



**HAL**  
open science

# Branch-and-price algorithms for the bi-objective vehicle routing problem

Nicolas Jozefowicz, Sandra Ulrich Ngueveu, Estèle Glize

► **To cite this version:**

Nicolas Jozefowicz, Sandra Ulrich Ngueveu, Estèle Glize. Branch-and-price algorithms for the bi-objective vehicle routing problem. Odysseus 2018 Seventh International Workshop on Freight Transportation and Logistics, Jun 2018, Cagliari, Italy. hal-01880521

**HAL Id: hal-01880521**

**<https://hal.science/hal-01880521>**

Submitted on 25 Sep 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Branch-and-price algorithms for the bi-objective vehicle routing problem

**Nicolas Jozefowiez**

LCOMS, Universit de Lorraine, Metz, France,

**Sandra Ulrich Ngueveu**

CNRS, LAAS, INP Toulouse, Toulouse, France,

**Estèle Glize**

CNRS, LAAS, INSA, Toulouse, France

Email: [estele.glize@laas.fr](mailto:estele.glize@laas.fr)

## 1 Context

This paper presents an exact method for the bi-objective Vehicle Routing Problem (BOVRP) where the objectives are to minimize two different lengths on the routes like the travel distance and the cost of the journey. These two objectives can be conflictive: in motor vehicle, the travel time differs from the fuel consumption. Many industrials are interested in finding a good compromise. So the BOVRP presenting two different costs  $c^1$  and  $c^2$  on each route is a relevant challenge.

The multi-objective VRP (MOVVRP) is more and more studied. But due to its complexity and the possibility of objective functions (number of vehicles, fairness...), this particular case has not been explored yet. A complete survey of MOVVRP can be found in *Jozefowiez et al.* [1]. *Reiter and Gutjahr* [2] has solved exactly the BOVRP minimizing the distance and the length of the longest route using an adaptive  $\epsilon$ -constraint method.

In the larger scope of multi-objective integer programming, exact methods are divided into two classes: methods working on the feasible solutions space [3] and those working on the objective functions space [4]. These last methods solve a sequence of mono-objective problem and so, lean on their efficiency on single-objective integer programming solver. The  $\epsilon$ -constraint method is one of the most effective objective space search algorithm [2, 5].

The main idea of this paper is to propose a competitive method for finding the complete and exact Pareto front of BOVRP where the objectives are to minimize the total lengths of the journey, mainly in clustered graph.

## 2 Methods

We propose an objective space search algorithm to solve the BOVRP based on an efficient method that optimally solves the mono-objective VRP: the method of *Baldacci et al.* [6]. It generates all routes whose reduced cost is contained between a lower bound  $LB$  (optimal solution of the linear relaxation) and an upper bound  $UB$  (a feasible solution of the integer problem). This column generation will be referred to as  $\mathcal{BALCD}(UB, LB)$  and produces a reduced set of routes  $\bar{\Omega}$ . The integer problem is finally solved on  $\bar{\Omega}$  by an integer solver.

### 2.1 reference method

This method is called the reference method as it is the more direct way to use the *Baldacci et al.* method in a BOVRP within an  $\epsilon$ -constraint method.

The  $\epsilon$ -constraint minimizes the first cost  $c^1$  under the constraint that the second cost  $c^2$  has to be lower than a certain value  $\epsilon$ . The *reference method* can be summed up as follows:

---

**Algorithm 1** Algorithm of the reference method

---

$\epsilon \leftarrow +\infty$

**while**  $\exists$  a solution **do**

    Solve the linear relaxation of the problem for  $\epsilon$  to obtain  $LB$

    Find a feasible solution  $UB$  of the integer problem for  $\epsilon$

$\bar{\Omega} \leftarrow \mathcal{BALCD}(UB, LB)$

    Solve the integer problem on  $\bar{\Omega}$  to obtain  $S_{OPT}$

    Set  $\epsilon \leftarrow S_{OPT}^2 - 1$

**end while**

---

### 2.2 two-steps method

This second method minimizes the weighted-sum of the two objectives  $\lambda c^1 + (1 - \lambda)c^2$ ,  $\lambda \in [0, 1]$ . We also need to introduce the call of *Baldacci et al.* method for the weight  $\lambda$ :  $\mathcal{BALCD}(UB, LB, \lambda)$ , where the cost of a solution is computed with respect to  $\lambda$ .

The algorithm is divided into two steps. First, the supported points are computed in a dichotomous approach and then, the non-supported points are found in area not explored yet (Algorithm 2).

The first step needs in input the optimal solutions  $S_1$  and  $S_2$  that minimizes the costs  $c^1$  and  $c^2$  respectively. The gradient  $\lambda_1$  of these two points is computed. Then, the optimal solution  $LB_1$  of the linear relaxation of the parametrized formulation for  $\lambda_1$  is computed. Finally, we apply the algorithm  $\mathcal{BALCD}(LB_1, UB_1=S_1, \lambda_1)$  to have a reduced set of routes  $\bar{\Omega}$  on which we optimally solve the integer problem for  $\lambda_1$  to obtain  $S_3$ . If the solution is

different from  $S_1$  and  $S_2$ , we apply a recursive algorithm that computes the gradient  $\lambda_i$  of successive solutions and optimally solves the integer problem on  $\bar{\Omega}$  in the new direction  $\lambda_i$ .

The second step aims to explore each triangle defined by two consecutive supported points and their nadir point. The algorithm between two consecutive non-dominated points is defined in Algorithm 2.

---

**Algorithm 2** Second step of *two-steps* method

---

**Require:** Supported Point  $S_i$  and  $S_{i+1}$

    Compute direction  $\lambda_i'$ , gradient between  $S_i$  and  $S_{i+1}$

    Solve the linear relaxation of the parametrized formulation for  $\lambda_i'$  to obtain  $LB_i'$

    Set  $UB_i'$  to the nadir point  $(S_{i+1}^1 - 1, S_i^2 - 1)$

$\bar{\Omega} \leftarrow \mathcal{BALCD}(LB_i', UB_i', \lambda_i')$

    Solve the integer parametrized problem for  $\lambda_i'$ ,  $\text{PMP}(\lambda_i')$ , on  $\bar{\Omega}$  to obtain  $S_i'$

**if**  $\text{PMP}(\lambda_i')$  is feasible **then**

        Repeat algorithm for  $S_i$  and  $S_i'$  and for  $S_i'$  and  $S_{i+1}$

**end if**

---

### 3 Results and Discussion

Each method returns the exact minimum complete set of non-dominated points. To compare the methods introduced in Section 2, pairs of Solomon’s instances have been merged to create BOVRPTW. The implementation is in C++ and the linear problems and the integer problems are solved with Gurobi 7.1.

The figure 1 shows the comparison of the execution time for the methods in graphs with 25 customers. It shows that, for small graphs and random structure, the *two-steps* method is generally more efficient than the *reference method*.

The methods have also been tested on clustered instances and graphs with 50 customers (*instance name beginning with 50*) and the results are presented in the table 1.

Methods	C106_RC1	C106_C2	C106_R1	50R101_C2	50R101_RC2	50RC101_R1	50R105_C1
<i>ref. method</i>	-	-	-	523	778	3208	41803
<i>two-steps</i>	3200	6150	22314	202	273	3061	15354

Table 1: Execution time of the methods in seconds. Time limit of 10 hours.

Finally, for larger graphs and clustered instances, the *two-steps* method becomes much more efficient. Indeed, the *two-steps* method is a good compromise between the number of mono-objective solution of the problem and the size of the gap between lower and upper bounds.

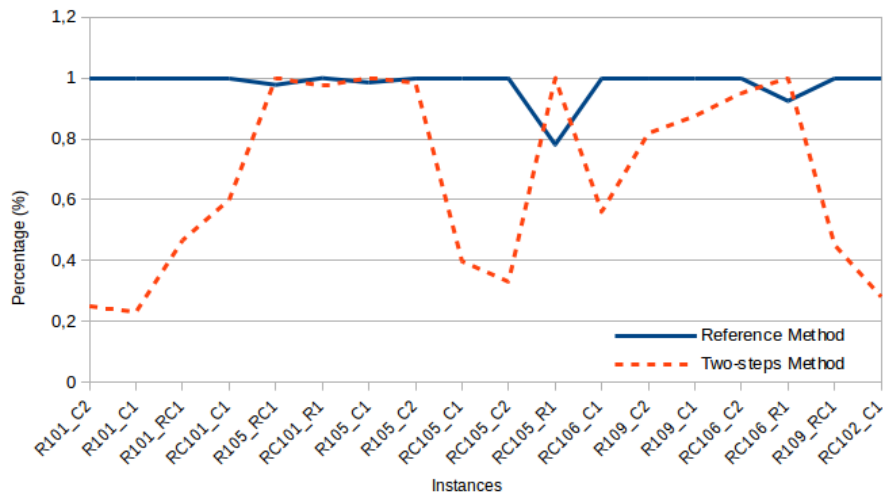


Figure 1: Relative execution time between *reference* and *two-steps* methods

## 4 Conclusion

We have proposed an exact method for BOVRP that can generate all non-dominated points, supported and non-supported. This method is also generic for the class of VRP as it doesn't exploit specific property. The results in Section 3 also prove that *two-steps* method is competitive compared to the *reference method* in most case, mainly for bigger graphs and clustered instances. The method will be improved to be more competitive on all instances by exploiting the bi-objective property directly in the *BACD* procedure and not only by cutting the objective space in different area of research.

## References

- [1] N. Jozefowicz, F. Semet, and T. El-Ghazali, "Multi-objective vehicle routing problems", *European journal of operations research*, vol. 189, 293-309, 2008.
- [2] P. Reiter, and W.J. Gutjahr, "Exact hybrid algorithms for solving a bi-objective vehicle routing problem", *Central European Journal of Operations Research*, vol. 189, 293-309, 2008.
- [3] S.N. Parragh, and F. Tricoire, "Branch-and-bound for bi-objective integer programming", *Optimization online*, 2015.
- [4] N. Boland, H. Charkhgard, and M. Savelsbergh, "The triangle splitting method for bi-objective mixed integer programming", *International Conference on Integer Programming and Combinatorial Optimization*, 162-173, 2014.
- [5] S. Moradi, A. Raith, and M. Ehrgott, "A bi-objective column generation algorithm for the multi-commodity minimum cost flow problem", *European Journal of Operational Research*, vol. 244, 369-378, 2015.
- [6] R. Baldacci, N. Christofides, and A. Mingozzi, "An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts", *Mathematical Programming*, vol. 115, 351-385, 2008.