# Robust MILP formulations for the two-stage weighted vertex p-center problem

Cristian Durán Mateluna, Zacharie Alès, Natalia Jorquera-Bravo, Sourour Elloumi

# Robust MILP formulations for the two-stage weighted vertex $p$-center problem

Cristian Durán-Mateluna[a,b,c,*], Natalia Jorquera-Bravo[a,b,c], Zacharie Alès[a,b], Sourour Elloumi[a,b]

[a] *UMA, ENSTA Paris, Institut Polytechnique de Paris, Palaiseau, France.*
[b] *CEDRIC, Conservatoire National des Arts et Métiers, Paris, France.*
[c] *LDSPS, Department of Industrial Engineering, University of Santiago of Chile, Santiago, Chile.*

---

**Abstract**

The weighted vertex $p$-center problem ($PCP$) consists of locating $p$ facilities among a set of potential sites such that the maximum weighted distance from any client to its closest open facility is minimized. This paper studies the exact resolution of the two-stage robust weighted vertex $p$-center problem ($RPCP_2$). In this problem, the opening of the centers is fixed in the first stage while the client allocations are recourse decisions fixed once the uncertainty is revealed. The problem uncertainty comes from both the nodal demands and the edge lengths. It is modeled by box uncertainty sets. We introduce three different robust reformulations based on MILPs from the literature. We prove that considering a finite subset of scenarios is sufficient to obtain an optimal solution of ($RPCP_2$). We leverage this result to introduce a *column-and-constraint* generation algorithm and a *branch-and-cut* algorithm to efficiently solve this problem optimally. We highlight how these algorithms can be adapted to solve, for the first time to optimality, the single-stage problem ($RPCP_1$) which is obtained when no recourse is considered. We present a numerical study to compare the performance of these formulations on randomly generated instances and a case study from the literature.

*Keywords:* discrete location; $p$-center problem; robust MILP formulations; *column-and-constraint* generation algorithm; *branch-and-cut* algorithm

---

## 1. Introduction

The vertex $p$-center problem ($PCP$) is one of the most studied facility location problems in the literature. It consists of installing $p$ centers out of $m$ available sites, assigning $n$ clients to these $p$ centers, in order to minimize the *radius* which corresponds to the maximum distance between a

*Corresponding author
*Email address:* `cristian.duran@ensta-paris.fr` (Cristian Durán-Mateluna)

client and its closest installed center. Likewise, a $p$-center is *weighted* if a demand is associated to each client. The $p$-center problem under uncertainty is well studied in the literature (Çalık et al. (2019)). This problem arises when parameters, such as demands or distances, vary across time or when their exact value is uncertain. The uncertainty is generally represented by parameters which can take any value in an uncertainty set. Each element of the uncertainty set is called a *scenario*. The most classical sets are the box, the ellipsoid and the budgeted uncertainty sets (see e.g., Ben-Tal et al. (2009); Bertsimas & Sim (2004); Du & Zhou (2018); Paul & Wang (2019)).

Two major approaches have been developed to address uncertainty: stochastic optimization and robust optimization. Stochastic optimization requires that a discrete or continuous probabilistic distribution of the uncertain parameters is known, and tries to have the best value on average. Robust optimization tries to protect itself against the worst case (Ben-Tal et al. (2009)). In this sense, stochastic optimization is more relevant in the context of repeated experiments while robust optimization is more suitable when one wishes at all costs to avoid the worst case, for example, when human lives are at stake.

The incorporation of uncertainty in $(PCP)$ has important applications in emergency logistics problems. Two approaches can be considered depending on whether the client allocations to the centers are made before (see e.g, Averbakh & Berman (1997), Lu (2013)) or after (see e.g, Du et al. (2020); Demange et al. (2020)) the uncertainty is revealed. The first case corresponds to single-stage problems while the second case leads to two-stage problems in which the clients allocations are recourse variables. In this context, most of works has been focused on the investigation of integer programming modeling and heuristic resolution approaches (see e.g., Baron et al. (2011); Hasani & Mokhtari (2018); Paul & Wang (2015); Trivedi & Singh (2017, 2019)).

### *1.1. Contribution and Outline*

We study the exact resolution of the two-stage robust weighted $p$-center problem $(RPCP_2)$ in which the uncertainty on the node demands and the travel times are modeled by box uncertainty sets. We present robust reformulations of this problem based on three MILP formulations of $(PCP)$. We prove that a finite subset of scenarios from the infinite box uncertainty set can be considered without losing optimality. We use this result to propose a *column-and-constraint* generation algorithm $(C\&CG)$ and a *branch-and-cut* algorithm $(B\&C)$ for the exact resolution of $(RPCP_2)$. We highlight how these algorithms can be adapted to the single-stage problem $(RPCP_1)$ for which no exact resolution method has been previously introduced. Finally, we show their efficiency on randomly generated instances and on the case study inspired from an earthquake that hit central Taiwan in 1999 presented in Lu (2013).

The rest of the paper is organized as follows. Section 2 presents the literature review of the deterministic and robust versions of the $(PCP)$. Section 3 describes the robust two-stage problem $(RPCP_2)$, proves how to reduce the number of scenarios, and introduces our three MILP formulations as well as the $(C\&CG)$ and $(B\&C)$ algorithms. Section 4 presents the computational results. In Section 5 we draw conclusions together with research perspectives.

## 2. Literature review

The $p$-center problem was introduced by Hakimi (1965), who presented and solved the absolute 1-center problem on a graph. In the absolute $p$-center problem, the center can be located either on the edges or the vertices of the graph. Later, Minieka (1970) extended the problem to the case $p > 1$ and proposed a method to restrict the continuous set of candidate centers to a discrete set of points, without losing optimality. Since then, several formulations, resolution methods, and variants of this problem have been presented. We refer to Çalık et al. (2019) for a more exhaustive review of applications and resolution methods of the $p$-center problem. In this section, we first focus on the deterministic $p$-center problem and then on its robust counterparts.

### *2.1. MILP formulations of the deterministic weighted p-center*

Let $U$ be the set of candidate centers or facility nodes, and $V$ be the set of clients or demand nodes. The travel time between any possible pair of demand node $i$ in $V$ and center $j$ in $U$ is denoted by $t_{ij}$. Each demand node $i \in V$ faces a demand $\xi_i$ and must be assigned to a single center $j$. In the following formulations, no client demands were originally considered. They only contained a distance $d_{ij}$ between each client $i$ and center $j$. Nevertheless, to model the weighted $p$-center $d_{ij}$ can equivalently be replaced by the product of demand $\xi_i$ and travel time $t_{ij}$.

The classical formulation of the $p$-center problem was presented in Daskin (1996). This model considers binary variables $x_j$ equal to 1 if and only if center $j \in U$ is opened, binary variables $y_{ij}$ equal to 1 if and only if the demand node $i$ is assigned to center $j$, and a variable $z$ equals to the radius:

$$
(F1): \quad
\begin{aligned}
& \text{Minimize } z, && && (1)\\
\text{s.t.} \quad & z \geq \sum_{j \in U} \xi_i t_{ij} y_{ij}, && i \in V, && (2)\\
& \sum_{j \in U} y_{ij} = 1, && i \in V, && (3)\\
& y_{ij} \leq x_j, && i \in V, j \in U, && (4)\\
& \sum_{j \in U} x_j = p, && && (5)\\
& x_j \in \{0,1\}, && j \in U, && (6)\\
& y_{ij} \in \{0,1\}, && i \in V, j \in U. && (7)
\end{aligned}
$$

Constraints (3) ensure that each client is assigned to only one center and Constraints (4) ensure that no client is assigned to a center that is not open. Constraint (5) fixes the number of open centers to $p$. Constraints (2) indicate that the distances between each client and its nearest center are less than the radius. We minimize the radius through the objective function (1).

An alternative formulation was introduced in Elloumi et al. (2004). This formulation proposes to associate one variable to each weighted distance in the considered instance. Let $D^0 < D^1 < \ldots < D^K$ be the distinct weighted distances and let $\mathcal{K}$ be the set $\{1, 2, \ldots, K\}$. The radius variable $z$ and the assignment variables $y$ are replaced by variables $z^k$ with $k \in \mathcal{K}$ equal to 1 if and only if the radius is greater than or equal do $D^k$:

$$
(F2) \quad
\begin{aligned}
& \min \; D^0 + \sum_{k \in \mathcal{K}} \left(D^k - D^{k-1}\right) z^k && && (8)\\
\text{s.t.} \quad & z^k + \sum_{j : \xi_i t_{ij} < D^k} x_j \geq 1, && i \in V, k \in \mathcal{K}, && (9)\\
& \sum_{j \in U} x_j = p, && && \\
& x_j \in \{0,1\}, && j \in U, && \\
& z^k \in \{0,1\}, && k \in \mathcal{K}. && (10)
\end{aligned}
$$

Constraints (9) indicate that a client is covered by a center at a distance less than $D^k$, or that the radius is greater than or equal to $D^k$. Thus, in the objective (8), if $z^k = 1$, $(D^k - D^{k-1})$ is added to the radius. Elloumi et al. (2004) show that $(F2)$ provides a continuous relaxation that dominates that of $(F1)$. Calik & Tansel (2013) introduced another formulation deduced from $(F2)$ by a change of variables. It contains, for all $k \in \mathcal{K}$, a binary variable $u^k$ equal to 1 if and only if the optimal radius is equal to $D^k$ (i.e., $u^k = z^k - z^{k+1}$). Ales & Elloumi (2018) present a formulation that improves the resolution performance of the previous formulation $(F2)$. They add the following family of valid inequalities:

$$z^k \geq z^{k+1} \qquad \forall\, k \in \{1, 2, \dots, K-1\} \tag{11}$$

and show that Constraints (11) enable to remove a significant number of now redundant Constraints (9). Let $N_i^k$ be the set of facilities located at less than $D^k$ from client $i$. Note that, $N_i^k$ is equal to $N_i^{k+1}$ if and only if there is no facility at distance $D^k$ from client $i$. Let $S_i$ be the set of indices $k \in \{1, ..., K-1\}$ such that $N_{ik}$ is different from $N_{ik+1}$. Constraints (9) can be replaced by:

$$z^k + \sum_{j:\xi_i t_{ij} < D^k} x_j \geq 1 \qquad i \in V, k \in S_i \cup K \tag{12}$$

Ales & Elloumi (2018) also presented another compact formulation, which contains less variables and constraints than $(F2)$. They replace the $K$ binary variables $z_k$ with a unique integer variable $r$ which represents the index of the optimal radius:

$$\begin{aligned}
& & \min r & & & (13)\\
& \text{s.t.} & r + k \sum_{j:\xi_i t_{ij} < D^k} x_j \geq k, & & i \in V, k \in \mathcal{K}, & \quad (14)\\
(F3) & & \sum_{j \in U} x_j = p, & & & \\
& & x_j \in \{0,1\}, & & j \in U, & \\
& & r \geq 0. & & & (15)
\end{aligned}$$

Constraints (14) play a role similar to that of Constraints (9). This formulation $(F3)$ provides a weaker linear relaxation than the previous formulations. However, as we will see in Section 4, it is particularly useful for the exact resolution of $(RPCP_2)$. Note that every feasible solution $(x, z)$ of $(F2)$ and $(x, r)$ of $(F3)$ can be easily transformed into a feasible solution $(x, y)$ of $(F1)$, by assigning each client to his nearest installed center in the feasible solution $x$.

### 2.2. Uncertainty representation and resolution methods

The positioning of facilities is a long-term decision which takes into account parameters such as client demands or distances between clients and facilities. Since these parameters are likely to vary, several models have been developed to study facility location problems under uncertainty. The stochastic optimization and robust optimization are the two main approaches to address uncertainty. We refer to Snyder (2006) and Correia & Saldanha-da Gama (2019) for a review of the literature on stochastic and robust facility location problems.

Box, budgeted, ellipsoidal and discrete uncertainty sets are commonly considered (see e.g. Ben-Tal et al. (2009); Baron et al. (2011); Du & Zhou (2018); Paul & Wang (2019, 2015); Snyder (2006)). Since most robust facility location problems are harder to solve than their deterministic counterparts, heuristic approaches have taken precedence over exact resolution methods (Correia & Saldanha-da Gama (2019)). Most robust facility location problems based on discrete uncertainty sets deal with generalizations of the $p$-median problem, focusing exclusively on analytical results or approximated polynomial-time algorithms (see e.g. Serra & Marianov (1998); Hasani & Mokhtari (2018)).

The presence or absence of recourse variables, the variables which are fixed once the uncertainty is revealed, has a great influence on the mathematical formulation of the problem. A single-stage problem can be considered when there is no recourse variables while a two-stage is required otherwise. Two-stage models are usually very difficult to solve (Ben-Tal et al. (2009)). When the second stage problem is a linear program, Benders decomposition method can be use to seek optimal solutions (Bertsimas et al. (2013); Rahmaniani et al. (2017)). However, it may not be efficient for large instances. Zeng & Zhao (2013) develop an other exact resolution method, the $(C\&CG)$ generation algorithm (also called row-and-column or scenario generation), which has performed better on different problems including facility localization problems (see e.g. An et al. (2014); Chan et al. (2018)).

Several robust variants of $(PCP)$ with either a single stage or two stages have been considered. For example, Averbakh & Berman (1997) consider the weighted $p$-center problem on a transportation network with uncertain node weights. They minimize the regret of the worst-case scenario and show that the problem can be solved through a number of particular weighted $p$-center problems. Averbakh & Berman (2000), consider a box uncertainty set for the weighted 1-center problem on a network with uncertainty node weights and edge lengths. Each uncertain parameter is assumed to be random with an unknown distribution. They present a polynomial algorithm to find the robust solution for the problem on a tree. Lu (2013) consider the single-stage weighted vertex $p$-center with uncertain nodal weights and edge lengths using also box uncertainty sets. They consider the single-stage robust problem $(RPCP_1)$, prove that it is sufficient to consider a discrete subset of scenarios, and propose a simulated annealing heuristic to solve the problem. Du & Zhou (2018) apply a single-stage approach to a $p$-center problem based on symmetric box uncertainty sets and a multiple allocation strategy. They consider three types of uncertainty sets: box uncertainty, ellipsoidal uncertainty, and cardinality-constrained uncertainty. Du et al. (2020) propose a two-stage robust model for reliable facility location problem when some facilities can be disrupted and the clients can be reallocated to another available facility. They consider uncertain demand and cost. They propose three resolution methods: a linear reformulation, a Benders dual cutting planes method, and a *column-and-constraint* generation method. Demange et al. (2020) introduce the robust $p$-center problem under pressure motivated by the context of locating shelters for evacuation in case of wildfires, where the uncertainty is in the available network connections. They present a MILP formulation and a decomposition scheme to solve it. Cheng et al. (2021) implement a *column-and-constraint* generation algorithm to solve a two-stage fixed-charge location problem, where demand and facility availability parameters are subject to uncertainties simultaneously.

Our research focuses on a problem $(RPCP_2)$ similar to the one in Lu (2013). The two main differences are that $(RPCP_2)$ is a two-stage problem unlike $(RPCP_1)$ and that we propose exact resolution algorithms. We also show how these algorithms can be adapted to solve exactly $(RPCP_1)$.

## 3. Robust weighted vertex *p*-center problem

We first define $(RPCP_2)$. We then prove that it is sufficient to consider a subset of the infinite scenarios in the box uncertainty set. Finally, we present the $(C\&CG)$ and $(B\&C)$ algorithms for the exact resolution of both $(RPCP_2)$ and $(RPCP_1)$.

### *3.1. Problem definition*

Following Lu (2013), we consider that the clients demand and the travel times can take any value in a box uncertainty set. More precisely, the demand $\xi_i$ of client $i \in V$ is assumed to be in $[\xi_i^-, \xi_i^+]$ where $0 \leq \xi_i^- \leq \xi_i^+$, while the travel time $t_{ij}$ between station $i \in V$ and center $j \in U$ takes its value in $[t_{ij}^-, t_{ij}^+]$ where $0 \leq t_{ij}^- \leq t_{ij}^+$.

Let $W \subset \mathbb{R}^{|V|+|U|\times|V|}$ be the Cartesian product of intervals $[\xi_i^-, \xi_i^+]$ and $[t_{ij}^-, t_{ij}^+]$ for each $i \in V$ and $j \in U$. Let $\Omega = \{x \in \{0,1\}^{|U|} \mid \sum_{j \in U} x_j = p\}$ be the set of vectors representing $p$ opened centers and let $J_x = \{j \in U \mid x_j = 1\}$ be the set of opened centers for vector $x \in \Omega$. For a given scenario $w \in W$ let $\xi_i^w$ and $t_{ij}^w$ respectively be the demand of client $i \in V$ and the travel time between $i$ and $j \in U$ in scenario $w$.

In $(RPCP_2)$, the clients are assigned after the uncertainty is revealed. This corresponds to a two-stage approach in which the opened centers are fixed at the first stage and the client assignments are the recourse decisions of the second stage. Consequently, the optimal radius associated with $x \in \Omega$ when scenario $w \in W$ occurs is:

$$Z(w, x) = \max_{i \in V} \left\{ \min_{j \in J_x} \xi_i^w t_{ij}^w \right\} \tag{16}$$

which represents an optimal assignment of clients in scenario $w$ when centers $J_x$ are opened. Let $x^*(w) \in \Omega$ be a vector such that the opening of $J_{x^*(w)}$ leads to an optimal radius for the deterministic $p$-center problem in which the uncertain data takes value $w \in W$. We define the *robust deviation* of $x \in \Omega$ for scenario $w$ as:

$$DEV(w, x) = Z(w, x) - Z(w, x^*(w)) \tag{17}$$

It corresponds to the increase in radius incurred by the opening of $J_x$ rather than $J_{x^*(w)}$ for the scenario $w$. The robustness cost of solution $x \in \Omega$ corresponds to the maximal possible robust deviation if centers $J_x$ are opened.

$$RC(x) = \max_{w \in W} DEV(w, x) \tag{18}$$

We denote by *worst-case scenario* a scenario which solves (18). The $(RPCP_2)$ aims to minimize the regret in the worst-case scenario for all $x \in \Omega$:

$$(RPCP_2): \ \min_{x \in \Omega} RC(x) \tag{19}$$

We now show that the uncertainty set $W$ can be significantly reduced.

### *3.2. Reducing the number of scenarios*

Since a box-uncertainty set contains an infinite number of scenarios for a given solution $x \in \Omega$, the evaluation of the robustness cost (18) is a major challenge when solving $(RPCP_2)$. We prove that it is sufficient to consider $n$ scenarios per solution $x \in \Omega$ to optimally solve $(RPCP_2)$.

**Definition 1.** *Let $w_{\bar{i}}(x)$ be the scenario for a client $\bar{i} \in V$ in a feasible solution $x \in \Omega$ such that:*

- $$\xi_i^{\omega_{\bar{i}}(x)} = \begin{cases} \xi_i^+ & \text{if } i = \bar{i} \\ \xi_i^- & \text{otherwise} \end{cases}$$

- $$t_{ij}^{\omega_{\bar{i}}(x)} = \begin{cases} t_{ij}^+ & \text{if } i = \bar{i} \text{ and } x_j = 1 \\ t_{ij}^- & \text{otherwise} \end{cases}$$

We now prove that at least one of the scenarios in $\{w_i(x)\}_{i \in V}$ leads to a maximal deviation for $x \in \Omega$.

**Theorem 1.** *Let $x \in \Omega$ be a first-stage solution of $(RPCP_2)$. There exists a scenario $w$ among the $n$ scenarios $\{w_i(x)\}_{i \in V}$ that solves* (18)*.*

**Proof:** Let $\overline{w} \in W$ be a scenario which provides a maximal deviation for $x$ (i.e., $RC(x) = DEV(\overline{w}, x)$). By definition, $DEV(\overline{w}, x) = Z(\overline{w}, x) - Z(\overline{w}, x^*(\overline{w}))$. Let us consider the two following couples (client, site):

- $(i_1, j_1) \in V \times J_x$ such that $Z(\overline{w}, x) = \xi_{i_1}^{\overline{w}} t_{i_1 j_1}^{\overline{w}}$ (i.e., $(i_1, j_1)$ allow to reach the optimal radius $Z(\overline{w}, x)$); and
- $(i_*, j_*) \in V \times J_{x^*(\overline{w})}$ such that $Z(\overline{w}, x^*(\overline{w})) = \xi_{i_*}^{\overline{w}} t_{i_* j_*}^{\overline{w}}$ (i.e., $(i_*, j_*)$ allow to reach the optimal radius $Z(\overline{w}, x^*(\overline{w}))$).

We now show that the worst-case scenario $\overline{w}$ can be transformed into scenario $w_{i_1}(x)$ without altering the value of the deviation. This will imply that $w_{i_1}(x)$ is also a worst-case scenario. To this end, we consider a scenario $w$ initially equal to $\overline{w}$ and apply five transformations to transform it into $w_{i_1}(x)$.

*<u>Transformation 1</u>*:

- $\xi_i^w \leftarrow \xi_i^- \ \forall\, i \in V \backslash \{i_1, i_*\}$
- $t_{ij}^w \leftarrow t_{ij}^- \ \forall\, i \in V \backslash \{i_1, i_*\} \ \forall\, j \in U$.

Neither $Z(w, x)$ nor $Z(w, x^*(w))$ is affected by this transformation since clients $i_1$ and $i_*$ remain at the same distance from their centers. Therefore, the deviation remains the same.

*Transformation 2*:

- $\xi^{w}_{i_1} \leftarrow \xi^{+}_{i_1}$.

This transformation increases:

(i) $Z(w, x)$ by $(\xi^{+}_{i_1} - \xi^{\overline{w}}_{i_1})\ \min_{j \in J_x} t^{\overline{w}}_{i_1 j}$;

(ii) $Z(w, x^*(w))$ by at most $(\xi^{+}_{i_1} - \xi^{\overline{w}}_{i_1})\ \min_{j \in J_{x^*(\overline{w})}} t^{\overline{w}}_{i_1 j}$.

We know that $\min_{j \in J_x} t^{\overline{w}}_{i_1 j} \geq \min_{j \in J_{x^*(\overline{w})}} t^{\overline{w}}_{i_1 j}$ (since otherwise $Z(\overline{w}, x^*(\overline{w})) > Z(\overline{w}, x)$) thus $Z(w, x)$ will increase at least as much as $Z(w, x^*(w))$ during this transformation. Thus, the deviation cannot decrease. It cannot increase either, because this would imply that $\overline{w}$ is not a scenario that maximizes the deviation.

*Transformation 3*:

If $i_1 \neq i_*$:

- $\xi^{w}_{i_*} \leftarrow \xi^{-}_{i_*}$
- $t^{w}_{i_* j} \leftarrow t^{-}_{i_* j}\ \forall\, j \in U$.

Since $i_1 \neq i_*$, decreasing $\xi^{w}_{i*}$ and $t^{w}_{i_* j}$ does not affect $Z(w, x)$. Therefore, this cannot reduce $Z(w, x^*(w))$ as it would increase the deviation.

*Transformation 4*:

- $t^{w}_{i_1, j} \leftarrow t^{-}_{i_1, j} \quad \forall\, j \in U \backslash J_x$.

These decreases do not affect $Z(w, x)$ because they involve closed sites ($x_j = 0$). Therefore, they cannot decrease $Z(w, x^*(w))$ because this would increase the deviation.

*Transformation 5*:

- $t^{w}_{i_1, j} \leftarrow t^{+}_{i_1, j}. \quad \forall j \in J_x$.

Let $j_2 \in U$ be the center to which client $i_1$ is allocated in an optimal allocation of solution $x^*(\overline{w})$ for scenario $\overline{w}$.

Let us first assume that $x_{j_2} = 1$, then $t^{\overline{w}}_{i_1 j_2} = t^{\overline{w}}_{i_1 j_1}$. Indeed,

(i) if $t^{\overline{w}}_{i_1 j_2} > t^{\overline{w}}_{i_1 j_1}$, then $Z(\overline{w}, x^*(\overline{w})) > Z(\overline{w}, x)$ which is impossible;

(ii) if $t^{\overline{w}}_{i_1 j_2} < t^{\overline{w}}_{i_1 j_1}$, then client $i_1$ would not be associated to $j_1$ when the centers $J_x$ are opened but rather to $j_2$ .

Thus, $Z(\overline{w}, x) = Z(\overline{w}, x^*(\overline{w}))$ and the deviation of $\overline{w}$ is zero. Therefore, the deviation of all the scenarios is also zero and the theorem is satisfied.

Now suppose that $x_{j_2} = 0$. For all $j \in J_x$, increasing $t^w_{i_1 j}$ does not increase $Z(w, x^*(w))$ since client $i_1$ is associated with $j_2$ when centers $J_{x^*(\overline{w})}$ are opened. $Z(w, x)$ cannot either be increased because this would increase the deviation.

Scenario $w$ is now equal to $w_{i_1}(x)$.

□

Since $\Omega$ is finite, Theorem (1) enables to only consider a finite set of scenarios $\overline{W} = \{w_i(x) \mid x \in \Omega,\ i \in V\}$ without losing the optimality:

$$(RPCP_2): \qquad \min_{x \in \Omega} \Big\{ \max_{w \in \overline{W}} DEV(w, x) \Big\} \tag{20}$$

Lu (2013) considered a similar problem that we denote by $(RPCP_1)$. He proved that a different subset of scenarios of $W$ can also be considered without losing optimality. The key difference between $(RPCP_2)$ and $(RPCP_1)$ is that the latter is a single-stage problem as it does not consider any recourse. Consequently, the client assignments are fixed before the uncertainty is revealed.

### *3.3. MILP formulations of the robust weighted vertex p-center problem*

We present how the three formulations of Section (2.1) can be adapted to solve $(RPCP_2)$. Let $Z^*_w$ be the optimal value of the $(PCP)$ problem when the uncertain parameters take value $w \in \overline{W}$ (i.e., $Z^*_w = Z(w, x^*(w))$). In the following formulations, we suppose that $Z^*_w$ can be computed using by an oracle and it is considered as a parameter.

Our robust formulation based on $(F1)$ uses one set of assignment variables $y^w_{ij}$ for each scenario $w \in \overline{W}$ to allow different client assignments depending on the scenario:

$$
(RF1): \quad \begin{aligned}
& \min RC && && (21)\\
\text{s.t.:} \quad & RC \geq \sum_{j \in U} \xi_i^w t_{ij}^w \cdot y_{ij}^w - Z_w^*, && i \in V,\ w \in \overline{W}, && (22)\\
& \sum_{j \in U} y_{ij}^w = 1, && i \in V,\ w \in \overline{W}, && (23)\\
& y_{ij}^w \leq x_j, && i \in V,\ j \in U,\ w \in \overline{W}, && (24)\\
& \sum_{j \in U} x_j = p, && && \\
& x_j \in \{0,1\}, && j \in U, && \\
& y_{ij}^w \in \{0,1\}, && i \in V,\ j \in U,\ w \in \overline{W}. && (25)
\end{aligned}
$$

Constraints (22) set a lower bound to the value of the robustness cost (RC) for each scenario. Objective (21) provides a solution with the lowest maximal deviation. This formulation contains an exponential number of variables and constraints as the size of $\overline{W}$ is proportional to $|\Omega|$.

To adapt formulation $(F2)$ to $(RPCP_2)$, one needs to sort the values $\{\xi_i^w t_{ij}^w\}_{i \in V, j \in U}$ in order to obtain a set of distinct distances $D_w$ for each scenario $w \in \overline{W}$. From these distances, we deduce the set of indices $S_i^w$ which plays a similar role than $S_i$ in Constraints (12). For each scenario $w \in \overline{W}$, we also consider one set of radius variables $z_w^k$:

$$
(RF2): \quad \begin{aligned}
& \min RC && && \\
\text{s.t.:} \quad & RC \geq D_w^0 + \sum_{k \in \mathcal{K}} \left( D_w^k - D_w^{k-1} \right) z_w^k - Z_w^*, && w \in \overline{W}, && (26)\\
& z_w^k \geq z_w^{k+1}, && k \in \{1, \ldots, K-1\}, w \in \overline{W}, && (27)\\
& z_w^k + \sum_{j : \xi_i^w t_{ij}^w < D_w^k} x_j \geq 1, && i \in V,\ k \in S_i^w \cup K,\ w \in \overline{W}, && (28)\\
& \sum_{j \in U} x_j = p, && && \\
& x_j \in \{0,1\}, && j \in U, && \\
& z_w^k \in \{0,1\}, && k \in \mathcal{K},\ w \in \overline{W}. && (29)
\end{aligned}
$$

Formulation $(F3)$ does not directly provide the value $R$ of the optimal radius but its index $r$ instead (i.e., $D^r = R$). This raises a problem when considering the adaptation of $(F3)$ to the resolution of $(RPCP_2)$ as a given index does not necessarily correspond to the same distance in different scenarios. Consequently, we first modify $(F3)$ so that it provides a distance rather than its index. We replace Constraints (14) by:

$$r + D^k \sum_{j:\xi_i t_{ij} < D^k} x_j \geq D^k, \qquad \forall\, i \in V, k \in \mathcal{K}, \tag{30}$$

We can now obtain a reformulation of $(RPCP_2)$ based on $(F3)$:

$$(RF3): \quad \begin{aligned} & \min RC \\ \text{s.t.:} \quad & RC \geq D_w^k(1 - \sum_{j:\xi_i^w t_{ij}^w < D_w^k} x_j) - Z_w^*, && i \in V,\ k \in \mathcal{K},\ w \in \overline{W}, \qquad (31) \\ & \sum_{j \in U} x_j = p, \\ & x_j \in \{0,1\}, && j \in U. \end{aligned}$$

Note that $(RF3)$ does not require an exponential number of variables, which is a significant advantage compared to $(RF1)$ and $(RF2)$.

### *3.4. Column-and-constraint generation algorithm*

We cannot directly solve a formulation with an exponential number of constraints or variables. Therefore, we first propose a $(C\&CG)$ algorithm. Let $(\overline{RF})$ be any of our three robust formulations (RF1), (RF2), or (RF3) in which $\overline{W}$ is initially empty. Our $(C\&CG)$ algorithm is presented in Algorithm 1.

At each iteration, Algorithm 1 generates a solution $(x, RC)$ which satisfy all the scenarios currently in $\overline{W}$ by solving $(\overline{RF})$ (Step 3). If the solution does not satisfy one of the scenario $\{w_i(x)\}_{i \in V}$ (Step 10), the most violated scenario is added to $\overline{W}$ (Step 1). When no violated scenario is found, an optimal solution is returned.

The value of the optimal radius considering a scenario $w_i(x)$ can be calculated by solving a deterministic $(PCP)$ (Step 7). Note that the radius associated with the feasible solution $x$ considering the same scenario $w_i$ can be calculated directly as it only requires to determine the distance between each client and its closest center in $J_x$ (Step 8).

**Algorithm 1:** *Column-and-constraint* generation algorithm

**input :**

- Instance data ($V$, $U$, $p$, $[\xi_i^-, \xi_i^+]$ and $[t_{ij}^-, t_{ij}^+]$ for each $i \in V$ and $j \in U$).
- A robust formulation ($\overline{RF}$) for the ($PCP$).
- A solver for the deterministic ($PCP$).

**output :**

- An optimal solution $x$ of ($\overline{RF}$) and its robustness cost $RC$.

$RC \leftarrow 0$, $\overline{W} \leftarrow \emptyset$, $isOptimal \leftarrow false$

**while** $isOptimal = false$ **do**

  $(x, RC) \leftarrow$ solve ($\overline{RF}$) with scenarios $\overline{W}$

  $isOptimal \leftarrow true$

  $\overline{w} \leftarrow \emptyset$

  **for** $i \in V$ **do**

    $Z^* \leftarrow$ optimal radius of the deterministic ($PCP$) for scenario $w_i(x)$

    $Z \leftarrow$ radius for $x$ in scenario $w_i(x)$

    $DEV \leftarrow (Z - Z^*)$

    **if** $DEV > RC$ **then**

      $isOptimal \leftarrow false$

      $RC \leftarrow DEV$

      $\overline{w} \leftarrow w_i(x)$

  $\overline{W} \leftarrow \overline{W} \cup \{\overline{w}\}$

**return** $x$ and $RC$

### 3.5. *Branch-and-cut algorithm*

The main advantage of $(RF3)$ over $(RF1)$ and $(RF2)$ is that no new variable is required when a scenario is added to $\overline{W}$. Consequently, we can define a $(B\&C)$ algorithm which checks if each obtained integer solution $(x, RC)$ satisfies all the scenarios $\{w_i(x)\}$. We generate violated inequalities if it does not. This can be performed through callbacks which is a feature provided by mixed integer programming solvers. Consequently, Steps 3-14 of Algorithm 1 are performed within the callbacks. This modification allows us to only generate a single search tree instead of solving $(\overline{RF})$ from scratch at each iteration.

### 3.6. *Adaptation to the single-stage problem*

The $(C\&CG)$ and $(B\&C)$ algorithms previously presented can be adapted to solve the single-stage problem $(RPCP_1)$ using $(RF1)$. For this purpose, we only consider one single set of assignment variables $y_{ij}$ which ensures that the clients assignments are the same regardless of the scenario. Note that for $(RPCP_1)$ the finite set of scenarios that a solution must satisfy to ensure its optimality is different from $\{w_i(x)\}_{i\in V}$ as proved in Theorem 1 of Lu (2013).

These adaptations are not possible for the $(C\&CG)$ and $(B\&C)$ algorithms based on $(RF2)$ or $(RF3)$. Indeed, only considering one set of variables $z_k$ in $(RF2)$ would only ensure that the distance index of the radius is the same in each scenario, not that the client assignments are. For the algorithms based on $(RF3)$ the adaptation seems even less possible as this formulation does not contain scenario variables and as the client assignments are determined implicitly in Constraints (31).

## 4. Computational study

We evaluate the efficiency of our $(C\&CG)$ and $(B\&C)$ algorithms on randomly generated instances and on a case study presented in Lu (2013). It was not possible to make a direct comparison with Lu (2013), because the solution values presented in Lu (2013) are not consistent with the ones obtained by an exact resolution. This is illustrated in Appendix A.

Our study was carried out on an Intel XEON W-2145 processor 3,7 GHz, with 16 threads, but only one was used, and 256 GB RAM. IBM ILOG CPLEX 20.1. For the $(B\&C)$ algorithm, we use the LazyCallback of CPLEX, which gets called whenever a feasible integer solution is found. We set optimality tolerance EpGap to $10^{-10}$. We consider a time limit of 7,200 seconds.

### 4.1. Randomly generated instances

To evaluate our exact resolution methods, we created random instances from a deterministic instance following Lu (2013). Two dimensional coordinates were uniformly drawn from $[0;\ 100] \times [40;\ 60]$. The travel times $t_{ij}$ between clients and centers was set to the nearest integer of their euclidean distance. The demand $\xi_i$ of each client $i \in V$ was uniformly drawn from the interval $[1,000;\ 3,000]$. The travel time uncertainty box is $[t_{ij};\ t_{ij}(1+\alpha_1)]$, and the demand uncertainty box is $[\xi_i(1-\alpha_2);\ \xi_i(1+\alpha_2)]$ with $\alpha_1 \in \{0.5,\ 1.5,\ 2.5\}$ and $\alpha_2 \in \{0.2,\ 0.4,\ 0.6\}$.

We consider 72 instances of the robust problem with the 9 possible combinations of parameter values $\alpha_1$ and $\alpha_2$ from the following 8 deterministic instances: $n = 15$, $m = 5$, $p = \{2, 3\}$; $n = 40$, $m = 8$, $p = \{3, 4\}$; $n = m = 10$, $p = \{2, 3\}$; and $n = m = 15$, $p = \{2, 3\}$. The results obtained for these instances are presented in Tables 1, 2, 3, and 4 respectively.

We can see that $(RF3)$ is the fastest $(C\&CG)$ algorithm and that the use of the $(B\&C)$ algorithm enables $(RF3)$ to be significantly faster in all instances. This could be explained by the lighter structure of $(RF3)$ with respect to $(RF1)$ and $(RF2)$, which does not need to add more variables when adding a scenario.

On the one hand, we can notice how the robustness cost and the resolution time increase with $\alpha_1$ and $\alpha_2$, i.e., with the size of the box uncertainty set. On the other hand, they decrease as the value of $p$ increases.

The difficulty of an instance for the $(RPCP_2)$ lies mainly in the number of feasible solutions which is proportional to $\begin{pmatrix} m \\ p \end{pmatrix}$ and to the sizes of the uncertainty boxes considered. The instances with $m = 10$ are all solved optimally by all algorithms. However, for $m = 15$ $(RF2)$ and $(RF3)$ do not solve the 3 instances with the largest values of $\alpha_1$ and $\alpha_2$ in our time limit of 7,200 seconds.

| Instance | | | | | | Time(s) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | C&CG | | | B&C |
| $n$ | $m$ | $p$ | $\alpha_1$ | $\alpha_2$ | RC | RF1 | RF2 | RF3 | RF3 |
| 15 | 5 | 2 | 0.5 | 0.2 | 89,159 | 0.7 | 0.7 | 0.7 | 0.8 |
| 15 | 5 | 2 | 0.5 | 0.4 | 124,236 | 1.0 | 0.8 | 0.7 | 0.7 |
| 15 | 5 | 2 | 0.5 | 0.6 | 141,984 | 0.8 | 0.9 | 0.6 | 0.5 |
| 15 | 5 | 2 | 1.5 | 0.2 | 200,226 | 1.4 | 1.4 | 0.9 | 0.9 |
| 15 | 5 | 2 | 1.5 | 0.4 | 261,540 | 1.0 | 1.0 | 0.7 | 0.7 |
| 15 | 5 | 2 | 1.5 | 0.6 | 298,980 | 0.8 | 0.8 | 0.6 | 0.5 |
| 15 | 5 | 2 | 2.5 | 0.2 | 310,349 | 1.4 | 1.4 | 0.9 | 0.8 |
| 15 | 5 | 2 | 2.5 | 0.4 | 392,310 | 0.9 | 0.9 | 0.7 | 0.7 |
| 15 | 5 | 2 | 2.5 | 0.6 | 448,470 | 0.8 | 0.7 | 0.6 | 0.5 |
| 15 | 5 | 3 | 0.5 | 0.2 | 55,454 | 0.6 | 1.1 | 0.7 | 0.8 |
| 15 | 5 | 3 | 0.5 | 0.4 | 67,774 | 0.4 | 0.4 | 0.3 | 0.3 |
| 15 | 5 | 3 | 0.5 | 0.6 | 100,032 | 0.4 | 0.6 | 0.4 | 0.3 |
| 15 | 5 | 3 | 1.5 | 0.2 | 153,314 | 1.1 | 1.0 | 0.8 | 0.6 |
| 15 | 5 | 3 | 1.5 | 0.4 | 178,835 | 0.9 | 0.8 | 0.6 | 0.5 |
| 15 | 5 | 3 | 1.5 | 0.6 | 204,403 | 0.7 | 0.7 | 0.5 | 0.4 |
| 15 | 5 | 3 | 2.5 | 0.2 | 251,174 | 1.0 | 1.0 | 0.7 | 0.7 |
| 15 | 5 | 3 | 2.5 | 0.4 | 292,985 | 0.8 | 0.8 | 0.6 | 0.6 |
| 15 | 5 | 3 | 2.5 | 0.6 | 334,873 | 0.7 | 0.8 | 0.5 | 0.4 |
| Total | | | | | | 15.4 | 15.7 | 11.7 | 10.8 |

Table 1: Results on randomly generated instances for the $(RPCP_2)$ with $n = 15$, $m = 5$, and $p = \{2, 3\}$.

| Instance | | | | | | Time(s) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | C&CG | | | B&C |
| $n$ | $m$ | $p$ | $\alpha_1$ | $\alpha_2$ | RC | RF1 | RF2 | RF3 | RF3 |
| 40 | 8 | 3 | 0.5 | 0.2 | 56,365 | 10 | 15 | 9 | 20 |
| 40 | 8 | 3 | 0.5 | 0.4 | 91,366 | 22 | 15 | 13 | 21 |
| 40 | 8 | 3 | 0.5 | 0.6 | 113,352 | 14 | 15 | 14 | 14 |
| 40 | 8 | 3 | 1.5 | 0.2 | 134,883 | 141 | 93 | 80 | 114 |
| 40 | 8 | 3 | 1.5 | 0.4 | 182,974 | 164 | 125 | 104 | 91 |
| 40 | 8 | 3 | 1.5 | 0.6 | 228,384 | 200 | 190 | 110 | 94 |
| 40 | 8 | 3 | 2.5 | 0.2 | 219,164 | 1,005 | 776 | 335 | 231 |
| 40 | 8 | 3 | 2.5 | 0.4 | 274,582 | 603 | 436 | 247 | 246 |
| 40 | 8 | 3 | 2.5 | 0.6 | 333,060 | 572 | 389 | 213 | 188 |
| 40 | 8 | 4 | 0.5 | 0.2 | 45,122 | 16 | 15 | 16 | 16 |
| 40 | 8 | 4 | 0.5 | 0.4 | 57,862 | 16 | 22 | 19 | 16 |
| 40 | 8 | 4 | 0.5 | 0.6 | 74,638 | 13 | 11 | 15 | 16 |
| 40 | 8 | 4 | 1.5 | 0.2 | 132,143 | 215 | 184 | 140 | 79 |
| 40 | 8 | 4 | 1.5 | 0.4 | 154,160 | 154 | 114 | 69 | 55 |
| 40 | 8 | 4 | 1.5 | 0.6 | 176,218 | 67 | 90 | 90 | 50 |
| 40 | 8 | 4 | 2.5 | 0.2 | 216,790 | 1,118 | 740 | 439 | 298 |
| 40 | 8 | 4 | 2.5 | 0.4 | 255,680 | 610 | 399 | 291 | 184 |
| 40 | 8 | 4 | 2.5 | 0.6 | 292,264 | 565 | 445 | 282 | 169 |
| Total | | | | | | 5,505 | 4,075 | 2,487 | 1,901 |

Table 2: Results on randomly generated instances for the $(RPCP_2)$ with $n = 40$, $m = 8$, and $p = \{3, 4\}$.

| Instance | | | | | | Time(s) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | C&CG | | | B&C |
| $n$ | $m$ | $p$ | $\alpha_1$ | $\alpha_2$ | RC | RF1 | RF2 | RF3 | RF3 |
| 10 | 10 | 2 | 0.5 | 0.2 | 26,808 | 0.9 | 2.1 | 0.9 | 0.5 |
| 10 | 10 | 2 | 0.5 | 0.4 | 51,454 | 0.7 | 1.2 | 0.4 | 0.5 |
| 10 | 10 | 2 | 0.5 | 0.6 | 66,958 | 0.3 | 0.4 | 0.3 | 0.3 |
| 10 | 10 | 2 | 1.5 | 0.2 | 137,000 | 14.1 | 16.9 | 5.9 | 1.7 |
| 10 | 10 | 2 | 1.5 | 0.4 | 127,778 | 5.6 | 8.4 | 3.8 | 1.5 |
| 10 | 10 | 2 | 1.5 | 0.6 | 222,318 | 14.7 | 23.5 | 21.2 | 1.8 |
| 10 | 10 | 2 | 2.5 | 0.2 | 117,351 | 1.7 | 3.3 | 0.5 | 0.6 |
| 10 | 10 | 2 | 2.5 | 0.4 | 190,575 | 2.1 | 4.9 | 1.3 | 0.8 |
| 10 | 10 | 2 | 2.5 | 0.6 | 177,420 | 4.6 | 7.3 | 1.6 | 0.9 |
| 10 | 10 | 3 | 0.5 | 0.2 | 30,672 | 8.7 | 6.8 | 3.0 | 1.6 |
| 10 | 10 | 3 | 0.5 | 0.4 | 51,066 | 0.8 | 3.0 | 0.8 | 0.8 |
| 10 | 10 | 3 | 0.5 | 0.6 | 44,911 | 0.4 | 0.8 | 0.3 | 0.4 |
| 10 | 10 | 3 | 1.5 | 0.2 | 97,944 | 5.3 | 5.0 | 3.5 | 1.4 |
| 10 | 10 | 3 | 1.5 | 0.4 | 67,966 | 0.8 | 1.6 | 0.8 | 1.2 |
| 10 | 10 | 3 | 1.5 | 0.6 | 133,830 | 5.4 | 7.7 | 4.4 | 1.0 |
| 10 | 10 | 3 | 2.5 | 0.2 | 100,860 | 7.5 | 14.8 | 11.9 | 1.2 |
| 10 | 10 | 3 | 2.5 | 0.4 | 147,884 | 10.5 | 20.9 | 3.2 | 1.5 |
| 10 | 10 | 3 | 2.5 | 0.6 | 162,844 | 9.7 | 13.8 | 5.6 | 4.5 |
| Total | | | | | | 93.8 | 142.2 | 69.3 | 22.2 |

Table 3: Results on randomly generated instances for the ($RPCP_2$) with $n = m = 10$, and $p = \{2, 3\}$.

| Instance | | | | | | Time(s) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | C&CG | | | B&C |
| $n$ | $m$ | $p$ | $\alpha_1$ | $\alpha_2$ | RC | RF1 | RF2 | RF3 | RF3 |
| 15 | 15 | 2 | 0.5 | 0.2 | 90,270 | 19 | 27 | 22 | 9 |
| 15 | 15 | 2 | 0.5 | 0.4 | 137,020 | 13 | 18 | 10 | 11 |
| 15 | 15 | 2 | 0.5 | 0.6 | 174,330 | 11 | 17 | 8 | 4 |
| 15 | 15 | 2 | 1.5 | 0.2 | 201,110 | 284 | 411 | 61 | 39 |
| 15 | 15 | 2 | 1.5 | 0.4 | 266,356 | 186 | 220 | 84 | 38 |
| 15 | 15 | 2 | 1.5 | 0.6 | 326,025 | 196 | 164 | 54 | 26 |
| 15 | 15 | 2 | 2.5 | 0.2 | 311,950 | 1,572 | 722 | 167 | 86 |
| 15 | 15 | 2 | 2.5 | 0.4 | 395,692 | 1,050 | 698 | 242 | 79 |
| 15 | 15 | 2 | 2.5 | 0.6 | 473,823 | 1,154 | 789 | 158 | 52 |
| 15 | 15 | 3 | 0.5 | 0.2 | 55,455 | 2 | 7 | 2 | 4 |
| 15 | 15 | 3 | 0.5 | 0.4 | 84,960 | 1 | 5 | 2 | 2 |
| 15 | 15 | 3 | 0.5 | 0.6 | 121254 | 13 | 16 | 7 | 6 |
| 15 | 15 | 3 | 1.5 | 0.2 | 132,961 | 923 | 1,377 | 183 | 46 |
| 15 | 15 | 3 | 1.5 | 0.4 | 175,380 | 791 | 954 | 113 | 51 |
| 15 | 15 | 3 | 1.5 | 0.6 | 224,610 | 902 | 962 | 230 | 67 |
| 15 | 15 | 3 | 2.5 | 0.2 | 210,467 | TL | TL | 932 | 149 |
| 15 | 15 | 3 | 2.5 | 0.4 | 265,800 | TL | TL | 1,010 | 181 |
| 15 | 15 | 3 | 2.5 | 0.6 | 327,966 | TL | TL | 1,063 | 219 |
| Total | | | | | | 7,118 | 6,386 | 4,348 | 1,071 |

Table 4: Results on randomly generated instances for the ($RPCP_2$) with $n = m = 15$, and $p = \{2, 3\}$. TL: Instance not solved within the time limit of 7,200 seconds.

*4.2. Case Study*

Lu (2013) presents a case study on the location of urgent relief distribution centers (URDCs) in a relief supply distribution network responding to the massive earthquake which hit central Taiwan on September 21, 1999. A three-tier relief supply distribution network was established in Nantou County immediately after this earthquake. Specifically, relief supplies were collected from six unaffected counties transported to two URDCs at Nantou Stadium and Jiji Town Hall, and then delivered to the 51 relief stations in the 11 townships in Nantou County. Five other candidate sites for URDCs were selected. Due to the difficulty to precisely estimate the relief demand faced by each relief station, they divided the number of survivors, i.e., the total population minus the number of deaths, by the number of relief stations of each township. They use the data collected in previous research for the travel time between a URDC and a relief station.

We solved nine instances obtained from this case study in which the travel time and the demand values were constructed following the same methodology as the random instances presented in subsection 4.1, considering 51 clients, 7 possible sites, and the selection of 2 centers. The travel time uncertainty box of a client $i$ and a center $j$ is in $[t_{ij},\ t_{ij}(1+\alpha_1)]$ and the demand uncertainty box is $[\xi_i(1-\alpha_2),\ \xi_i(1+\alpha_2)]$ with $\alpha_1 \in \{0.5,\ 1.5,\ 2.5\}$ and $\alpha_2 \in \{0.2,\ 0.4,\ 0.6\}$.

Table 5 shows the results obtained for the resolution of $(RPCP_2)$ by applying our algorithms to the case study. Similarly to randomly generated instances, $(RF3)$ is faster than $(RF2)$ and $(RF3)$ within the $(C\&CG)$ and $(B\&C)$ algorithms.

| Instance | | | | | | Time (s) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | C&CG | | | B&C |
| $n$ | $m$ | $p$ | $\alpha_1$ | $\alpha_2$ | $RC$ | RF1 | RF2 | RF2 | RF3 |
| 52 | 7 | 2 | 0,5 | 0,2 | 495,000 | 5.71 | 6.27 | 5.82 | 6.04 |
| 52 | 7 | 2 | 0,5 | 0,4 | 838,500 | 9.01 | 9.21 | 6.82 | 10.83 |
| 52 | 7 | 2 | 0,5 | 0,6 | 1,159,200 | 8.56 | 6.97 | 6.62 | 6.12 |
| 52 | 7 | 2 | 1,5 | 0,2 | 1,238,400 | 41.23 | 40.37 | 34.22 | 20.68 |
| 52 | 7 | 2 | 1,5 | 0,4 | 1.705,800 | 34.30 | 32.76 | 29.12 | 23.97 |
| 52 | 7 | 2 | 1,5 | 0,6 | 2,150,400 | 32.67 | 32.36 | 28.60 | 18.84 |
| 52 | 7 | 2 | 2,5 | 0,2 | 1,981,800 | 58.74 | 54.53 | 41.84 | 30.42 |
| 52 | 7 | 2 | 2,5 | 0,4 | 2,573,100 | 59.61 | 46.74 | 43.79 | 28.05 |
| 52 | 7 | 2 | 2,5 | 0,6 | 3,141,600 | 59.63 | 50.81 | 47.16 | 27.77 |
| Total | | | | | | 309.46 | 280.02 | 243.99 | 172.71 |

Table 5: Results obtained for the resolution of $(RPCP_2)$ on the case study instance.

## 5. Conclusions

The weighted vertex $p$-center problem consists of locating $p$ facilities among a set of potential sites such that the maximum weighted distance from any demand node to its closest open facility is minimized. The incorporation of uncertain information within a robust optimization approach allows us to solve emergency logistics problems. However, the robust counterpart of this problem is even more difficult. Therefore, most studies propose a heuristic approach instead of an exact resolution.

We studied the resolution of a robust weighted vertex $p$-center problem, considering uncertain nodal weights demand and edge lengths using box uncertainty sets. Two variants of this problem are possible depending on whether the client assignments to the centers are made after the uncertainty is revealed $(RPCP_2)$ or not $(RPCP_1)$.

For $(RPCP_2)$, similarly to $(RPCP_1)$, we prove that a finite subset of scenarios from the box uncertainty set can be considered without losing optimality. We use this result to propose three robust reformulations based on different MILP formulations of the vertex $p$-center problem. To optimally solve these reformulations, we introduce a *column-and-constraint* generation algorithm and a *branch-and-cut* algorithm. Moreover, we highlight how they can be adapted to optimally solve $(RPCP_1)$.

We present a numerical study to compare the performances of the algorithms on randomly generated instances and a case study. We see how our algorithms were able to solve optimally the 80 instances considered. The *column-and-constraint* generation algorithm based on formulation $(RF3)$ is more efficient than the one based on $(RF1)$ and $(RF3)$. This is because adding a scenario does not require the addition of any variable. This formulation enables the implementation of *branch-and-cut* algorithm which significantly reduces the resolution time.

In future work, analysis of larger instances with other random box uncertainty sets could be considered. To further improve the performance of the *branch-and-cut* algorithm, other branching strategies could be evaluated and integrality cuts could be dynamically generated.

## Appendix A. Comparison of results with Lu (2013)

We were not able to compare the performances of our exact algorithms and the heuristic in Lu (2013) as its results do not seem to be correct. We prove that several robustness costs obtained with this heuristic and reported in Lu (2013) are undervalued.

Since we consider problem ($RPCP_1$), the client allocations are fixed before the uncertainty is revealed. Consequently, allocation variables $y$ are now necessary to compute the radius and the robustness cost of a solution. Given a feasible solution $(x, y)$, let $t_i^w$ be the travel time in scenario $w \in W$ between client $i \in V$ and its assigned center (i.e., $t_i^w = t_{ij}^w$ with $j \in U$ the only center such that $y_{ij} = 1$). In this single-stage problem, the radius of solution $(x, y)$ is $\max\limits_{i \in V} \xi_i^w t_i^w$ and its robustness cost is $RC(x, y) = \max\limits_{w \in W} \left\{ \max\limits_{i \in V} \xi_i^w t_i^w - Z_w^* \right\}$, where $Z_w^*$ is the optimal solution of the deterministic $p$-center problem in which the uncertain parameters take value $w$.

A difficulty to evaluate the robustness costs reported in Lu (2013) is that for each solution only one client's allocation is provided. Let us consider the instance described in Subsection 4.2 in which $\alpha_1 = 0.5$ and $\alpha_2 = 0.2$ and let $(x^h, y^h)$ be its associated solution in Lu (2013). We only know in $(x^h, y^h)$ that centers 1 and 2 are opened and that client 21 is assigned to center 1. Nevertheless, this is sufficient to obtain a lower bound on the robustness cost as for any scenario $w \in W$ and any client $i \in V$, the expression $\xi_i^w t_i^w - Z_w^*$ constitutes a lower bound of $RC(x^h, y^h)$. Let us consider a scenario $w_{21}$ in which the demand of client 21 is $\xi_{21}^w = \xi_{21}^+ = 34,800$ and the distance to its closest opened center is $t_{21,1}^w = t_{21,1}^+ = 41$. The optimal radius $Z_{w_{21}}^* = 495,600$ is obtained by solving a deterministic $p$-center problem. Consequently, a lower bound on the robustness cost of value $34,800 \times 41 - 495,600 = 931,200$ is obtained, which is higher than the value $93,619$ reported in Lu (2013).

Table A.6 present similar results on several instances. The third column contains the robustness costs reported in Lu (2013) which are all undervalued. Indeed, they are significantly lower than their associated lower bounds presented in Column 4. Column 5 contains the robustness cost of optimal solutions obtained by our ($B\&C$) algorithm adapted to ($RPCP_1$) (see Section 3.6). Note that the branch-and-cut always returns a solution which robustness cost is always lower than the lower bound of the heuristic solution.

| Instance | | Robustness cost | | |
|---|---|---|---|---|
| | | Heuristic solution from Lu (2013) | | Optimal solution |
| $\alpha_1$ | $\alpha_2$ | According to Lu (2013) | According to this article | |
| 0.5 | 0.2 | = 93,619 | ≥ 931,200 | = 495,000 |
| 0.5 | 0.4 | = 587,837 | ≥ 1,292,900 | = 838,500 |
| 0.5 | 0.6 | = 1,477,709 | ≥ 1,906,600 | = 1,159,200 |
| 1.5 | 0.2 | = 940,858 | ≥ 1,870,800 | = 1,238,400 |
| 1.5 | 0.4 | = 1,859,069 | ≥ 2,389,100 | = 1,705,800 |
| 1.5 | 0.6 | = 2,934,605 | ≥ 3,362,600 | = 2,150,400 |
| 2.5 | 0.2 | = 1,883,309 | ≥ 2,810,400 | = 1,981,800 |
| 2.5 | 0.4 | = 3,400,291 | ≥ 4,029,900 | = 2,573,100 |
| 2.5 | 0.6 | = 4,356,480 | ≥ 4,129,600 | = 3,141,600 |

Table A.6: Comparison of the robustness cost of solutions obtained by the heuristic presented in Lu (2013) and optimal solutions obtained by the branch-and-cut algorithm adapted for $(RPCP_1)$.

## References


Ales, Z., & Elloumi, S. (2018). Compact milp formulations for the p-center problem. In *International Symposium on Combinatorial Optimization* (pp. 14–25). Springer. doi:10.1007/978-3-319-96151-4_2.

An, Y., Zeng, B., Zhang, Y., & Zhao, L. (2014). Reliable p-median facility location problem: two-stage robust models and algorithms. *Transportation Research Part B: Methodological*, *64*, 54–72. doi:https://doi.org/10.1016/j.trb.2014.02.005.

Averbakh, I., & Berman, O. (1997). Minimax regret p-center location on a network with demand uncertainty. *Location Science*, *5*, 247–254. doi:10.1016/S0966-8349(98)00033-3.

Averbakh, I., & Berman, O. (2000). Algorithms for the robust 1-center problem on a tree. *European Journal of Operational Research*, *123*, 292–302. doi:10.1016/S0377-2217(99)00257-X.

Baron, O., Milner, J., & Naseraldin, H. (2011). Facility location: A robust optimization approach. *Production and Operations Management*, *20*. doi:10.1111/j.1937-5956.2010.01194.x.

Ben-Tal, A., Ghaoui, L., & Nemirovski, A. (2009). *Robust Optimization*. Princeton Series in Applied Mathematics. Princeton University Press. URL: https://books.google.cl/books?id=DttjR7IpjUEC.

Bertsimas, D., Litvinov, E., Sun, X. A., Zhao, J., & Zheng, T. (2013). Adaptive robust optimization for the security constrained unit commitment problem. *IEEE Transactions on Power Systems*, *28*, 52–63. doi:10.1109/TPWRS.2012.2205021.

Bertsimas, D., & Sim, M. (2004). The price of robustness. *Operations Research*, *52*, 35–53. doi:10.1287/opre.1030.0065.

Çalık, H., Labbé, M., & Yaman, H. (2019). p-center problems. In *Location science* (pp. 51–65). Springer. doi:10.1007/978-3-030-32177-2_3.

Calik, H., & Tansel, B. C. (2013). Double bound method for solving the p-center location problem. *Computers & Operations Research*, *40*, 2991–2999. doi:10.1016/j.cor.2013.07.011.

Chan, T. C. Y., Shen, Z.-J. M., & Siddiq, A. (2018). Robust defibrillator deployment under cardiac arrest location uncertainty via row-and-column generation. *Operations Research*, *66*, 358–379. URL: https://doi.org/10.1287/opre.2017.1660. doi:10.1287/opre.2017.1660.

Cheng, C., Adulyasak, Y., & Rousseau, L.-M. (2021). Robust facility location under demand uncertainty and facility disruptions. *Omega*, *103*, 102429. doi:10.1016/j.omega.2021.102429.

Correia, I., & Saldanha-da Gama, F. (2019). Facility location under uncertainty. In *Location science* (pp. 185–213). Springer. doi:10.1007/978-3-030-32177-2_8.

Daskin, M. (1996). Network and discrete location: Models, algorithms and applications. *Journal of the Operational Research Society*, *48*. doi:10.1057/palgrave.jors.2600828.

Demange, M., Gabrel, V., Haddad, M. A., & Murat, C. (2020). A robust p-center problem under pressure to locate shelters in wildfire context. *EURO Journal on Computational Optimization*, *8*, 103–139. doi:10.1007/s13675-020-00124-x.

Du, B., & Zhou, H. (2018). A robust optimization approach to the multiple allocation p-center facility location problem. *Symmetry*, *10*, 588. doi:10.3390/sym10110588.

Du, B., Zhou, H., & Leus, R. (2020). A two-stage robust model for a reliable p-center facility location problem. *Applied Mathematical Modelling*, *77*, 99–114. doi:10.1016/j.apm.2019.07.025.

Elloumi, S., Labbé, M., & Pochet, Y. (2004). A New Formulation and Resolution Method for the p-Center Problem. *INFORMS Journal on Computing*, *16*, 84–94. doi:10.1287/ijoc.1030.0028.

Hakimi, S. L. (1965). Optimum distribution of switching centers in a communication network and some related graph theoretic problems. *Operations research*, *13*, 462–475. doi:10.1287/opre.13.3.462.

Hasani, A., & Mokhtari, H. (2018). Redesign strategies of a comprehensive robust relief network for disaster management. *Socio-Economic Planning Sciences*, *64*, 92–102. doi:10.1016/j.seps.2018.01.003.

Lu, C.-C. (2013). Robust weighted vertex p-center model considering uncertain data: An application to emergency management. *European Journal of Operational Research*, *230*, 113–121. doi:10.1016/j.ejor.2013.03.028.

Minieka, E. (1970). The m-center problem. *SIAM Review*, *12*, 138–139. doi:10.1137/1012016.

Paul, J. A., & Wang, X. J. (2015). Robust optimization for united states department of agriculture food aid bid allocations. *Transportation Research Part E: Logistics and Transportation Review*, *82*, 129–146. doi:10.1016/j.tre.2015.08.001.

Paul, J. A., & Wang, X. J. (2019). Robust location-allocation network design for earthquake preparedness. *Transportation research part B: methodological*, *119*, 139–155. doi:10.1016/j.trb.2018.11.009.

Rahmaniani, R., Crainic, T. G., Gendreau, M., & Rei, W. (2017). The benders decomposition algorithm: A literature review. *European Journal of Operational Research*, *259*, 801 – 817. doi:10.1016/j.ejor.2016.12.005.

Serra, D., & Marianov, V. (1998). The p-median problem in a changing network: the case of barcelona. *Location Science*, *6*, 383–394. doi:10.1016/S0966-8349(98)00049-7.

Snyder, L. V. (2006). Facility location under uncertainty: a review. *IIE transactions*, *38*, 547–564. doi:10.1080/07408170500216480.

Trivedi, A., & Singh, A. (2017). A hybrid multi-objective decision model for emergency shelter location-relocation projects using fuzzy analytic hierarchy process and goal programming approach. *International Journal of Project Management*, *35*, 827–840. doi:10.1016/j.ijproman.2016.12.004.

Trivedi, A., & Singh, A. (2019). Shelter planning for uncertain seismic hazards using multicriteria decision approach: a case of nepal earthquake. *Journal of Multi-Criteria Decision Analysis*, *26*, 99–111. doi:10.1002/mcda.1665.

Zeng, B., & Zhao, L. (2013). Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters*, *41*, 457–461. doi:https://doi.org/10.1016/j.orl.2013.05.003.