Computers & Operations Research 74 (2016) 21–30

Contents lists available at ScienceDirect

Computers & Operations Research

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

Solving the team orienteering problem with cutting planes

Racha El-Hajj [a,c,*], Duc-Cuong Dang [b], Aziz Moukrim [a]

[a] Sorbonne universités, Université de Technologie de Compiègne, CNRS, Heudiasyc UMR 7253, CS 60 319, 60 203 Compiègne cedex
[b] University of Nottingham, School of Computer Science, Jubilee Campus, Wollaton Road, Nottingham NG8 1BB, United Kingdom
[c] Université Libanaise, École Doctorale des Sciences et de Technologie, Campus Hadath, Beyrouth, Liban

ARTICLE INFO

Article history:
Received 24 September 2015
Received in revised form
26 March 2016
Accepted 8 April 2016
Available online 9 April 2016

Keywords:
Team Orienteering Problem
Cutting planes
Dominance property
Incompatibility
Clique cut
Independent-set cut

ABSTRACT

The Team Orienteering Problem (TOP) is an attractive variant of the Vehicle Routing Problem (VRP). The aim is to select customers and at the same time organize the visits for a vehicle fleet so as to maximize the collected profits and subject to a travel time restriction on each vehicle.

In this paper, we investigate the effective use of a linear formulation with polynomial number of variables to solve TOP. Cutting planes are the core components of our solving algorithm. It is first used to solve smaller and intermediate models of the original problem by considering fewer vehicles. Useful information are then retrieved to solve larger models, and eventually reaching the original problem. Relatively new and dedicated methods for TOP, such as identification of irrelevant arcs and mandatory customers, clique and independent-set cuts based on the incompatibilities, and profit/customer restriction on subsets of vehicles, are introduced.

We evaluated our algorithm on the standard benchmark of TOP. The results show that the algorithm is competitive and is able to prove the optimality for 12 instances previously unsolved.

1. Introduction

The Team Orienteering Problem (TOP) was first mentioned in Butt and Cavalier [7] as the Multiple Tour Maximum Collection Problem (MTMCP). Later, the term TOP was formally introduced in Chao et al. [9]. TOP is a variant of the Vehicle Routing Problem (VRP) [4]. In this variant, a limited number of identical vehicles is available to visit customers from a potential set. Two particular depots, the *departure* and the *arrival* points are considered. Each vehicle must perform its route starting from the departure depot and returning to the arrival depot without exceeding its predefined travel time limit. A certain amount of profit is associated with each customer and must be collected at most once by the fleet of vehicles. The aim is to organize an itinerary of visits respecting the above constraints for the fleet in such a way that the total amount of collected profits from the visited customers is maximized.

A special case of TOP is the one with a single vehicle. The resulted problem is known as the Orienteering Problem (OP), or the Selective Travelling Salesman Problem (STSP) (see the surveys by Feillet et al. [13], Vansteenwegen et al. [32] and Gavalas et al. [16]). OP/STSP is already NP-Hard [22], and so is TOP [9]. The applications of TOP arise in various situations. For example in Bouly et al. [5], the authors used TOP to model the schedule of inspecting and repairing tasks in water distribution. Each task in this case has a specific level of urgency which is similar to a profit. Due to the limitation of available human and material resources, the efficient selection of tasks as well as the route planning become crucial to the quality of the schedule. A very similar application was described in Tang and Miller-Hooks [29] to route technicians to repair sites. In Souffriau et al. [26], Vansteenwegen et al. [31] and Gavalas et al. [16], the tourist guide service that offers to the customers the possibility to personalize their trips is discussed as variants of TOP/OP. In this case, the objective is to maximize the interest of customers on attractive places subject to their duration of stay. Those planning problems are called Tourist Trip Design Problems (TTDPs). Many other applications include the team-orienteering sport game, bearing the original name of TOP, the home fuel delivery problem with multiple vehicles (e.g., Chao et al. [9]) and the athlete recruiting from high schools for a college team (e.g., Butt and Cavalier [7]).

Many heuristics have been proposed to solve TOP, like the ones in Archetti et al. [2], Souffriau et al. [27], Dang et al. [12] and Kim et al. [20]. These approaches are able to construct solutions of good quality in short computational times, but those solutions are not necessarily optimal. In order to validate them and evaluate the performance of the heuristic approaches, either optimal solutions or upper bounds are required. For this reason, some researches have been dedicated to elaborate exact solution methods for TOP.

* Corresponding author.
E-mail addresses: racha.el-hajj@hds.utc.fr (R. El-Hajj), duc-cuong.dang@nottingham.ac.uk (D.-C. Dang), aziz.moukrim@hds.utc.fr (A. Moukrim).

http://dx.doi.org/10.1016/j.cor.2016.04.008
0305-0548/

Butt and Ryan [8] introduced a procedure based on the set covering formulation. A column generation algorithm was developed to solve this problem. In Boussier et al. [6], the authors proposed a branch-and-price (B–P) algorithm in which they used a dynamic programming approach to solve the pricing problem. Their approach has the advantage of being easily adaptable to different variants of the problem. Later, Poggi de Aragão et al. [25] introduced a pseudo-polynomial linear model for TOP and proposed a branch-cut-and-price (B–C–P) algorithm. New classes of inequalities, including min-cut and triangle clique, were added to the model and the resulting formulation was solved using a column generation approach. Afterwards, Dang et al. [11] proposed a branch-and-cut (B–C) algorithm based on a linear formulation and features a new set of valid inequalities and dominance properties in order to accelerate the solution process. Recently, Keshtkarana et al. [19] proposed a Branch-and-Price algorithm with two relaxation stages (B–P–2R) and a Branch-and-Cut-and-Price (B–C–P) approach to solve TOP, where a bounded bidirectional dynamic programming algorithm with decremental state space relaxation was used to solve the subproblems. These five methods were able to prove the optimality for a large part of the standard benchmark of TOP [9], however there is a large number of instances that are still open until now. Furthermore, according to the recent studies of Dang et al. [12] and Kim et al. [20], it appears that it is hardly possible to improve the already-known solutions for the standard benchmark of TOP using heuristics. These studies suggest that the known heuristic solutions could be optimal but there is a lack of variety of effective methods to prove their optimality.

Motivated by the above facts, in this paper we propose a new exact algorithm to solve TOP. It is based on a linear formulation with a polynomial number of binary variables. Our algorithmic scheme is a cutting plane algorithm which exploits integer solutions of successive models with the *subtour* elimination constraints being relaxed at first and then iteratively reinforced. Recently, Pferschy and Staněk [24] demonstrated on the Travelling Salesman Problem (TSP) that such a technique which was almost forgotten could be made efficient now a day with the impressive performance of modern solvers for Mixed-Integer Programming (MIP), especially with a careful control over the reinforcing of the subtour elimination. Our approach is similar but in addition to subtour elimination, we also make use of other valid inequalities and useful dominance properties to enhance the intermediate models. The properties include breaking the symmetry and exploiting bounds or optimal solutions of smaller instances/models with fewer number of vehicles, while the proposed valid inequalities are the clique cuts and the independent set cuts based on the incompatibilities between customers and between arcs. In addition, bounds on smaller restricted models are used to locate mandatory customers and inaccessible customers/arcs. Some of these cuts were introduced and tested in Dang et al. [11] yielding some interesting results for TOP, this encourages us to implement them in our cutting plane algorithm. We evaluated our algorithm on the standard benchmark of TOP. The obtained results clearly show the competitiveness of our algorithm. The algorithm is able to prove the optimality for 12 instances that none of the previous exact algorithms had been able to solve.

The remainder of the paper is organized as follows. A short description of the problem with its mathematical formulation is first given in Section 2, where the use of the generalized subtour elimination constraints is also discussed. In Section 3, the set of dominance properties, which includes symmetry breaking, removal of irrelevant components, identification of mandatory customers and boundaries on profits/numbers of customers, is presented. The graphs of incompatibilities between variables are also described in this section, along with the clique cuts and the independent set cuts. In Section 4, all the techniques used to generate these efficient cuts are detailed, and the pseudocode of the main algorithmic scheme is given. Finally, the numerical results are discussed in Section 5, and some conclusions are drawn.

2. Problem formulation

TOP is modeled with a complete directed graph $G = (V, A)$ where $V = \{1, \ldots, n\} \cup \{d, a\}$ is the set of vertices representing the customers and the depots, and $A = \{(i, j) | i, j \in V, i \neq j\}$ the set of arcs linking the different vertices together. The departure and the arrival depots for the vehicles are represented by the vertices d and a. For convenience, we use three sets V^-, V^d and V^a to denote respectively the sets of the customers only, of the customers with the departure depot and of the customers with the arrival one. A profit p_i is associated for each vertex i and is considered zero for the two depots ($p_d = p_a = 0$). Each arc $(i, j) \in A$ is associated with a travel cost c_{ij}. Theses costs are assumed to be symmetric and to satisfy the triangle inequality. All arcs incoming to the departure depot and outgoing from the arrival one must not be considered ($c_{id} = c_{ai} = \infty, \forall i \in V^-$). Let F represent the fleet of the m identical vehicles available to visit customers. Each vehicle must start its route from d, visit a certain number of customers and return to a without exceeding its predefined travel cost limit L. Using these definitions, we can formulate TOP with a linear Mixed Integer Program (MIP) using a polynomial number of decision variables y_{ir} and x_{ijr}. Variable y_{ir} is set to 1 if vehicle r has served client i and to 0 otherwise, while variable x_{ijr} takes the value 1 when vehicle r uses arc (i,j) to serve customer j immediately after customer i and 0 otherwise.

$$\max \sum_{i \in V^-} \sum_{r \in F} y_{ir} p_i \tag{1}$$

$$\sum_{r \in F} y_{ir} \leq 1 \quad \forall\, i \in V^- \tag{2}$$

$$\sum_{j \in V^a} x_{djr} = \sum_{j \in V^d} x_{jar} = 1 \quad \forall\, r \in F \tag{3}$$

$$\sum_{i \in V^a \setminus \{k\}} x_{kir} = \sum_{j \in V^d \setminus \{k\}} x_{jkr} = y_{kr} \quad \forall\, k \in V^-, \forall\, r \in F \tag{4}$$

$$\sum_{i \in V^d} \sum_{j \in V^a \setminus \{i\}} c_{ij} x_{ijr} \leq L \quad \forall\, r \in F \tag{5}$$

$$\sum_{(i,j) \in U \times U} x_{ijr} \leq |U| - 1 \quad \forall\, U \subseteq V^-, |U| \geq 2, \forall\, r \in F \tag{6}$$

$$x_{ijr} \in \{0, 1\} \quad \forall\, i \in V, \forall\, j \in V, \forall\, r \in F \tag{7}$$

$$y_{ir} \in \{0, 1\} \quad \forall\, i \in V^-, \forall\, r \in F$$

The objective function (1) maximizes the sum of collected profits from the visited customers. Constraints (2) impose that each customer must be visited at most once by one vehicle. Constraints (3) guarantee that each vehicle starts its path at vertex d and ends it at vertex a, while constraints (4) ensure the connectivity of each tour. Constraints (5) are used to impose the travel length restriction, while constraints (6) eliminate all possible subtours, i.e. cycles excluding the depots, from the solution. Finally, constraints (7) set the integral requirement on the variables.

Enumerating all constraints (6) yields a formulation with an exponential number of constraints. In practice, these constraints are first relaxed from the formulation, then only added to the model whenever needed. The latter can be detected with the