

**A bi-objective vehicle routing problem  
integrating routing operations into  
tactical clustering decisions**

D. Rocha, D. Aloise,  
D. J. Aloise, C. Contardo

G-2017-30

April 2017

---

Cette version est mise à votre disposition conformément à la politique de libre accès aux publications des organismes subventionnaires canadiens et québécois.

**Avant de citer ce rapport**, veuillez visiter notre site Web (<https://www.gerad.ca/fr/papers/G-2017-30>) afin de mettre à jour vos données de référence, s'il a été publié dans une revue scientifique.

This version is available to you under the open access policy of Canadian and Quebec funding agencies.

**Before citing this report**, please visit our website (<https://www.gerad.ca/en/papers/G-2017-30>) to update your reference data, if it has been published in a scientific journal.

---

Les textes publiés dans la série des rapports de recherche *Les Cahiers du GERAD* n'engagent que la responsabilité de leurs auteurs.

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2017  
– Bibliothèque et Archives Canada, 2017

The authors are exclusively responsible for the content of their research papers published in the series *Les Cahiers du GERAD*.

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2017  
– Library and Archives Canada, 2017



# **A bi-objective vehicle routing problem integrating routing operations into tactical clustering decisions**

**Diego Rocha** <sup>a</sup>

**Daniel Aloise** <sup>b,e</sup>

**Dario J. Aloise** <sup>c</sup>

**Claudio Contardo** <sup>d,e</sup>

<sup>a</sup> *Instituto Federal de Educaçao, Cincia e Tecnologia do Cear, Fortaleza, CE, Brazil*

<sup>b</sup> *Department of Mathematics and Industrial Engineering, Polytechnique Montréal (Québec) Canada*

<sup>c</sup> *Departamento de Informtica, Universidade do Estado do Rio Grande do Norte Mossor, RN, Brasil*

<sup>d</sup> *Département de management et technologie, ESG UQAM, Montréal (Québec) Canada*

<sup>e</sup> *Group for research in decision analysis (GERAD), Montréal (Québec), Canada*

diego.rocha@ifce.edu.br  
daniel.aloise@polymtl.ca  
darioaloise@uern.br  
contardo.claudio@uqam.ca

**April 2017**  
**Les Cahiers du GERAD**  
**G-2017-30**

**Abstract:** In this article we consider a bi-objective vehicle routing problem in which, in addition to the classical minimization of the total routing cost, the operator is also required to minimize the maximum diameter of the routes, this is the maximum distance between any two customers serviced within the same route. This problem arises in applications in which a decision planner needs to integrate the routing decisions into his tactical planning so as to reduce the cost of a potential derouting under uncertainty. In addition to the problem description, we provide a formal linear-integer formulation of the problem and an ad-hoc  $\epsilon$ -constraint method capable of handling small-size problems. We also introduce a variable neighborhood search-based algorithm for the solution of larger problems. We provide a critical analysis of the results obtained after executing our algorithms on some classical instances of the capacitated vehicle routing problem.

---

**Acknowledgments:** Diego Rocha thanks the *Programa de Ps-Graduao em Engenharia Eltrica e Computao* (PPgEEC) of UFRN and the *Coordenao de Aperfeioamento de Pessoal de Nvel Superior* (CAPES) of Brazil, for their financial support. Daniel Aloise and Dario J. Aloise thank the *Conselho Nacional de Pesquisa do Brasil* (CNPq) for its financial support. Claudio Contardo thanks the Natural Sciences and Engineering Research Council of Canada (NSERC) and the *Fonds de recherche du Qubec - Nature et technologies* (FRQNT) for their financial support.

## 1 Introduction

The vehicle routing problem (VRP, Dantzig and Ramser, 1959) is arguably one of the most classic combinatorial optimization problems arising in the logistics chain. The VRP consists in determining the routes that a certain fleet of vehicles must take in order to collect items at known demand points. Each item typically has a certain size or weight associated. The total amount (in terms of either weight or size) of the quantities collected by a single vehicle cannot exceed its capacity. The VRP is, unfortunately, strongly NP-hard even for a single objective as the traveling salesman problem (TSP, Dantzig et al., 1954) can be polynomially reduced to it (Lenstra and Kan, 1981).

Although typical applications of the VRP are single-objective —most notably to minimize the total routing cost—, it is not uncommon to find practitioners being forced to adapt the existing algorithms to be able to provide fast response times to large-scale problems. Perhaps the most classic —and also most intuitive— way of achieving this is by clustering the customers in such a way that the problem may be decomposed into several smaller VRPs, each of which can be handled separately. The danger of doing so lies in the fact that the operational routing costs are completely dismissed when clustering the customer locations. It is not hard to see that this may lead to a loss of efficiency or to needlessly too expensive routing solutions.

Practical situations in which a decision planner may take advantage of an integrated optimization are the following. First, on problems in which the drivers (or equivalently, the operator) are likely to have to react to unexpected changes in the system. In this case, having the customers close to each other within a route may prevent the operator from incurring into excessive extra costs for any derouting that may be necessary to accommodate to the new situation (Zhong et al., 2007; Haugland et al., 2007). Second, to allow a compact regionalization of the service zone so as to allow the drivers to get familiar with their assigned area, while still keeping near-optimal routing costs, which in the long run may lead to a better service (Mourgaya and Vanderbeck, 2007).

In this article, we introduce the VRP with integrated minimization of the total routing cost and the maximum diameter of the clustering associated with the routes. We describe the problem and provide a formal linear-integer formulation and an  $\epsilon$ -constraint method, capable of handling small-size problems. For larger problems, we introduce a variable neighborhood search (VNS) algorithm. We provide a critical analysis of the results obtained and find that in many situations the operator may incur into substantial gains by approaching a VRP using our modeling framework, as compared to the more traditional —sequential— procedure. From an algorithmic viewpoint, we show that our VNS algorithm can provide a fast and accurate response to medium-size problems.

The remainder of this article is organized as follows. In Section 2 we present a detailed literature review covering the most important aspects associated with our paper. In Section 3 we provide a brief but precise description of the problem, including a small illustrative example to motivate our study. In Section 4 we provide a mathematical formulation of the problem. In Section 5 we introduce an  $\epsilon$ -constraint method to heuristically solve the problem using our model. In Section 6 we describe a VNS algorithm capable of handling larger problems. In Section 7 we perform a critical analysis of the proposed framework by executing our algorithms on some classical problems from the VRP literature. Finally, Section 8 concludes the paper.

## 2 Literature review

In this section we provide a detailed literature review on the most important aspects associated with our article. Notably, we review the literature on multi-objective vehicle routing problems —including applications where the objectives are not necessarily handled simultaneously, but in a sequential way— and the literature on clustering problems and algorithms. The vast literature on vehicle routing problems and algorithms in single-objective settings is not discussed in this review, for which we refer to Toth and Vigo (2014); Corberán and Laporte (2013).

Vehicle routing algorithms have, since the very early times, included clustering subroutines to reduce the computational burden associated with the routing of the entire problem. The sweep algorithm introduced by

Gillet and Miller (1974) is an example of such decomposition. In the sweep algorithm, customers are grouped according to their proximity using polar coordinates. This can be seen as the ordering in which the nodes would be swept by an imaginary clock hand. Fisher and Jaikumar (1981) proposed a so-called cluster-first-route-second algorithm for vehicle routing problems in which the customers are first grouped according to their proximity solving a generalized assignment problem. For each cluster, a traveling salesman problem (TSP) is then solved. Taillard (1993) uses a similar decomposition in which the clustering of the nodes is performed by solving a minimum spanning forest of the nodes, rooted at the depot. A TSP is then solved for each subtree. Recent heuristics are now less dependent on a pre-clustering of the nodes, mainly because of the additional computational power available that allows the simultaneous routing of several thousands of nodes at once within reasonable time limits. However, some rich vehicle routing problems that are challenging even for medium-size problems still benefit from such decomposition scheme (Miranda-Bront et al., 2016; Kloimüller et al., 2015).

Regarding the simultaneous optimization of multiple objectives in vehicle routing, the surveys of Jozefowicz et al. (2008); Labadie and Prodhon (2014) cover the most important advances in the past 30 years. In most cases, authors have focused into integrating routing optimization (minimization of total routing costs, minimization of makespan, minimization of total traveling distance) with other aspects of the routing. These aspects span from maximizing the customer overall satisfaction (Hong and Park, 1999; Giannikos, 1998; Wang and Li, 2011; Rath and Gutjahr, 2014; Lehuédé et al., 2014), to minimizing the environmental impact of the routing decisions (Demir et al., 2014; Teoh et al., 2016), among several others.

In what concerns the integration of routing and clustering decisions, we are not aware of any article considering these two decisions in an integrated manner. The closest we could get from this is the work of Mourgaya and Vanderbeck (2007), in which the authors introduce a clustering problem that integrates regionalization and route balancing. A routing decisional layer is only included *a posteriori*. Their analysis suggests that by using the clustering provided by this tactical planning the operator can find well balanced and compact solutions, at the expense of larger routing costs.

With respect to the solution of multi-objective optimization problems in a broader context, we can most notably mention the following frameworks. The weighted sum method (Zadeh, 1963; Zionts, 1988) —in which the different objectives are combined to form a single objective— is perhaps the most intuitive method for multi-objective optimization, however is known to suffer from scaling issues that might easily bias the finding of solutions. In lexicographic or hierarchical methods (Waltz, 1967), the different objectives are first sorted, and then each of them is optimized separately in a sequential manner. When optimizing with respect to the  $i$ -th objective, constraints are added to the problem to make sure that the new value of the first  $(i-1)$  objectives remain close in the new solution to the optimal values found in the previous iterations. Goal programming (Charnes et al., 1955) is a technique that aims at finding a compromise of the different objectives, this is a solution that would remain equally close (or far) from some target for each of the objectives. This technique does not aim at finding several solutions, but one which would be a good compromise among them. All these methods assume that the decision maker has declared its preferences with respect to each of the objectives, and so the solutions found reflect that. When no preference is a priori declared, we find tools like the normal boundary intersection method (Das and Dennis, 1998) or the normal constraint method (Messac et al., 2003), which aim at generating several alternate solutions. For comprehensive surveys of multi-objective optimization we refer to Ehrgott and Gandibleux (2000); Marler and Arora (2004); Deb et al. (2016).

The literature on clustering algorithms, criteria and applications is vast. For comprehensive compendiums we refer to Jain and Dubes (1988); Hansen and Jaumard (1997); Jain et al. (1999); Aggarwal and Reddy (2013). Cluster analysis is the task of grouping objects that share similar characteristics, and to separate objects that differ. A clustering of objects is characterized by: 1) a clustering criterion; and 2) a clustering algorithm. The clustering criterion defines the measure used to tell if a group of objects is either compact or not, and at what extent. The minimax diameter criterion used in our article declares a group of objects as compact if the two objects that differ the most in the group are still alike (Aloise and Contardo, 2015). The minimum sum-of-squares criterion declares a group of objects to be compact if the sum of the squared distances to its centroid is small with respect to that of other groupings (Tao et al., 2014; Ordin and Bagirov, 2015). Clustering algorithms are closely associated with the criterion chosen to determine the similarity of

the objects. Unfortunately, the clustering problems associated with these two criteria are NP-hard (Aloise et al., 2009; Garey and Johnson, 2002). Exact algorithms for the minimum sum-of-squares clustering problem can handle a few thousand objects (Aloise et al., 2012) while for the minimax diameter clustering problem this number may reach several hundreds of thousand objects (Duong et al., 2015; Aloise and Contardo, 2015). For larger problems, authors usually resort to heuristics, such as the complete-linkage heuristic for minimax diameter clustering (Defays, 1977), or the k-means algorithm for minimum sum-of-squares clustering (Forgy, 1965).

### 3 Problem description

In this section we provide a brief description of the problem. We are given a set of  $n+1$  nodes  $V = \{0, 1, \dots, n\}$ . The node labeled 0 represents the depot, whereas the remaining nodes represent the customers. The set of customer nodes is denoted  $V^+$ . With each customer  $i \in V^+$  is associated a demand  $d_i > 0$ . We are also given a set of  $K$  identical vehicles, each of which has a capacity equal to  $Q$ . With every pair of nodes  $(i, j), i < j$ , is associated an edge  $\{i, j\}$  with a routing cost  $c_{ij}$ . The VRP with simultaneous optimization of the total routing cost and customer clustering is the problem of routing each of the  $K$  vehicles, so as to visit every customer node exactly once, while respecting the total demand collected by each vehicle on its route. The objectives are: 1) to minimize the total routing cost; and 2) to minimize the maximum distance between any two nodes within the same route. As it may be impossible to find a single solution that minimizes both objectives simultaneously, the real goal of this optimization problem is to find (or at least to approximate) the *Pareto frontier* (Pareto, 1964), i.e., the set of all solutions of the problem that are not dominated by any other solution. A solution  $x$  is said to be dominated by another solution  $y$  if  $y$  is at least as good as  $x$  for all the objectives, being strictly better for at least one of them. To further motivate our discussion, let us introduce the following example. Let us consider a problem containing 13 nodes, whose coordinates are given in Table 1. The demands are assumed unitary and the vehicle capacity is equal to 5 units of demand. The distances are assumed Euclidean.

**Table 1: Node coordinates for the illustrative example**

Node	x-axis	y-axis
0	5	3
1	0	1
2	2	3
3	3	5
4	4	7
5	2	9
6	1	5
7	4	1
8	6	2
9	8	8
10	9	3
11	8	4
12	3	0

In Figure 1 we illustrate 4 possible solutions for the problem, each of which with a different routing cost and/or maximum diameter. Among these 4 solutions, please observe that there is no solution that dominates all the other three, i.e., that is at least as good for all the objectives, being strictly better for at least one of them. Indeed, the solution of best routing cost is plotted in (1c), but its maximum diameter is worse than that of solutions (1a) and (1b). On the other hand, the solutions with the minimum maximum diameter are solutions (1a) and (1b), but none of them has a better routing cost than (1c). Finally, note that solutions (1b) and (1d) could be discarded as they are not interesting at all, being strictly worse than solution (1a) for at least one objective, and not better for the other one. Although this exercise in no way characterizes the Pareto frontier of this problem, it at least assures that solutions (1b) and (1d) cannot belong to it.

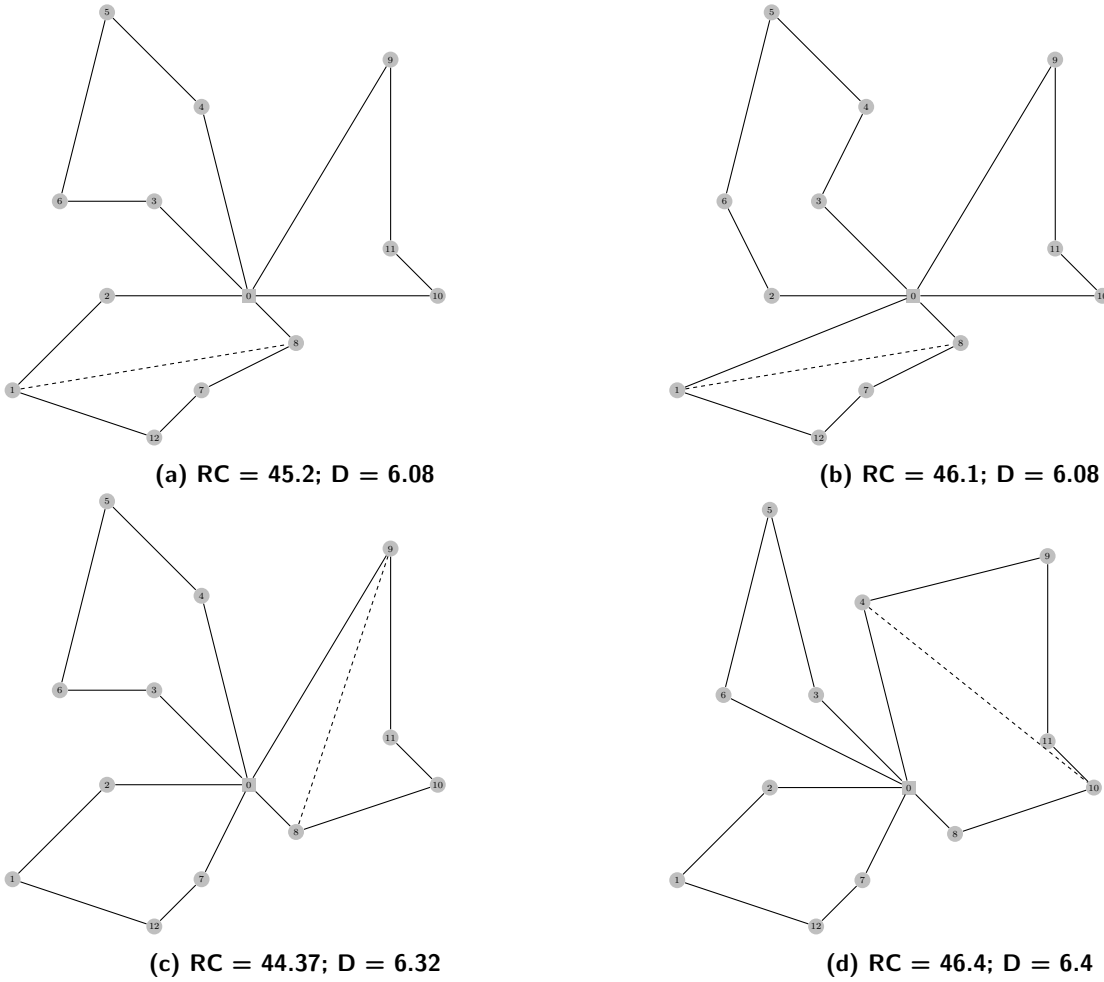


Figure 1: Four different solutions for the illustrative example

## 4 Mathematical formulation

The VRP with simultaneous minimization of the total routing cost and maximum diameter can be formulated as a multi-objective integer-linear problem, as follows. For each edge  $\{i, j\}$ , we let  $x_{ij}$  be an integer variable representing the number of times that edge  $\{i, j\}$  is taken by some vehicle. For depot-to-customer edges  $\{0, i\}, i \in V^+$ , this variable may take integer values between 0 and 2, whereas for customer-to-customer edges it is a binary variable. We also let  $y_{ij}$  be a binary variable taking the value 1 iff nodes  $i$  and  $j$  are serviced by the same vehicle, for any two nodes  $i, j \in V^+, i < j$ . Finally, we let  $D$  be a real-valued variable equal to the maximum diameter among all routes. For notational simplicity, for any set  $S \subset V$ , we denote  $x(\delta(S)) = \sum_{i \in S, j \notin S, i < j} x_{ij} + \sum_{i \in S, j \notin S, i > j} x_{ji}$ , and if in addition  $S \subseteq V^+$ , we also let  $r(S)$  be a lower bound on the number of vehicles needed to service the customers in  $S$ . It is common to define  $r(S) = \lceil \sum_{j \in S} d_j / Q \rceil$ . The following model—derived from the two-index vehicle-flow formulation of the CVRP introduced by Laporte et al. (1985)—is valid for the problem:

$$\min \quad \text{total routing cost} = \sum_{i, j \in V, i < j} c_{ij} x_{ij} \quad (1)$$

$$\min \quad \text{maximum diameter} = D \quad (2)$$

subject to

$$x(\delta(\{i\})) = 2 \quad i \in V^+ \quad (3)$$

$$x(\delta(\{0\})) = 2K \quad (4)$$



$$x(\delta(S)) \geq 2r(S) \quad S \subseteq V^+, |S| \geq 2 \quad (5)$$

$$D \geq c_{ij}y_{ij} \quad i, j \in V^+, i < j \quad (6)$$

$$y_{ij} \geq x_{ij} \quad i, j \in V^+, i < j \quad (7)$$

$$y_{ik} - y_{ij} - y_{jk} + 1 \geq 0 \quad i, j, k \in V^+, i < j < k \quad (8)$$

$$y_{ij} - y_{ik} - y_{jk} + 1 \geq 0 \quad i, j, k \in V^+, i < j < k \quad (9)$$

$$y_{jk} - y_{ij} - y_{ik} + 1 \geq 0 \quad i, j, k \in V^+, i < j < k \quad (10)$$

$$x_{0j} \in \{0, 1, 2\} \quad j \in V^+ \quad (11)$$

$$x_{ij} \in \{0, 1\} \quad i, j \in V^+, i < j \quad (12)$$

$$y_{ij} \in \{0, 1\} \quad i, j \in V^+, i < j. \quad (13)$$

In this problem, the two objectives (1)–(2) seek to simultaneously minimize the total routing cost and the maximum diameter, respectively. Constraints (3)–(5) are classical VRP constraints: degree, fleet size and capacity constraints, respectively. Constraints (6) define the maximum diameter. Constraints (7)–(10) impose that customers serviced by the same vehicle must belong to the same cluster. Finally, constraints (11)–(13) express the integer nature of the variables  $x$  and  $y$ .

## 5 An $\epsilon$ -constraint method

In this section we describe an  $\epsilon$ -constraint method (Mavrotas, 2009) for the solution of our bi-objective problem. The  $\epsilon$ -constraint method is an iterative method that at every iteration generates one Pareto-optimal solution. The choice of the  $\epsilon$ -constraint method is supported on the two following remarks. On the one hand, it is a known algorithm that has been shown to behave well with other variants of multi-objective vehicle routing problems (Bérubé et al., 2009) to find or at least to approximate the Pareto frontier. On the other hand, it is a method that is friendly with mathematical programming-based algorithms. Indeed, the implementation of an  $\epsilon$ -constraint method as a mathematical program is straightforward as it only requires the addition of constraints to a model, which can be handled efficiently by any state-of-the-art solver.

Let us denote, for a given target diameter  $\delta$  and for a given  $\epsilon > 0$ ,  $P(\delta)$  as the following single-objective, linear-integer problem derived from (1)–(13):

$$\min \quad \text{total routing cost} = \sum_{i,j \in V, i < j} c_{ij}x_{ij} \quad (14)$$

subject to

$$(3) - (13) \quad (15)$$

$$D \leq \delta - \epsilon. \quad (16)$$

This problem, when solved to optimality, finds a solution of minimum routing cost among those whose maximum diameter is strictly lower than  $\delta$  (given that  $\epsilon$  is sufficiently small so as to not cut any solution of smaller routing cost and whose maximum diameter lies between  $\delta - \epsilon$  and  $\delta$ ). In our problem, we begin with  $\delta \leftarrow +\infty$  and iteratively refine the diameter so as to search for problems more and more compact each time. In addition, we set to zero all variables  $x_{ij}, y_{ij}$  such that  $c_{ij} \geq \delta$ , thus reducing the computational burden associated with the solution of this linear-integer problem. If we call the optimal solution of this problem  $x$ , and we let  $RC(x), D(x)$  be the routing cost and the maximum diameter of that solution, we set  $\delta \leftarrow D(x)$  to solve the problem again. This is done until no solutions can be found by the algorithm (either because it is proven unfeasible or because the solution process reaches some predefined time limit). The following pseudo-code illustrates our method.

In this pseudo-code, routine  $Solve(P(\delta))$  provides a solution to problem  $P(\delta)$  or  $\emptyset$  if no such solution can be found.

---

**Algorithm 1**  $\epsilon$ -constraint method for multi-objective VRP

---

```

Set  $\delta \leftarrow +\infty, \mathcal{S} \leftarrow \emptyset$ 
while  $x \leftarrow \text{Solve}(P(\delta)) \neq \emptyset$  do
  Set  $\mathcal{S} \leftarrow \mathcal{S} \cup \{x\}$ 
  Set  $\delta \leftarrow D(x)$ 
end while
return  $\mathcal{S}$ 

```

---

## 6 Variable neighborhood search

In this section we introduce a variable neighborhood search (VNS) algorithm for multi-objective vehicle routing problems, especially tailored for the solution of the combined optimization of routing and clustering decisions. Our algorithm is based on the iterated local search (ILS) algorithm developed by Penna et al. (2013) for the VRP with heterogeneous fleet (HFVRP). We utilize the same 11 neighborhoods used in their algorithm, but embedded within a VNS instead of an ILS as the original method. Our method uses as a basis a single-objective VNS algorithm, fully described in the next paragraph, for routing cost minimization. Then we embed this VNS into an  $\epsilon$ -constraint method similar to that of Section 5.

### 6.1 A VNS for routing cost minimization

A high level description of the VNS algorithm for the minimization of the routing cost is as follows. The neighborhoods  $(N_i)_{i=1}^{11}$  are first sorted according to the same ordering proposed by Penna et al. (2013). For a given solution  $x$ ,  $N_i(x)$  is the set of neighboring solutions of  $x$  that can be reached by performing some simple move on  $x$ . A starting solution  $x$  is found using the Clarke & Wright savings heuristic (CWSH, Clarke and Wright (1964)). Starting from  $x$  and  $k \leftarrow 1$ , first a shaking is performed. In our implementation, the shaking is uniform across the different neighborhoods and consists in performing, three consecutive times, the swapping of any two random customers. Let us call this procedure  $Shake(x)$ , and let  $x'$  denote the resulting solution. Then we perform local search on  $x'$  to find the solution  $x'' \in N_k(x')$  of lowest routing cost. Let us denote  $LocalSearch(x', N_k)$  this procedure. If the routing cost  $RC(x'')$  associated with  $x''$  is greater than or equal to that of  $x$ , we set  $k \leftarrow k + 1$  and the process is repeated. Otherwise the solution  $x$  is updated to  $x''$  and the whole process is restarted. This is repeated at most  $\Delta$  times without success, where  $\Delta > 0$  is a predefined parameter. The following pseudo-code illustrates our VNS for routing cost minimization.

---

**Algorithm 2** VNS algorithm for VRP with routing cost minimization

---

```

 $x \leftarrow CWSH$ 
 $l \leftarrow 1$ 
while  $l \leq \Delta$  do
   $k \leftarrow 1$ 
  while  $k \leq 11$  do
     $x' \leftarrow Shake(x)$ 
     $x'' \leftarrow LocalSearch(x', N_k)$ 
    if  $RC(x'') < RC(x)$  then
       $x \leftarrow x''$ 
       $k \leftarrow 1$ 
       $l \leftarrow 1$ 
    else
       $k \leftarrow k + 1$ 
    end if
  end while
   $l \leftarrow l + 1$ 
end while
return  $x$ 

```

---

### 6.2 A VNS-based $\epsilon$ -constraint method

Our algorithm for the multi-objective VRP uses the same  $\epsilon$ -constraint method described in Section 5 but replaces routine  $Solve(P(\delta))$  by a method called  $VNS(\delta)$ , which corresponds to the VNS described in Section 6.1, but in which an additional parameter  $\delta$  is used to deem unfeasible every solution whose maximum

diameter is of  $\delta$  or larger (pretty much as problem  $P(\delta)$  described in Section 5). As there may be no solution whose maximum diameter is strictly lower than  $\delta$ ,  $VNS(\delta)$  may return an empty solution. For a given solution  $x$ , we denote by  $D(x)$  its maximum diameter. The following pseudo-code illustrates our VNS-based  $\epsilon$ -constraint method:

---

**Algorithm 3**  $\epsilon$ -constraint VNS algorithm for multi-objective VRP
 

---

```

Set  $\delta \leftarrow +\infty, S \leftarrow \emptyset$ 
while  $x \leftarrow VNS(\delta) \neq \emptyset$  do
  Set  $S \leftarrow S \cup \{x\}$ 
  Set  $\delta \leftarrow D(x)$ 
end while
return  $S$ 

```

---

## 7 Computational experience

In this section, we report and analyze the results obtained by our two methods on some classical problems from the CVRP literature (available from <http://neo.lcc.uma.es/vrp>), namely the so-called instances A-B-E-P. Both algorithms (the model-based and the VNS-based  $\epsilon$ -constraint methods) have been coded in C++ using the GNU g++ compiler v5, running under a Linux machine with 4 GB of RAM, with an Intel Core i3-2310M @ 2.1 GHz. We used CPLEX 12.6.1 as multi-purpose optimization solver, and the CVRPSEP C library (Lysgaard, 2003) to separate the rounded capacity inequalities (5). The sole parameter  $\Delta$  in Algorithm 2 was set to 30 after calibration experiments.

In Table 2 we present detailed results on the performance of the model-based  $\epsilon$ -constraint method introduced. We restrict our analysis to the problems that our model could handle within a time limit of 1 hour. In this table we report, for each solution found, its routing cost (under column labeled RC) and maximum diameter (under column labeled D). We also report, under column labeled T, the CPU time required for our algorithm to complete. Finally, for comparison purposes, we also report, under column labeled T VNS, the CPU time required for the VNS to complete. Please note that our VNS, on all of these small instances, was capable of finding the same approximate Pareto frontier as the model-based  $\epsilon$ -constraint method. Also, please note that we omit the results for some easy problems for which only a single solution was found. This is the case for problems P-n19-k2, E-n30-k3.

**Table 2: Detailed results for the model-based  $\epsilon$ -constraint method**

Instance	RC	D	T (s)	T VNS (s)
E-n22-k4	375	40	15.3	12.2
	395	38		
E-n23-k3	569	97	62.4	64.8
	595	96		
	600	81		
P-n16-k8	450	19	5.6	1.2
	456	16		
P-n20-k2	216	37	25.0	11.5
	218	36		
P-n22-k8	603	39	2,766.6	67.4
	604	36		
	605	34		

As the results show, the bi-objective setting might be capable of finding multiple non-dominated solutions even in the case of small-size problems. The number of solutions in the frontier is usually small and contains a handful of them. On the other hand, please note that the VNS-based  $\epsilon$ -constraint method proves efficient to find the same Pareto frontier as our model, however it is much more robust with respect to the CPU time required to complete this task.

In Tables 3–6 we describe the results provided by the VNS algorithm, i.e., the solutions belonging to the approximated Pareto frontier as well as the CPU time taken by the method (in seconds). For several of the

problems included in our experiments, we found that the objectives were not really conflicting and thus the solution of minimum routing cost and that of minimum maximum diameter as found by the VNS coincided. These instances are: A-n33-k6, A-n38-k5, A-n39-k5, A-n39-k6, A-n45-k6, A-n46-k7, A-n53-k7, A-n54-k7, A-n55-k9, A-n60-k9, A-n61-k9, A-n62-k8, A-n63-k9, A-n63-k10, A-n65-k9, A-n69-k9, A-n80-k10, B-n31-k5, B-n39-k5, B-n41-k6, B-n45-k5, B-n50-k7, B-n50-k8, B-n51-k7, B-n63-k10, B-n66-k9, B-n67-k10, B-n78-k10, E-n30-k3, E-n51-k5, E-n76-k7, E-n76-k8, E-n76-k10, E-n76-k14, P-n19-k2, P-n21-k2, P-n22-k2, P-n40-k5, P-n45-k5, P-n50-k8, P-n51-k10, P-n55-k7, P-n55-k15, P-n60-k10, P-n60-k15, P-n65-k10, P-n70-k10, P-n76-k4, P-n76-k5. We do not report the results obtained for those instances.

An *a posteriori* analysis of the results obtained after executing our algorithm on those instances reveals, however, the following pattern. We observe that the two customers that entail and define the maximum diameter in the solution of minimum routing cost are usually far from the remaining clusters (or routes), in such a way that moving any (or both) of those two customers to a different route would necessarily entail an increase in the maximum diameter. This can somehow be observed *a priori* on problems with very clustered customer nodes, but it is not a necessary condition for it to happen. To exemplify this remark, please observe the optimal routing solution for instance P-n19-k2 as depicted in Figure 2. Nodes 11 and 17 are those that define the maximum diameter. If any of these two nodes (or both) were inserted into the other route, their distance to node 13 would necessarily produce an increase in the maximum diameter.

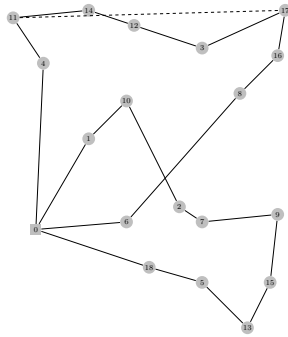


Figure 2: Optimal routing and clustering solution for Problem P-n19-k2

From the results reported in Tables 3–6 we can extract several remarks. First and foremost, we can observe that the Pareto frontier can contain several undominated solutions, but rarely more than a handful of them. Second, several solutions being almost identical with respect to one of the objectives, present a much higher variance with respect to the other. Look, for instance, at instance B-n45-k6. The two solutions with lowest routing costs share almost identical maximum diameters (less than a 2% difference), but differ in more than a 5% in terms of the total routing cost. On the other hand, the opposite can also be observed. Look, for instance, to problem E-n22-k4, where the only two solutions in the Pareto frontier are less than a 0.5% apart in terms of routing cost, but are more than a 12% apart in terms of maximum diameter. Now, look at the two solutions obtained for instance B-n57-k7. They only differ in less than a 2% in terms of their maximum diameter, but present more than a 8% of difference in terms of routing cost. This means that, by choosing not the best clustering, but the second best, the compactness of the solution is almost maintained, and a saving of more than a 8% can be observed in the routing costs. This type of solution would be unavailable to the decision planner if both objectives were optimized in sequence.

**Table 3: Detailed results on instances of set A**

Instance	RC	D	T
A-n32-k5	784	78	325
	796	74	
	800	72	
	828	58	
A-n33-k5	661	65	253
	671	64	
	683	59	
	687	51	
A-n34-k5	778	70	190
	786	62	
A-n36-k5	799	58	316
	805	55	
A-n37-k5	669	83	959
	673	71	
	678	69	
	685	64	
	687	59	
	690	56	
A-n37-k6	949	59	115
	966	54	
A-n44-k6	937	68	416
	942	63	
	975	61	
A-n45-k7	1146	79	684
	1148	74	
	1161	64	
A-n48-k7	1073	56	1,635
	1179	54	
A-n64-k9	1401	82	10,445
	1454	68	
	1468	58	

**Table 4: Detailed results on instances of set B**

Instance	RC	D	T
B-n34-k5	788	80	275
	790	76	
B-n35-k5	955	103	384
	989	100	
	1000	88	
B-n38-k6	805	66	629
	864	64	
	865	61	
B-n43-k6	742	80	1,882
	760	67	
	770	63	
	804	61	
	805	59	
	807	58	
B-n44-k7	909	80	1,192
	957	66	
	964	63	
B-n45-k6	678	61	1,267
	716	60	
	719	49	
	758	47	
B-n52-k7	747	74	2,226
	765	68	
B-n56-k7	707	86	2,544
	773	75	
B-n57-k7	1153	61	3,388
	1254	60	
B-n57-k9	1598	61	2,868
	1695	53	
B-n64-k9	861	74	6,700
	915	65	
	916	63	
B-n68-k9	1272	76	13,242
	1324	75	
	1327	66	
	1328	65	

## 8 Concluding remarks

In this article, we have introduced a bi-objective approach for a vehicle routing problem integrating operational planning into tactical clustering decisions. The problem has been formally defined and a mathematical formulation has been provided. While the mathematical formulation cannot handle problems containing more than a few tens of nodes, we have introduced a VNS algorithm capable of handling larger problems. We have executed the VNS on a set of classic instances from the vehicle routing literature, and analyzed the potential impact of using our decision tool instead of applying the two objectives in a sequential manner. We have shown that our multi-objective approach can provide solutions that would otherwise be missed by a sequential optimization, thus forbidding the decision maker from benefiting of alternate solutions.

As a matter of future research, we have detected several potential avenues. First, we believe that richer vehicle routing problems (as vehicle routing with time windows, synchronization constraints, pickup and delivery, to name just a few) could benefit from similar modeling techniques, which would in turn allow decision makers to take potentially better decisions. Second, we also believe that authors could investigate the potential benefit of considering other clustering criteria within the same framework, as for instance minimum sum-of-squares clustering. Finally, we also believe that there is room for improvement of our methods. For the model, we believe that the addition of cutting planes or perhaps an entirely new modeling

**Table 5: Detailed results on instances of set E**

Instance	RC	D	T
E-n22-k4	420	56	9
	422	49	
E-n23-k3	570	97	38
	595	96	
	600	81	
E-n33-k4	835	86	253
	860	67	
	861	57	

**Table 6: Detailed results on instances of set P**

Instance	RC	D	T
P-n16-k8	450	19	1
	456	16	
P-n20-k2	216	37	9
	218	36	
P-n22-k8	603	39	12
	604	36	
	605	34	
P-n23-k8	529	30	10
	530	22	
P-n50-k7	554	42	1,568
	596	39	
P-n50-k10	696	42	1,422
	708	41	
	709	29	
P-n55-k8	576	42	3,366
	613	40	
P-n55-k10	694	36	3,396
	708	32	

paradigm (a set-partitioning formulation most notably) could help handling much larger problems. For the VNS, we believe that investigating the potential benefit of granularity (Toth and Vigo, 2003) could lead to a reduction of the CPU times without damaging the overall performance of our algorithm.

## References

- C. C. Aggarwal and C. K. Reddy. Data clustering: algorithms and applications. Chapman and Hall/CRC, 2013.
- D. Aloise and C. Contardo. An iterative algorithm for the solution of very large-scale diameter clustering problems. Cahiers du GERAD, GERAD, 2015.
- D. Aloise, A. Deshpande, P. Hansen, and P. Popat. Np-hardness of Euclidean sum-of-squares clustering. Machine learning, 75(2):245–248, 2009.
- D. Aloise, P. Hansen, and L. Liberti. An improved column generation algorithm for minimum sum-of-squares clustering. Mathematical Programming, 131(1-2):195–220, 2012.
- J.-F. Bérubé, M. Gendreau, and J.-Y. Potvin. An exact  $\epsilon$ -constraint method for bi-objective combinatorial optimization problems: Application to the traveling salesman problem with profits. European Journal of Operational Research, 194(1):39–50, 2009.
- A. Charnes, W. W. Cooper, and R. O. Ferguson. Optimal estimation of executive compensation by linear programming. Management science, 1(2):138–151, 1955.
- G. Clarke and J. W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. Operations research, 12(4):568–581, 1964.
- Á. Corberán and G. Laporte. Arc routing: problems, methods, and applications. SIAM, 2013.
- G. Dantzig, R. Fulkerson, and S. Johnson. Solution of a large-scale traveling-salesman problem. Journal of the operations research society of America, 2(4):393–410, 1954.
- G. B. Dantzig and J. H. Ramser. The truck dispatching problem. Management science, 6(1):80–91, 1959.
- I. Das and J. E. Dennis. Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. SIAM Journal on Optimization, 8(3):631–657, 1998.
- K. Deb, K. Sindhya, and J. Hakanen. Multi-objective optimization. In Decision Sciences: Theory and Practice, pages 145–184. CRC Press, 2016.
- D. Defays. An efficient algorithm for a complete link method. The Computer Journal, 20(4):364–366, 1977.
- E. Demir, T. Bektaş, and G. Laporte. The bi-objective pollution-routing problem. European Journal of Operational Research, 232(3):464–478, 2014.
- K.-C. Duong, C. Vrain, et al. Constrained clustering by constraint programming. Artificial Intelligence, 2015.

- M. Ehrgott and X. Gandibleux. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR-Spektrum*, 22(4):425–460, 2000.
- M. L. Fisher and R. Jaikumar. A generalized assignment heuristic for vehicle routing. *Networks*, 11:109–124, 1981.
- E. W. Forgy. Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *Biometrics*, 21:768–769, 1965.
- M. R. Garey and D. S. Johnson. *Computers and intractability*, volume 29. wh freeman New York, 2002.
- I. Giannikos. A multiobjective programming model for locating treatment sites and routing hazardous wastes. *European Journal of Operational Research*, 104(2):333–342, 1998.
- B. E. Gillet and L. R. Miller. A heuristic algorithm for the vehicle dispatch problem. *Operations Research*, 22:340–349, 1974.
- P. Hansen and B. Jaumard. Cluster analysis and mathematical programming. *Mathematical programming*, 79(1-3):191–215, 1997.
- D. Haugland, S. C. Ho, and G. Laporte. Designing delivery districts for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 180(3):997–1010, 2007.
- S.-C. Hong and Y.-B. Park. A heuristic for a bi-objective vehicle routing with time window constraints. *International Journal of Production Economics*, 62:249–258, 1999.
- A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.
- A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323, 1999.
- N. Jozefowiez, F. Semet, and E.-G. Talbi. Multi-objective vehicle routing problems. *European Journal of Operational Research*, 189(2):293–309, 2008.
- C. Kloimüller, P. Papazek, B. Hu, and G. R. Raidl. *A Cluster-First Route-Second Approach for Balancing Bicycle Sharing Systems*, pages 439–446. Springer International Publishing, Cham, 2015.
- N. Labadie and C. Prodhon. A survey on multi-criteria analysis in logistics: Focus on vehicle routing problems. In *Applications of Multi-Criteria and Game Theory Approaches*, pages 3–29. Springer, 2014.
- G. Laporte, Y. Nobert, and M. Desrochers. Optimal routing under capacity and distance restrictions. *Operations Research*, 33(5):1050–1073, 1985.
- F. Lehuédé, R. Masson, S. N. Parragh, O. Péton, and F. Tricoire. A multi-criteria large neighbourhood search for the transportation of disabled people. *Journal of the Operational Research Society*, 65(7):983–1000, 2014.
- J. K. Lenstra and A. Kan. Complexity of vehicle routing and scheduling problems. *Networks*, 11(2):221–227, 1981.
- J. Lysgaard. CVRSEP: A package of separation routines for the capacitated vehicle routing problem. 2003. URL <http://www.asb.dk/~lys>.
- R. T. Marler and J. S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26(6):369–395, 2004.
- G. Mavrotas. Effective implementation of the  $\varepsilon$ -constraint method in multi-objective mathematical programming problems. *Applied mathematics and computation*, 213(2):455–465, 2009.
- A. Messac, A. Ismail-Yahaya, and C. A. Mattson. The normalized normal constraint method for generating the pareto frontier. *Structural and multidisciplinary optimization*, 25(2):86–98, 2003.
- J. J. Miranda-Bront, B. Curcio, I. Méndez-Díaz, A. Montero, F. Pousa, and P. Zabala. A cluster-first route-second approach for the swap body vehicle routing problem. *Annals of Operations Research*, pages 1–22, 2016.
- M. Mourgaya and F. Vanderbeck. Column generation based heuristic for tactical planning in multi-period vehicle routing. *European Journal of Operational Research*, 183(3):1028–1041, 2007.
- B. Ordin and A. M. Bagirov. A heuristic algorithm for solving the minimum sum-of-squares clustering problems. *Journal of Global Optimization*, 61(2):341–361, 2015.
- V. Pareto. *Cours d'économie politique*, volume 1. Librairie Droz, 1964.
- P. H. V. Penna, A. Subramanian, and L. S. Ochi. An iterated local search heuristic for the heterogeneous fleet vehicle routing problem. *Journal of Heuristics*, 19(2):201–232, 2013.
- S. Rath and W. J. Gutjahr. A math-heuristic for the warehouse location–routing problem in disaster relief. *Computers & Operations Research*, 42:25–39, 2014.
- É. Taillard. Parallel iterative search methods for vehicle routing problems. *Networks*, 23:661–673, 1993.
- P. D. Tao et al. New and efficient dca based algorithms for minimum sum-of-squares clustering. *Pattern Recognition*, 47(1):388–401, 2014.
- B. E. Teoh, S. Ponnambalam, and N. Subramanian. Data driven safe vehicle routing analytics: a differential evolution algorithm to reduce CO<sub>2</sub> emissions and hazardous risks. *Annals of Operations Research*, pages 1–24, 2016.

- 
- P. Toth and D. Vigo. The granular tabu search and its application to the vehicle-routing problem. *Informs Journal on Computing*, 15(4):333–346, 2003.
- P. Toth and D. Vigo. *Vehicle routing: problems, methods, and applications*. SIAM, 2014.
- F. Waltz. An engineering approach: hierarchical optimization criteria. *IEEE Transactions on Automatic Control*, 12(2):179–180, 1967.
- C.-H. Wang and C.-H. Li. Optimization of an established multi-objective delivering problem by an improved hybrid algorithm. *Expert Systems with Applications*, 38(4):4361–4367, 2011.
- L. Zadeh. Optimality and non-scalar-valued performance criteria. *IEEE transactions on Automatic Control*, 8(1):59–60, 1963.
- H. Zhong, R. W. Hall, and M. Dessouky. Territory planning and vehicle dispatching with driver learning. *Transportation Science*, 41:74–89, 2007.
- S. Zionts. Multiple criteria mathematical programming: An updated overview and several approaches. In *Mathematical models for decision support*, pages 135–167. Springer, 1988.