Computers & Operations Research 64 (2015) 11–27

Contents lists available at ScienceDirect

Computers & Operations Research

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

A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows

Çağrı Koç [a], Tolga Bektaş [a], Ola Jabali [b,*], Gilbert Laporte [c]

[a] CORMSIS and Southampton Business School, University of Southampton, Southampton SO17 1BJ, United Kingdom
[b] CIRRELT and HEC Montréal, 3000, chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7
[c] CIRRELT and Canada Research Chair in Distribution Management, HEC Montréal, 3000, chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7

ARTICLE INFO

Available online 18 May 2015

Keywords:
Vehicle routing
Time windows
Heterogeneous fleet
Genetic algorithm
Neighborhood search

ABSTRACT

This paper presents a hybrid evolutionary algorithm (HEA) to solve heterogeneous fleet vehicle routing problems with time windows. There are two main types of such problems, namely the fleet size and mix vehicle routing problem with time windows (F) and the heterogeneous fixed fleet vehicle routing problem with time windows (H), where the latter, in contrast to the former, assumes a limited availability of vehicles. The main objective is to minimize the fixed vehicle cost and the distribution cost, where the latter can be defined with respect to en-route time (T) or distance (D). The proposed unified algorithm is able to solve the four variants of heterogeneous fleet routing problem, called FT, FD, HT and HD, where the last variant is new. The HEA successfully combines several metaheuristics and offers a number of new advanced efficient procedures tailored to handle the heterogeneous fleet dimension. Extensive computational experiments on benchmark instances have shown that the HEA is highly effective on FT, FD and HT. In particular, out of the 360 instances we obtained 75 new best solutions and matched 102 within reasonable computational times. New benchmark results on HD are also presented.

1. Introduction

In heterogeneous fleet vehicle routing problems with time windows, one considers a fleet of vehicles with various capacities and vehicle-related costs, as well as a set of customers with known demands and time windows. These problems consist of determining a set of vehicle routes such that each customer is visited exactly once by a vehicle within a prespecified time window, all vehicles start and end their routes at a depot, and the load of each vehicle does not exceed its capacity. As is normally the case in vehicle routing problem with time windows (VRPTW), customer service must start within the time window, but the vehicle may wait at a customer location if it arrives before the beginning of the time window. There are two main categories of such problems, namely the fleet size and mix vehicle routing problem with time windows (F) and the heterogeneous fixed fleet vehicle routing problem with time windows (H). In category F, there is no limit in the number of available vehicles of each type, whereas such a limit exists in category H. Note that it is easy to find feasible solutions to the instances of category F since there always exists a feasible assignment of vehicles to routes. However, this is not always the case for the instances of category H.

Two measures are used to compute the total cost to be minimized. The first is the sum of the fixed vehicle cost and of the *en-route time* (T), which includes traveling time and possible waiting time at the customer locations before the opening of their time windows (we assume that travel time and cost are equivalent). In this case, service times are only used to check feasibility and for performing adjustments to the departure time from the depot in order to minimize pre-service waiting times. The second cost measure is based on *distance* (D) and consists of the fixed vehicle cost and the distance traveled by the vehicle, as is the case in the standard VRPTW [30].

We differentiate between four variants defined with respect to the problem category and to the way in which the objective function is defined, namely FT, FD, HT and HD. The first variant is FT, described by Liu and Shen [20] and the second is FD, introduced by Braysy et al. [7]. The third variant HT was defined and solved by Paraskevopoulos et al. [22]. Finally, HD is a new variant which we introduce in this paper. HD differs from HT by considering the objective function D instead of T. This variant has never been studied before.

Hoff et al. [16] and Belfiore and Yoshizaki [4] describe several industrial aspects and practical applications of heterogeneous vehicle routing problems. The most studied versions are the fleet

* Corresponding author. Tel.: +1 514 340 6154.
E-mail addresses: C.Koc@soton.ac.uk (Ç. Koç), T.Bektas@soton.ac.uk (T. Bektaş), ola.jabali@hec.ca (O. Jabali), Gilbert.Laporte@cirrelt.ca (G. Laporte).

http://dx.doi.org/10.1016/j.cor.2015.05.004
0305-0548/

size and mix vehicle routing problem, described by Golden et al. [15], which considers an unlimited heterogeneous fleet, and the heterogeneous fixed fleet vehicle routing problem, proposed by Taillard [31]. For further details, the reader is referred to the surveys of Baldacci et al. [1] and of Baldacci and Mingozzi [2].

The FT variant has several extensions, e.g., multiple depots [13,6], overloads [17], and split deliveries [4,5]. There exist several exact algorithms for the capacitated vehicle routing problem (VRP) [32,3], and for the heterogeneous VRP [2]. However, to the best of our knowledge, no exact algorithm has been proposed for the heterogeneous VRP with time windows, i.e., FT, FD and HT. The existing heuristic algorithms for these three variants are briefly described below.

Liu and Shen [20] proposed a heuristic for FT which starts by determining an initial solution through an adaptation of the Clarke and Wright [9] savings algorithm, previously presented by Golden et al. [15]. The second stage improves the initial solution by moving customers by means of parallel insertions. The algorithm was tested on a set of 168 benchmark instances derived from the set of Solomon [30] for the VRPTW. Dullaert et al. [14] described a sequential construction algorithm for FT, which is an extension of the insertion heuristic of Golden et al. [15]. Dell'Amico et al. [11] described a multi-start parallel regret construction heuristic for FT, which is embedded into a ruin and recreate metaheuristic. Bräysy et al. [7] presented a deterministic annealing metaheuristic for FT and FD. In a later study, Bräysy et al. [8] described a hybrid metaheuristic algorithm for large scale FD instances. Their algorithm combines the well-known threshold acceptance heuristic with a guided local search metaheuristic having several search limitation strategies. An adaptive memory programming algorithm was proposed by Repoussis and Tarantilis [26] for FT, which combines a probabilistic semi-parallel construction heuristic, a reconstruction mechanism and a tabu search algorithm. Computational results indicate that their method is highly successful and improves many best known solutions. In a recent study, Vidal et al. [35] introduced a genetic algorithm based on a unified solution framework for different variants of the VRPs, including FT and FD. To our knowledge, Paraskevopoulos et al. [22] are the only authors who have studied HT. Their two-phase solution methodology is based on a hybridized tabu search algorithm capable of solving both FT and HT.

This brief review shows that the two problem categories F and H have already been solved independently through different methodologies. We believe there exists merit for the development of a unified algorithm capable of efficiently solving the two problem categories. This is the main motivation behind this paper.

This paper makes three main scientific contributions. First, we develop a unified hybrid evolutionary algorithm (HEA) capable of handling the four variants of the problem. The HEA combines two state-of-the-art metaheuristic concepts which have proved highly successful on a variety of VRPs: adaptive large neighborhood search (ALNS) (see [27,23,12]) and population based search (see [24,35]). The second contribution is the introduction of several algorithmic improvements to the procedures developed by Prins [25] and Vidal et al. [33]. We use a ALNS equipped with a range of operators as the main Education procedure within the search. We also propose an advanced version of the Split algorithm of Prins [25] capable of handling infeasibilities. Finally, we introduce an innovative aggressive Intensification procedure on elite solutions, as well as a new diversification scheme through the Regeneration and the Mutation procedures of solutions. The third contribution is to introduce HD as a new problem variant.

The remainder of this paper is structured as follows. Section 2 presents a detailed description of the HEA. Computational experiments are presented in Section 3, and conclusions are provided in Section 4.

2. Description of the hybrid evolutionary algorithm

We start by introducing the notation related to FT, FD, HT and HD. All problems are defined on a complete graph $G=(N,A)$, where $N=\{0,\dots,n\}$ is the set of nodes, and node 0 corresponds to the depot. Let $A=\{(i,j): 0 \le i,j\} \le n, i \ne j\}$ denote the set of arcs. The distance from i to j is denoted by d_{ij}. The customer set is N_c in which each customer i has a demand q_i and a service time s_i, which must start within time window $[a_i, b_i]$. If a vehicle arrives at customer i before a_i, it then waits until a_i. Let $K=\{1,\dots,k\}$ be the set of available vehicle types. Let e_k and Q_k denote the fixed vehicle cost and the capacity of vehicle type k, respectively. The travel time from i to j is denoted by t_{ij} and is independent of the vehicle type. The distribution cost from i to j associated with a vehicle of type k is c_{ij}^k for all problem types. In HT and HD, the available number of vehicles of type $k \in K$ is n_k, whereas the constant can be set to an arbitrary large value for problems FT and FD. The objectives are as discussed in the Introduction.

The remainder of this section introduces the main components of the HEA. A general overview of the HEA is given in Section 2.1. More specifically, Section 2.2 presents the offspring Education procedure. Section 2.3 presents the initialization of the population. The selection of parent solutions, the ordered crossover operator and the advanced algorithm Split are described in Sections 2.4, 2.5 and 2.6, respectively. Section 2.7 presents the Sntensification procedure. The survivor selection mechanism is detailed in Section 2.8. Finally, the diversification stage, including the Regeneration and Mutation procedures, is described in Section 2.9.

2.1. Overview of the hybrid evolutionary algorithm

The general structure of the HEA is presented in Algorithm 1. The modified version of the classical Clarke and Wright savings algorithm and the ALNS operators are combined to generate the initial population (Line 1). Two parents are selected (Line 3) through a binary tournament, following which the crossover operation (Line 4) generates a new offspring C. The advanced Split algorithm is applied to the offspring C (Line 5), which optimally segments the giant tour by choosing the vehicle type for each route. The Education procedure (Line 6) uses the ALNS operators to educate offspring C and inserts it back into the population. If C is infeasible, the Education procedure is iteratively applied until a modified version of C is feasible, which is then inserted into the population.

The probabilities associated with the operators used in the Education procedure and the penalty parameters are updated by means of an adaptive weight adjustment procedure (AWAP) (Line 7). Elite solutions are put through an aggressive Intensification procedure, based on the ALNS algorithm (Line 8) in order to improve their quality. If, at any iteration, the population size n_a reaches n_p+n_o, then a survivor selection mechanism is applied (Line 9). The population size, shown by n_a, changes during the algorithm as new offsprings are added, but is limited by n_p+n_o, where n_p is a constant denoting the size of the population initialized at the beginning of the algorithm and n_o is a constant showing the maximum allowable number of offsprings that can be inserted into the population. At each iteration of the algorithm, Mutation is applied to a randomly selected individual from the population with probability p_m. If there are no improvements in the best known solution for a number of consecutive iterations it_r, the entire population undergoes a Regeneration (Line 10). The HEA