because excessive use of the positive feedback mechanism [1] that guides a large number of ants to the high pheromone path leads to a decreasing probability of other pheromone edges being selected. To further enhance the performance of the ant colony optimization algorithm, this paper proposes some corresponding improvement mechanisms.

On one hand, this paper proposes a novel strengthened pheromone updating mechanism based on information from the current best path. The ACO algorithm generates the former iterative optimal path. Then, the best iterative path up to now is compared with it to determine the dynamic best path-varying information whose edges are newly explored. The dynamically varying information is determined, and the pheromone is strengthened. Furthermore, the dynamic change information can narrow the search range of the algorithm and improve the search efficiency. Particularly when the ACO algorithm is applied to large-scale combinatorial optimization problems, the proposed approach enhances the convergence speed without affecting the solution quality.

On the other hand, when most ants explore along the path of the convergence, the search will easily hover around the local optimal state, which causes search stagnation. To improve the limited search efforts and the ACO's exploration ability, a novel pheromone smoothing mechanism is presented that reinitializes the pheromone matrix. First, it determines whether the algorithm is about to enter a defined stagnation state; if so, the novel pheromone smoothing mechanism reinitializes the pheromone matrix to balance the difference of the pheromone on every edge. This approach enhances the search capabilities of the algorithm in subsequent iterations and improves its global optimization ability. Moreover, it also improves the solution quality of the algorithm to some extent.

This remainder of this paper is organized as follows. Section 2 introduces the basic steps of the MAX-MIN Ant System [29,38]. Section 3 describes the improved ACO algorithm with the hybrid strengthened pheromone updating and pheromone smoothing mechanisms. Section 4 reports experimental results on CSP and TSP benchmark test cases along with those of other compared algorithms. Finally, conclusions are drawn, and future work is discussed in Section 5.