Computers & Operations Research 53 (2015) 234–249

Contents lists available at ScienceDirect

Computers & Operations Research

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

An iterated local search heuristic for the split delivery vehicle routing problem

Marcos Melo Silva [a,*], Anand Subramanian [b], Luiz Satoru Ochi [a]

[a] Universidade Federal Fluminense, Instituto de Computação, Rua Passo da Pátria 156, Bloco E – 3° andar, São Domingos, Niterói-RJ 24210-240, Brazil
[b] Universidade Federal da Paraíba, Departamento de Engenharia de Produção, Centro de Tecnologia, Campus I – Bloco G, Cidade Universitária, João Pessoa-PB 58051-970, Brazil

ARTICLE INFO

Available online 19 August 2014

Keywords:
Metaheuristics
Vehicle routing
Split deliveries
Iterated local search

ABSTRACT

This paper concerns the Split Delivery Vehicle Routing Problem (SDVRP). This problem is a relaxation of the Capacitated Vehicle Routing Problem (CVRP) since the customers' demands are allowed to be split. We deal with the cases where the fleet is unlimited (SDVRP-UF) and limited (SDVRP-LF). In order to solve them, we implemented a multi-start Iterated Local Search (ILS) based heuristic that includes a novel perturbation mechanism. Extensive computational experiments were carried out on benchmark instances available in the literature. The results obtained are highly competitive, more precisely, 55 best known solutions were equaled and new improved solutions were found for 243 out of 324 instances, with an average and maximum improvement of 1.15% and 2.81%, respectively.

1. Introduction

The Split Delivery Vehicle Routing Problem (SDVRP) is a relaxation of the well-known Capacitated Vehicle Routing Problem (CVRP) where the customers' demands are allowed to be split. The SDVRP can be formally defined as follows. Let $G=(V,E)$ be an undirected graph, where $V=\{0,\dots,n\}$ is the set of vertices and $E=\{\{i,j\}: i,j\in V, i<j\}$ is the set of edges. Vertex 0 is the depot where a fleet of K identical vehicles with capacity Q are placed, while other vertices, $V'=\{1,\dots,n\}$, represent customers, each one with a positive demand d_i ($i\in V; d_0=0$). A non-negative travel cost c_{ij} is associated with every edge $\{i,j\}\in E$. The objective is to minimize the sum of the travel costs in such a way that (i) the vehicle capacity is not exceeded; (ii) the demand $d_i, i\in V'$, which can be less than, equal, or greater than Q, is either delivered entirely by a single vehicle or split among different vehicles (therefore customers are allowed to be visited more than once); (iii) every route starts and ends at the depot.

A number of real-life applications of the SDVRP are available in the literature. Sierksma and Tijssen [32] described a practical case of specifying a flight schedule for helicopters to off-shore platforms, located in the Dutch continental shelf of the North Sea, for transporting workers to these platforms. Song et al. [34] dealt with the problem of determining an optimal assignment of newspaper agents as well as optimal routes and schedules for newspaper delivery in South Korea. Ambrosino and Sciomachen [4] studied the food distribution problem of an Italian company by considering a split delivery scheme. Belfiore and Yoshizaki [12] presented a case study in a large retail market in Brazil where the authors were faced with the SDVRP with additional features such as time windows, heterogeneous fleet and site dependence. All these practical problems were solved using heuristic approaches.

In spite of being a relaxation of the CVRP, the SDVRP was proven to be $\mathcal{NP}$-hard by Dror and Trudeau [21]. Archetti et al. [7] showed that the SDVRP can be solved in polynomial time when the capacity of the vehicle is 2. Furthermore, there are two main versions of the SDVRP studied in the literature: the one with limited fleet (SDVRP-LF) and the one with unlimited fleet (SDVRP-UF). With respect to the first, the minimum possible number of vehicles ($K_{min}=\lceil\sum_{i=1}^{n} d_i/Q\rceil$) must be used. It can be verified that there always exists a feasible solution when using $K_{\min}$ vehicles [21]. Archetti et al. [6] studied the complexity of these versions when the graph has special structures, namely line, tree, star and circle. Regarding the SDVRP-LF, the authors showed that the problem has complexity $\mathcal{O}(n^2)$ when the graph is a circle, $\mathcal{O}(n)$ when the graph is a line and $\mathcal{NP}$-hard for the remaining cases. Moreover, they also proved that the SDVRP-UF is $\mathcal{NP}$-hard for a tree, $\mathcal{O}(n^2)$ for a circle and $\mathcal{O}(n)$ for the other graphs.

The objective of this work is to present an Iterated Local Search (ILS) based heuristic for both the SDVRP-LF and SDVRP-UF. This metaheuristic proved to be highly efficient when applied to a variety of important routing problems as can be seen in the recent

* Corresponding author. Tel.: +55 21 2629 5665; fax: +55 21 2629 5666.
E-mail addresses: mmsilva@ic.uff.br (M.M. Silva), anand@ct.ufpb.br (A. Subramanian), satoru@ic.uff.br (L.S. Ochi).

http://dx.doi.org/10.1016/j.cor.2014.08.005
0305-0548/

works of Subramanian et al. [36], Penna et al. [30], Silva et al. [33] and Subramanian and Battarra [35]. The proposed algorithm differs from these previous works by incorporating efficient procedures especially devoted to the SDVRP such as (i) specific local search operators to enhance the intensification phase; (ii) a new perturbation scheme to improve the diversification phase; and (iii) auxiliary procedures that are useful in several parts of the algorithm such as when repairing a given solution or when trying to empty a route. These three modifications together play a remarkable role in the performance of the algorithm, resulting in a number of new improved solutions. Furthermore, when compared to other heuristic approaches presented in the literature, the algorithm developed here relies on very few parameters, thus requiring less tuning efforts. Extensive computational experiments were carried out on benchmark instances available in the literature. The results obtained are highly competitive, more precisely, 55 best known solutions were equaled and new improved solutions were found for 243 out of 324 instances, with an average and maximum improvement of 1.15% and 2.81%, respectively.

The remainder of the paper is organized as follows. Section 2 presents some related works. Section 3 describes the proposed ILS algorithm. Section 4 contains the computational results. Finally, Section 5 concludes the work.

2. Related work

The SDVRP was introduced by Dror and Trudeau [20,21] where the authors demonstrated empirically that split deliveries can generate savings both in terms of number of vehicles and travel costs. This claim was verified through computational experiments involving a set of 540 instances with up to 150 customers.

Some exact approaches for the SDVRP were suggested in the literature. Dror et al. [19] presented a mixed integer programming (MIP) formulation for the SDVRP-UF along with new classes of valid inequalities that were incorporated in a cutting-plane algorithm. Belenguer et al. [11] performed a polyhedral study on the SDVRP-LF and, as a result, several classes of valid inequalities were developed and a cutting-plane based algorithm was implemented. Jin et al. [26] proposed a two-phase exact algorithm for the SDVRP-LF, while Jin et al. [27] developed a column generation algorithm for the same variant. Moreno et al. [28] put forward a cut-and-price based approach over an extended formulation together with new valid inequalities for the SDVRP-LF. More recently, Archetti et al. [5] devised a branch-cut-and-price algorithm for the SDVRP-LF and SDVRP-UF.

In spite of the efforts in developing exact solution approaches for the SDVRP, such methods are still only capable of solving instances with up to 30 customers in a systematic fashion. This number is limited when compared to the CVRP where there are exact algorithms capable of systematically solving instances with up to 135 customers. Heuristic algorithms, however, appear to be more suitable for dealing with medium-large SDVRP instances.

Archetti et al. [9] proposed a three-phase Tabu Search (TS) heuristic for the SDVRP-UF. Chen et al. [15] developed a hybrid algorithm for the SDVRP-UF by combining a savings based heuristic with a mixed integer programming formulation and a record-to-record procedure. Campos et al. [14] proposed a Scatter Search based algorithm for the SDVRP-LF with two different procedures for generating initial populations. Boudia et al. [13] developed a Memetic Algorithm with population management for the SDVRP-UF and three new neighborhood structures especially devoted to the SDVRP. Archetti et al. [10] put forward a hybrid approach that combines a TS heuristic with an integer programming formulation. Aleman et al. [2,3] presented constructive and local search procedures for the SDVRP-LF, whereas Aleman and Hill [1] implemented a TS heuristic with vocabulary building for the SDVRP-UF. Derigs et al. [17] presented a study where they compared the behavior of several local search based metaheuristics for the SDVRP-UF. More recently, Wilck IV and Cavalier [37] developed a two-phase constructive procedure for the SDVRP-LF. They also proposed a Genetic Algorithm [38] for the same variant.

A detailed and comprehensive survey of SDVRPs can be found in the recent work of Archetti and Speranza [8].

3. Description of the iterated local search heuristic

The proposed algorithm, called SplitILS, consists of a multi-start heuristic mainly based on ILS. Initial solutions are generated using an extension of the cheapest insertion procedure, while local search is performed using Randomized Variable Neighborhood Descent (RVND) [25,36]. Every time a solution gets trapped in a local optimum, i.e., none of the local search operators is capable of improving it, a perturbation mechanism is applied over this solution.

Algorithm 1 presents the pseudocode of SplitILS. The algorithm executes I_{Max} iterations (lines 3–18) and at each of them an initial solution is generated using an insertion-based constructive procedure (line 4). The initial solution is improved by applying a RVND procedure in the local search phase (lines 7–15). The solutions initially generated always contain the minimum number of vehicles (K_{min}). However, during the local search, the number of routes can increase due to a neighborhood operator called *RouteAddition* (see Section 3.3.2) designed to bring together parts of customers' demands that were previously split. The number of routes associated to the solution returned at the end of the local search may need to be adjusted or not depending on the version of the problem, i.e., SDVRP-LF or SDVRP-UF. If the fleet is limited (SDVRP-LF) and the current solution contains more routes than the minimum number of vehicles, the least-loaded routes are emptied using a procedure called *EmptyRoutes* (see Section 3.1) (lines 9–10). Otherwise (SDVRP-UF), the solution is not modified. If an improved solution is found, *iterILS* is reinitialized (lines 11–13). The best current solution is then perturbed (line 14) using the perturbation mechanism described in Section 3.4.

Algorithm 1. SplitILS.

```
1   Procedure SplitILS(I_Max, I_ILS)
2   f* ← ∞;
3   for i ← 1, …, I_Max do
4   |  s ← Construction();
5   |  s' ← s;
6   |  iterILS ← 0;
7   |  while iterILS < I_ILS do
8   |  |  s ← RVND(s);
9   |  |  if SDVRP − LF then
10  |  |  |  s ← EmptyRoutes(s);
11  |  |  if f(s) < f(s') then
12  |  |  |  s' ← s;
13  |  |  |  iterILS ← 0;
14  |  |  s ← Perturb(s');
15  |  |  iterILS ← iterILS + 1;
16  |  |
17  |  if f(s') < f* then
18  |  |  s* ← s';
    |  |  f* ← f(s');
19  return s*;
```

3.1. Auxiliary procedures

In this section we describe two important auxiliary procedures that are used in other parts of the algorithm. The first one is called