A Branch-And-Check Solution Method for a Tourist Trip Design Problem with Rich Constraints

Duc Minh Vu, Yannick Kergosien, Jorge E. Mendoza, Pierre Desport

A Branch-And-Check Solution Method for a Tourist Trip Design Problem with Rich Constraints

Duc Minh Vu[1], Yannick Kergosien[1], Jorge E. Mendoza[2], Pierre Desport[1]

[1] PolyTech Tours, LIFAT, 64 avenue Jean Portalis, Tours, 37000, France
{ducminh.vu,yannick.kergosien,pierre.desport}@univ-tours.fr
[2] HEC Montréal, 3000 Chemin de la Côte Sainte Catherine, Montréal QC, H3T 2A7, Canada
jorge.mendoza@hec.ca

Mots-clés : *tourist trip design, orienteering problem, branch-and-check, constraint programming.*

1 Introduction

The Tourist Trip Design problem (TTDP) consists of building an itinerary to visit a set of points of interest (POI) that maximizes tourist satisfaction while meeting a set of constraints (e.g., budget, maximum duration). The TTDP is basically an application of the orienteering problem (OP) in tourism context[3, 8]. For a more detailed explanation of the TTDP and its variants, the readers can refer to [2, 5]. In this research, we address a TTDP variant laying at the heart of online and mobile applications for tourist trip planning. Our problem is summarized as follows. A tourist can visit a set of possible POIs, each POI has : a *score* (e.g. indicating how attractive to him/her the POI is), an opening time window, an entrance fee and its categories (e.g. castle, museum, etc.). The problem is to find the subset of POIs that the tourist should visit by maximizing the total score of visited POIs and then the best order to visit them while meeting the following constraints :

1. Starting and ending locations : the trip starts from a specified location and ends at a specified location.
2. Distinction : a POI can be visited at most once.
3. Time windows : the tourist can only visit a POI during its opening time window.
4. Duration of the tour (D) : the duration of the trip does not exceed a parameter D.
5. Available budget (B) : the sum of entrance fees to the visited sites does not exceed a parameter B.
6. Category constraint (E_z) : the number of POIs of category z that are visited does not exceed a parameter E_z.
7. Mandatory locations M : the trip must visit all POIs in M.
8. Conflict between POIs $(u, v) \in \Gamma$: the trip cannot include both POIs u and v if $(u, v) \in \Gamma$, the set of conflict POIs.
9. Implication between POIs $(u, v) \in \Theta$: the trip must visit POI v if it visits POI u.
10. Precedence between POIs $(u, v) \in \Omega$: the trip must visit POI v after visiting u if both POIs are selected to be part of the trip.

2 Solution approach

We formulated the problem as a mixed integer program (MIP) and ran experiments with a state-of-art solver (GUROBI 7.5.2). To conduct our experiments, we generated a benchmark set of 37 instances based on the benchmarks of [6]. After running the solver for 1 hour, we could only reach solutions with an average gap of 36%. We then turned to the branch-and-check (B&Ck) framework [7] in an attempt to develop a better exact solution method for our problem. The underlying idea of B&Ck is to exploit the strengths of mixed integer programming and constraint programming (CP) solvers. In a nutshell, in the B&Ck framework, a problem is modeled as a mixture of CP and MIP. Then, the method typically solves a linear relaxation of the MIP part (the master problem) at every node and branches on discrete variables, but it only solves the CP part (the subproblem) at the nodes of the branching tree where it is advantageous (e.g. if an improved feasible solution could be found). B&Ck has been successfully applied to problems such as Vehicle Routing with Time Windows [4] and Wind Turbine Maintenance Scheduling [1].

Our approach solves the TTDP in two steps : the first step solves a master problem, which is a relaxation of the original problem, aiming to determine a set of POIs that satisfy a subset of constraints (1), (2), (5)-(9). Given the selected set of POIs, the constraint programming solver will determine if there is a feasible visiting sequence that both covering the selected POIs and satisfying constraints (1), (3), (4) and (10). If there is no feasible solution, a cut is generated and added to the master problem to eliminate this solution and (hopefully) many other similar solutions. Otherwise, if those POIs belong to an optimal solution of the master problem, the algorithm stops since the optimal solution is found. More details about the proposed method and computation results will be presented during the ROADEF 2019 conference.

Références

A. Froger, M. Gendreau, J. E. Mendoza, E. Pinson, and L.-M. Rousseau. A branch-and-check approach for a wind turbine maintenance scheduling problem. *Computers & Operations Research*, 88 :117–136, 2017.

D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou. A survey on algorithmic approaches for solving tourist trip design problems. *Journal of Heuristics*, 20(3) :291–328, 2014.

A. Gunawan, H. C. Lau, and P. Vansteenwegen. Orienteering problem : A survey of recent variants, solution approaches and applications. *European Journal of Operational Research*, 255(2) :315–332, 2016.

E. Lam and P. Van Hentenryck. Branch-and-check with explanations for the vehicle routing problem with time windows. In *International Conference on Principles and Practice of Constraint Programming*, pages 579–595. Springer, 2017.

W. Souffriau and P. Vansteenwegen. Tourist trip planning functionalities : State–of–the–art and future. In F. Daniel and F. M. Facca, editors, *Current Trends in Web Engineering*, pages 474–485, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, and D. Van Oudheusden. The multi-constraint team orienteering problem with multiple time windows. *Transportation Science*, 47(1) :53–63, 2013.

E. S. Thorsteinsson. Branch-and-check : A hybrid framework integrating mixed integer programming and constraint logic programming. In *International Conference on Principles and Practice of Constraint Programming*, pages 16–30. Springer, 2001.

P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden. The orienteering problem : A survey. *European Journal of Operational Research*, 209(1) :1–10, 2011.