Computers & Operations Research 37 (2010) 2106–2120

Contents lists available at ScienceDirect

Computers & Operations Research

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

An ILP improvement procedure for the Open Vehicle Routing Problem

Majid Salari, Paolo Toth *, Andrea Tramontani

DEIS, University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy

ARTICLE INFO

Available online 6 March 2010

Keywords:
Integer Linear Programming
Local search
Heuristics
Open Vehicle Routing Problem

ABSTRACT

We address the Open Vehicle Routing Problem (OVRP), a variant of the "classical" (capacitated and distance constrained) Vehicle Routing Problem (VRP) in which the vehicles are not required to return to the depot after completing their service. We present a heuristic improvement procedure for OVRP based on Integer Linear Programming (ILP) techniques. Given an initial feasible solution to be possibly improved, the method follows a destruct-and-repair paradigm, where the given solution is randomly destroyed (i.e., customers are removed in a random way) and repaired by solving an ILP model, in the attempt of finding a new improved feasible solution. The overall procedure can be considered as a general framework which could be extended to cover other variants of Vehicle Routing Problems. We report computational results on benchmark instances from the literature. In several cases, the proposed algorithm is able to find the new best known solution for the considered instances.

1. Introduction

We address the Open Vehicle Routing Problem (OVRP), a variant of the "classical" (capacitated and distance constrained) Vehicle Routing Problem (VRP) in which the vehicles are not required to return to the depot after completing their service. OVRP can be formally stated as follows. We are given a central *depot* and a set of n *customers*, which are associated with the nodes of a complete undirected graph $G=(V,E)$ (where $V=\{0,1,\ldots,n\}$, node 0 represents the depot and $V\backslash\{0\}$ is the set of customers). Each edge $e \in E$ has an associated finite *cost* $c_e \geq 0$ and each customer $v \in V\backslash\{0\}$ has a *demand* $q_v > 0$ (with $q_0=0$). A fleet of m identical *vehicles* is located at the depot, each one with a *fixed cost* F, a *capacity* Q and a *total distance-traveled* (*duration*) limit D. The customers must be served by at most m Hamiltonian paths (*open routes*), each path associated with one vehicle, starting at the depot and ending at one of the customers. Each route must have a duration (computed as the sum of the edge costs in the route) not exceeding the given limit D of the vehicles, and can visit a subset S of customers whose total demand $\sum_{v \in S} q_v$ does not exceed the given capacity Q. The problem consists of finding a feasible solution covering (i.e., visiting) exactly once all the customers and having a minimum overall cost, computed as the sum of the traveled edge costs plus the fixed costs associated with the vehicles used to serve the customers. OVRP is known to be $\mathcal{NP}$-hard in the strong sense, as it generalizes the Bin Packing Problem and the Hamiltonian Path Problem.

In this paper we present a heuristic improvement procedure for OVRP based on Integer Linear Programming (ILP) techniques. Given an initial feasible solution to be possibly improved, the procedure iteratively performs the following steps: (a) randomly select several customers from the current solution, and build the restricted solution obtained from the current one by extracting (i.e., short-cutting) the selected customers; (b) reallocate the extracted customers to the restricted solution by solving an ILP problem, in the attempt of finding a new improved feasible solution. This method has been proposed by De Franceschi et al. [7] and deeply investigated by Toth and Tramontani [27] in the context of the classical VRP. Here, we consider a simpler version of this approach, which exploits no particular feature of the addressed problem. The method follows a destruct-and-repair paradigm, where the current solution is randomly destroyed (i.e., customers are removed in a random way) and repaired by following ILP techniques. Hence, the overall procedure can be considered as a general framework which could be extended to cover other variants of Vehicle Routing Problems.

The notion of using ILP techniques to improve a feasible solution of a combinatorial optimization problem has emerged in several papers in the last few years. Addressing the split delivery VRP, Archetti et al. [2] developed a heuristic algorithm that integrates tabu search with ILP by solving integer programs to explore promising parts of the solution space identified by a tabu search heuristic. A similar approach has been presented by Archetti et al. [1] for an inventory routing problem. Hewitt et al. [15] proposed to solve the capacitated fixed charge network flow problem by combining exact and heuristic approaches. In this

* Corresponding author. Tel.: +39 051 2093028; fax: +39 051 2093073.
E-mail addresses: majid.salari2@unibo.it (M. Salari), paolo.toth@unibo.it (P. Toth), andrea.tramontani@unibo.it (A. Tramontani).

0305-0548/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2010.02.010

case as well a key ingredient of the method is to use ILP to improve feasible solutions found during the search. Finally, the idea of exploiting ILP to explore promising neighborhoods of feasible solutions has been also investigated in the context of general purpose integer programs; see, e.g., Fischetti and Lodi [10] and Danna et al. [6]. The methods presented in [6,10] are currently embedded in the commercial mixed integer programming solver Cplex [16].

The paper is organized as follows. Section 2 recalls the main works proposed in the literature for OVRP. In Section 3 we describe a neighborhood for OVRP and the ILP model which allows to implicitly define and explore the presented neighborhood. The implementation of the heuristic improvement procedure is given in Section 4, while Section 5 reports the computational experiments on benchmark capacitated OVRP instances from the literature (with/without distance constraints), comparing the presented method with the most effective metaheuristic techniques proposed for OVRP. Some conclusions are finally drawn in Section 6.

2. Literature review

The classical VRP is a fundamental combinatorial optimization problem which has been widely studied in the literature (see, e.g., Toth and Vigo [28] and Cordeau et al. [5]). At first glance, having open routes instead of closed ones looks like a minor change, and in fact OVRP can be also formulated as a VRP on a directed graph, by fixing to 0 the cost of each arc entering the depot. However, if the undirected case is considered, the open version turns out to be more general than the closed one. Indeed, as shown by Letchford et al. [17], any closed VRP on n customers in a complete undirected graph can be transformed into an OVRP on n customers, but there is no transformation in the reverse direction. Further, there are many practical applications in which OVRP naturally arises. This happens, of course, when a company does not own a vehicle fleet, and hence customers are served by hired vehicles which are not required to come back to the depot (see, e.g., Tarantilis et al. [26]). But the *open model* also arises in pick-up and delivery applications, where each vehicle starts at the depot, delivers to a set of customers and then it is required to visit the same customers in reverse order, picking up items that have to be backhauled to the depot. An application of this type is described in Schrage [23]. Further areas of application, involving the planning of train services and of school bus routes, are reported by Fu et al. [13].

OVRP has recently received an increasing attention in the literature. Exact branch-and-cut and branch-cut-and-price approaches have been proposed, respectively, by Letchford et al. [17] and Pessoa et al. [19], addressing the capacitated problem with no distance constraints and no empty routes allowed (i.e., $D=\infty$ and exactly m vehicles must be used). Heuristic and metaheuristic algorithms usually take into account both capacity and distance constraints, and consider the number of routes as a decision variable. In particular, an unlimited number of vehicles is supposed to be available (i.e., $m=\infty$) and the objective function is generally to minimize the number of used vehicles first and the traveling cost second, assuming that the fixed cost of an additional vehicle always exceeds any traveling cost that could be saved by its use (i.e., considering $F=\infty$). However, several authors address as well the variant in which there are no fixed costs associated with the vehicles (i.e., $F=0$) and hence the objective function is to minimize the total traveling cost with no attention to the number of used vehicles (see, e.g., Tarantilis et al. [26]). Considering capacity constraints only (i.e., taking $D=\infty$), Sariklis and Powell [22] propose a two-phase heuristic which first assigns customers to clusters and then builds a Hamiltonian path for each cluster, Tarantilis et al. [24] describe a population-based heuristic, while Tarantilis et al. [25,26] present threshold accepting metaheuristics. Taking into account both capacity and distance constraints, Brandão [3], Fu et al. [13,14] and Derigs and Reuter [8] propose tabu search heuristics, Li et al. [18] describe a record-to-record travel heuristic, Pisinger and Ropke [20] present an adaptive large neighborhood search heuristic which follows a destruct-and-repair paradigm, while Fleszar et al. [12] propose a variable neighborhood search heuristic.

3. Reallocation model

Let z be a feasible solution of the OVRP defined on G. For any given node subset $\mathcal{F}\subset V\backslash\{0\}$, we define $z(\mathcal{F})$ as the *restricted solution* obtained from z by *extracting* (i.e., by short-cutting) all the nodes $v\in\mathcal{F}$. Let $\mathcal{R}$ be the set of routes in the restricted solution, $\mathcal{I}=\mathcal{I}(z,\mathcal{F})$ the set of all the edges in $z(\mathcal{F})$, and $\mathcal{S}=\mathcal{S}(\mathcal{F})$ the set of all the *sequences* which can be obtained through the recombination of nodes in $\mathcal{F}$ (i.e., the set of all the elementary paths in $\mathcal{F}$). Each edge $i\in\mathcal{I}$ is viewed as a potential *insertion point* which can allocate one or more nodes in $\mathcal{F}$ through at most one sequence $s\in\mathcal{S}$. We say that the insertion point $i=(a,b)\in\mathcal{I}$ allocates the nodes $\{v_j\in\mathcal{F}: j=1,\dots,h\}$ through the sequence $s=(v_1,v_2,\dots,v_h)\in\mathcal{S}$, if the edge (a,b) in the restricted solution is replaced by the edges $(a,v_1),(v_1,v_2),\dots,(v_h,b)$ in the new feasible solution. Since the restricted routes, as well as the final ones, are open paths starting at the depot, in addition to the edges of the restricted solution we also consider the insertion points (called *appending insertion points* in the following) $i=(p_r,0)$, where p_r denotes the last customer visited by route $r\in\mathcal{R}$, which allow to append any sequence to the last customer of any restricted route. Further, empty routes in the restricted solution are associated with insertion points (0,0).

For each sequence $s\in\mathcal{S}$, $c(s)$ and $q(s)$ denote, respectively, the cost of the elementary path corresponding to s and the sum of the demands of the nodes in s. For each insertion point $i=(a,b)\in\mathcal{I}$ and for each sequence $s=(v_1,v_2,\dots,v_h)\in\mathcal{S}$, γ_{si} denotes the extra-cost (i.e., the extra-distance) for assigning sequence s to insertion point i in its best possible orientation (i.e., $\gamma_{si}:=c(s)-c_{ab}+\min\{c_{av_1}+c_{v_hb},c_{av_h}+c_{v_1b}\}$). Note that, for the appending insertion points $i=(p_r,0)$, γ_{si} is computed as $c(s)+\min\{c_{p_rv_1},c_{p_rv_h}\}$. The extra-cost for assigning the sequence s to the insertion point $i=(0,0)$ associated with an empty route is simply $c(s)+\min\{c_{0v_1},c_{0v_h}\}$. For each route $r\in\mathcal{R}$, $\mathcal{I}(r)$ denotes the set of insertion points associated with r, while $\tilde{q}(r)$ and $\tilde{c}(r)$ denote, respectively, the total demand and the total distance computed for route r, still in the restricted solution.

For each $i\in\mathcal{I}$, $\mathcal{S}_i\subseteq\mathcal{S}$ denotes a sequence subset containing the sequences which can be allocated to the specific insertion point i. The definition of $\mathcal{S}_i$ will be discussed later in this section. Then, a neighborhood of the given solution z can be formulated (and explored) by solving an ILP problem (denoted as the *Reallocation Model*) based on the decision variables

$$x_{si}=\begin{cases}1 & \text{if sequence } s\in\mathcal{S}_i \text{ is allocated to insertion point } i\in\mathcal{I},\\ 0 & \text{otherwise,}\end{cases} \tag{1}$$

which reads as follows:

$$\sum_{r\in\mathcal{R}}\tilde{c}(r)+\min\sum_{i\in\mathcal{I}}\sum_{s\in\mathcal{S}_i}\gamma_{si}x_{si} \tag{2}$$

subject to

$$\sum_{i\in\mathcal{I}}\sum_{s\in\mathcal{S}_i(v)}x_{si}=1,\quad v\in\mathcal{F}, \tag{3}$$

$$\sum_{s\in\mathcal{S}_i}x_{si}\le 1,\quad i\in\mathcal{I}, \tag{4}$$