Expert Systems with Applications 41 (2014) 2535–2545

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Improved Shuffled Frog Leaping Algorithm and its multi-phase model for multi-depot vehicle routing problem

Jianping Luo *, Min-Rong Chen

College of Information Engineering, Shenzhen University, Shenzhen 518060, China

ARTICLE INFO

Keywords:
Evolutionary computation
Vehicle routing problem
Shuffled Frog Leaping Algorithm
Extremal optimization

ABSTRACT

In the present work, an improved Shuffled Frog Leaping Algorithm (SFLA) and its multi-phase model are presented to solve the multi-depots vehicle routing problems (MDVRPs). To further improve the local search ability of SFLA and speed up convergence, a Power Law Extremal Optimization Neighborhood Search (PLEONS) is introduced to SFLA. In the multi-phase model, firstly the proposed algorithm generates some clusters randomly to perform the clustering analyses considering the depots as the centroids of the clusters for all the customers of MDVRP. Afterward, it implements the local depth search using the SFLA for every cluster, and then globally re-adjusts the solutions, i.e., rectifies the positions of all frogs by PLEONS. In the next step, a new clustering analyses is performed to generate new clusters according to the best solution achieved by the preceding process. The improved path information is inherited to the new clusters, and the local search using SFLA for every cluster is used again. The processes continue until the convergence criterions are satisfied. The experiment results show that the proposed algorithm possesses outstanding performance to solve the MDVRP and the MDVRP with time windows.

1. Introduction

The vehicle routing problem (VRP), first introduced by Dantzig and Ramser (1959), has been studied extensively because of its widespread applications in many real-world situations. Related to the logistics distribution problem, the VRP is a very important problem in research because it plays an important role in satisfying customer demands in an exact and timely manner. The multi-depot vehicle routing problem (MDVRP) is also an NP-hard problem for simultaneously determining the routes for several vehicles from more than one depot to a set of customers, and then returning to the same depot without exceeding the capacity constraint of each vehicle. In practice, the problem is aimed at minimizing the total cost of the combined routes for a fleet of vehicles. Since cost is closely associated with distance, in general, the goal is to minimize the distance traveled by a fleet of vehicles with various constraints.

The classical VRP has been focused on strongly in the literature over the past 40 years. Comparatively, the number of research projects on the MDVRP is fewer. To solve the MDVRP, Wren and Holliday (1972) described a heuristic consisting of two parts: constructing an initial solution, followed by a method of saving in each depot and refinements. Chao, Golden, and Wasil (1993) used a simple initialization heuristic combined with an improvement phase that is more powerful than the earlier studies. Nagy and Salhi (2005) presented a number of heuristic methods to solve the single-depot VRP with pickup and delivery (VRPPD). The methods can be modified to tackle the multi-depot VRPPD (MDVRPPD). Tabu search heuristics for the MDVRP have been proposed by Renaud, Laporte, and Boctor (1996) and Cordeau, Gendreau, and Laporte (1997). Pisinger and Ropke (2007) presented a unified heuristic that could solve different variants of the VRP. Crevier, Cordeau, and Laporte (2007) addressed an extension of the MDVRP where vehicles may be replenished at intermediate depots along their routes. The authors designed and compared six heuristics for assigning customers to depots while using the same VRP heuristic for each depot. Farhang and Seyedhosseini (2010) proposed a particle swarm optimization (PSO) to solve this kind of problem. Ho, Ho, Ji, and Lau (2008) developed two hybrid genetic algorithms (HGAs) for MDVRP. The major difference between the HGAs was the initial solutions. Ghoseiri and Ghannadpour (2010) presented an HGA to solve a multi-depot homogenous locomotive assignment problem with time windows. In the locomotive assignment problem, a set of homogeneous locomotives locating in a set of dispersed depots has to be assigned to a set of pre-scheduled trains that are supposed to be serviced in pre-specified hard/soft time windows. Liu, Jiang, Fung, Chen and Liu (2010) proposed a two-phase greedy algorithm to solve practical large-scale MDVRPs. In the first phase, a set of directed cycles was created to fulfill the transportation orders. In the second phase, chains composed of cycles were generated. Mirabi, Fatemi Ghomi, and Jolai (2010) addressed the MDVRP to minimize

* Corresponding author.
E-mail address: camelrock@126.com (J. Luo).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.10.001

the delivery time of vehicle objective. Three hybrid heuristics were presented to solve the MDVRP. Each hybrid heuristic combined elements from constructive heuristic search and improvement techniques. The improvement techniques include deterministic, stochastic, and simulated annealing (SA) methods.

Due to the complexity of the problem, solving the MDVRP to optimality is extremely time-consuming. To tackle the problem efficiently, all previous researchers preferred heuristic methods to exact algorithms. According to the above literature review, there are two common steps among these proposed methodologies. First, the MDVRP is decomposed, and then the sub-problems are solved sequentially and iteratively. Second, the heuristic methods consist of two mechanisms: construction and improvement. The first mechanism generates initial feasible solutions, whereas the second mechanism modifies the existing solutions to yield better results. However, the relations between the two steps are lacking, i.e., the results of construction and improvement can hardly respond to the decomposition process.

Evolutionary computing, which mimics the metaphor of natural biological evolution and/or the social behavior of species, is an exciting development in computer science. Memetic algorithm (MA) is a special class of heuristic search method derived from the models of adaptation in natural systems, which combine the evolutionary adaptation of a population with individual learning within the lifetimes of their members. The term "memetic algorithm" comes from "meme". Meme (consisting of memetypes) is a contagious information pattern that replicates by parasitically infecting human and/or animal minds and altering their behavior, which causes them to propagate the pattern. MA is based on the evolution of memes carried by interactive individuals. Its notable characteristic is that all memes are allowed to gain some experience through a local search before being involved in the evolutionary process (see Merz & Freisleben, 1997).

Shuffled Frog Leaping Algorithm (SFLA), which was developed by Eusuff and Lansey (2003), belongs to the MA family. It is a meta-heuristic optimization method inspired from the memetic evolution of a group of frogs when seeking for food. In this algorithm, the evolution of memes is driven by the exchange of information among interactive individuals. SFLA combines the advantages of the genetic-based MA and the social behavior-based PSO algorithm (see Kennedy & Eberhart, 1995). It has been tested on several combinatorial problems and found to be effective in searching the global solutions (e.g., Alireza, Mostafa, Hamed, & Ehsan, 2009; Babak, Mohammad, & Ali, 2009; Bhaduri 2009; Li, Luo, Chen, & Wang, 2010; Luo, Li, & Chen, 2011; Park & Joyoung, 2009).

Recently, a new general-purpose local search optimization approach, Extremal Optimization (EO), has been proposed by Boettcher and Percus (1999) based on the fundamentals of statistical physics and self-organized criticality (see Bak & Sneppen, 1993). The evolution in this method is driven by a process where the weakest species in the population, together with its nearest neighbors, is always forced to mutate. EO successively eliminates those worst components in the sub-optimal solutions, and has been successfully applied to many continuous and discrete optimization problems (e.g., Boettcher, 2005; Chen & Lu, 2008).

In this paper, an improved multi-phase SFLA based on clustering is presented to solve the MDVRP and the MDVRP with time windows (MDVRPTW). To further improve the local search ability of SFLA, a power-law Extremal Optimization neighborhood search, called PLEONS, is introduced to SFLA.

This paper is organized as follows: The SFLA and the hybrid algorithm, which merges PLEONS into SFLA (SFLA-PLEONS), is proposed to solve the MDVRP in Section 2. The multi-phase SFLA-PLEONS algorithm is introduced in Section 3. Experimental evaluations and result discussions are shown in Section 4. Finally, the conclusions are drawn.

2. SFLA with PLEONS for MDVRP

2.1. Shuffled Frog Leaping Algorithm (SFLA)

SFLA is a meta-heuristic optimization method that mimics the memetic evolution of a group of frogs when seeking for the location that has the maximum amount of available food. It is described in detail as follows. First, an initial population of F frogs is created randomly. For the d-dimensional problem, the position of the 'ith' frog is represented as $x_i = (x_{i1}, x_{i2,} \ldots, x_{id})$. Afterwards, the frogs are sorted in a descending order according to their fitness. The F frogs are then separated into m memeplexes according to their fitness ordering, each containing n frogs (i.e., $F = m \times n$), in such a way that the first frog goes to the first memeplex, the second frog goes to the second memeplex, the mth frog goes to the mth memeplex, the $(m + 1)$th frog goes back to the first memeplex, and so forth. Next, for each memeplex, a sub-memeplex including q frogs is constructed. It is reasonable to assume that the better the frog's position, the higher the probability of the frog to be selected in the sub-memeplex will be. Higher probability generated by a triangle probability distribution expressed in (1) is assigned to the frog with higher performance, according to which q frogs are selected out:

$$P_j = 2(n + 1 - j)/(n(n + 1)), \tag{1}$$

where j is the fitness sorting number of the current frog in the population according to the fitness value in descending order.

The main work of SFLA is to update the position of the worst-performing frog through iterative operation in each sub-memeplex. Its position is improved by learning from the best frog of the sub-memeplex or its own population and position. In each sub-memeplex, the new position of the worst frog is updated according to Eqs. (2) and (3):

$$D = r \cdot (x_s - x_w(k)), \tag{2}$$

$$x_w(k + 1) = x_w(k) + D, \quad \|D_{\min} \leqslant \|D\| \leqslant D_{\max}\|, \tag{3}$$

where $x_w(k)$ and x_s are the worst frog position and the best frog position, respectively, in the sub-memeplex; r is a random number in range[0,1]; k is the iteration number of the sub-memeplex; and $\|D_{\min}\|$ and $||D_{\max}||$ are the maximum and minimum allowed change in a frog's position, respectively. If the new position of the worst frog is better than before, it replaces the worst frog's position; otherwise, the calculations in Eqs. (2) and (3) are repeated but with respect to the global best frog (i.e., x_b replaces x_s). If this process still cannot obtain the better performance, the position is randomly generated for the worst frog. The iteration continues for a predefined number of memetic evolutionary times within each memeplex, and then the whole population is mixed together in the shuffling process. The local evolution and global shuffling continue until convergence criterions are satisfied.

2.2. Evaluation of individual

The individual evaluation function in population-based meta-heuristics aims to determine for each individual a relative value with respect to the entire population. But this so-called fitness evolution measure is generally myopic with respect to the possible impact of the evaluation and selection processes on the diversity of the population, and it is critical performance factor for this class of meta-heuristics such as SFLA. In this paper, we define the evaluation function accounting for the cost of an individual and its contribution to the population diversity.