# Using a perturbation strategy for the team orienting problem

Adeline Goullieux[1], Mhand Hifi[1], Shohre Sadeghsa[1]

EPROAD EA 4669, Université de Picardie Jules Verne
7 rue du Moulin Neuf, 80000 Amiens, France
{adeline.goullieux,mhand.hifi,shohre.sadeghsa}@u-picardie.fr

**Mots-clés** : *Team orienteering, population, perturbation.*

## 1 Introduction

The Orienting Problem (OP) was first introduced [**?**], where they described a variety of Traveling Salesman Problem (TSP) in which a vehicle will start its trajectory from a starting point called "depot". The vehicle should visit subset of points in order to maximize the profit that gains from each visited point. If we consider the OP with multiple vehicles, TSP will turns into a VRP (Vehicle Routing Problem). Then by adapting the feature of choosing points, OP will be replaced with TOP (Team Orienting Problem). It would be interesting to refer to cyclic inventory routing problem [**?**], where authors solved the problem with exact methods (Note that TOP has attracted a lot of interest in recent papers such as in [**?**] and [**?**]). Herein, the TOP is tackled as a single objective function optimization problem, where the study can be viewed as a first step for studying the multi-objective optimization problem, like presented in [**?**].

## 2 Problem description

VRP is often represented as a directed graph, where nodes and edges characterize the customers and routes, respectively. Let define $S = (N, R)$ as a feasible solution, where $N$ represents the set of unvisited nodes and $R = r_1, r_2, .., r_m$ the set of feasible routes, where m denotes a maximum number of vehicles. A route is defined as a permutation of visited nodes. In a feasible solution each route starts from a node (depot), visits a subset of customers in a route and, finally returns to the depot. Let $t_{i,j}$ be the travel time between each two visited nodes (namely $i$ and $j$, where $i \neq j$). The objective function is to maximize the gained profit from the customers using a fixed number of vehicles and fixed maximum travel time for each route. In each feasible route $\sum_{i,j} t_{i,j} \leq T_{max} \forall r \in R$, where $T_{max}$ denotes the maximum travel time of the stated route. It should be noted that during the one solution time, each customer must be served fully and in one time. Due to its complexity, a population-based approach is proposed, where both destroying and rebuilding strategies are combined with a deep searching.

## 3 Solution procedure

The proposed approach mimics the genetic algorithm frame-work enhanced with variable neighborhood search. The process is based upon building, breaking (destroying) and rebuilding a series of feasible solutions to seek the search space with high diversification and deep local search strategies. The process starts with a feasible solution : (i) local search operators are used for minimizing the travelled time, (ii) other search operators are applied for maximizing the profit by adding nodes from unvisited ones and, (iii) a diversification procedure using a perturbation strategy is applied to the current solution. In order to maintain certain degree of diversify of the solutions in the population, different strategies to create a starting population are considered.

• *Initial population.* The starting population consist of a set of solutions, where each of them consists of sets of routes (a sequence of visited nodes) and unvisited nodes. In this case, the costumers will be collected in a parallel way, where these vehicles are allowed to visit only one client based on its preference ratio. Such an initialization tries to create a balanced solution. Size of the population and the number of iterations is a portion of the instance size. This affects directly the execution time. (Maximum execution time for the most difficult instances is up to two hours and for the easy and moderate instances is within less than 1 to 10 minutes).

• *Deep searching strategy.* Often the local search will converge to a solution toward the local optima. As a matter of fact local search is a complementary procedure for the evolutionary process. This study applied search operators to intensify the search process, local moves in the intensification strategy are applied (like, permutation in one tour, permutation between tours).

• *Perturbation procedure.* The diversification strategy tries to discover the search space and with sufficient iterations guarantee to converge to an eventual global optima. The diversification randomly removes $\alpha$ (a given percentage) customers from the current solution, providing a partial solution (namely $S'$) with the rest of the customers. Such a process will perturb the solution and will move the search space to unvisited areas. Hence, the diversification strategy can improve the quality of the solution although there is no guarantee to always improve it.

# 4 Experimental part

In our preliminary study, the method was coded in C++ on OS version 10.14.5 with 2.3 GHz Intel Core i5 processor. All tested instances were taken from [**?**] and all bounds are compared the available ones extracted from Bouly et al. [9] – (cf. Table 1).

| Inst. | Size | Difficulty | Best bound | [**?**] | [**?**] | [**?**] | This study |
|---|---|---|---|---|---|---|---|
| $P1\_04\_r$ | 32 | easy | 210 | 210 | 210 | 210 | 210 |
| $P2\_03\_k$ | 21 | moderate | 200 | 200 | 200 | 200 | 200 |
| $P2\_04\_e$ | 21 | moderate | 70 | 70 | 70 | 70 | 70 |
| $P2\_04\_k$ | 21 | moderate | 180 | 180 | 180 | 180 | 180 |
| $P3\_03\_i$ | 33 | moderate | 330 | 330 | 330 | 330 | 330 |
| $P3\_04\_t$ | 33 | moderate | 670 | 670 | 670 | 670 | 670 |
| $P5\_02\_z$ | 66 | difficult | 1680 | 1672.5 | 1670 | 1680 | 1680 |
| $P5\_04\_z$ | 66 | difficult | 1620 | 1585.5 | 1620 | 1620 | 1620 |

TAB. 1 – Behavior of the proposed method versus three available methods

From Table 1, one can observe that for all tested instances, the proposed preliminary method is able to match all better solutions available in the literature. Indeed, for these instances, the proposed method matches bounds reached by Bouly et al's [**?**]algorithm, it improves one bound when compared to those provided by Archetti et al.'s [**?**] approach and, in two cases it dominates those achieved by Ke et al.'s [**?**] method. Note that the proposed method achieves all results in minutes seconds when compared to the highest runtimes needed by exact or/and hybrid methods of the literature.

# Références


[1] Archetti, C., Hertz, A. and Speranza, M.G., Metaheuristics for the team orienteering problem. *Journal of Heuristics*, 13(1), 49-76, 2007.

[2] Bederina, H. and Hifi, M., 2017. A hybrid multi-objective evolutionary algorithm for the team orienteering problem. In *Proceedings of the IEEE 4th International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 0898-0903), 2017.

[3] Bouly, H., Dang, D.C. and Moukrim, A. A memetic algorithm for the team orienteering problem. *4OR*, 8(1), 49-70, 2010.

[4] Chao, I.M., Golden, B.L. and Wasil, E.A. The team orienteering problem. *European journal of operational research*, 88(3), 464-474, 1996.

[5] Ke, L., Archetti, C. and Feng, Z. Ants can solve the team orienteering problem. *Computers & Industrial Engineering*, 54(3), 648-665, 2008.

[6] Lefever, W., Aghezzaf, E.H. and Hadj-Hamou, K., A convex optimization approach for solving the single-vehicle cyclic inventory routing problem. *Computers & Operations Research*, 72, pp.97-106, 2016.

[7] Bianchessi, N., Mansini, R. and Speranza, M.G. A branch-and-cut algorithm for the Team Orienteering Problem. *International Transactions in Operational Research*,25(2), pp.627-635, 2018.

[8] Pessoa, A., Sadykov, R., Uchoa, E. and Vanderbeck, F. May. A generic exact solver for vehicle routing and related problems. *International Conference on Integer Programming and Combinatorial Optimization*, pp. 354-369, 2019.