RAIRO-Oper. Res. 55 (2021) 141–165
https://doi.org/10.1051/ro/2020024

RAIRO Operations Research
www.rairo-ro.org

LOWER AND UPPER BOUNDS FOR THE CONTINUOUS SINGLE FACILITY LOCATION PROBLEM IN THE PRESENCE OF A FORBIDDEN REGION AND TRAVEL BARRIER

Intesar M. Al-Mudahka*, Marwa S. Al-Jeraiwi and Rym M'Hallah

Abstract. In this paper, we investigate FRB, which is the single facility Euclidean location problem in the presence of a (non-)convex polygonal forbidden region where travel and location are not permitted. We search for a new facility's location that minimizes the weighted Euclidean distances to existing ones. To overcome the non-convexity and non-differentiability of the problem's objective function, we propose an equivalent reformulation (RFRB) whose objective is linear. Using RFRB, we limit the search space to regions of a set of non-overlapping candidate domains that may contain the optimum; thus we reduce RFRB to a finite series of tight mixed integer convex programming sub-problems. Each sub-problem has a linear objective function and both linear and quadratic constraints that are defined on a candidate domain. Based on these sub-problems, we propose an efficient bounding-based algorithm (BA) that converges to a (near-)optimum. Within BA, we use two lower and four upper bounds for the solution value of FRB. The two lower and two upper bounds are solution values of relaxations of the restricted problem. The third upper bound is the near-optimum of a nested partitioning heuristic. The fourth upper bound is the outcome of a divide and conquer technique that solves a smooth sub-problem for each sub-region. We reveal *via* our computational investigation that BA matches an existing upper bound and improves two more.

Mathematics Subject Classification. 90B85, 9008, 90C26.

Received September 22, 2019. Accepted February 22, 2020.

1. Introduction

Facility location problems appear in many challenging real-world applications: supply chain management, distribution management, transportation, health, and telecommunication networks. They occur in plant layouts, in location of schools, hospitals, fire stations, garbage damps, utility plants, and manufacturing cells, in design of circuit boards and pipe networks for ships or buildings, in routing of communication networks and robots, etc. Some of these facility location problems impose limitations on the location. For example, it is impossible to locate a new facility in one or more regions or to travel through certain regions such as lakes, mountains, and military areas. These restrictions give rise to restricted facility location problems with travel permitted (FR) and with barriers to travel (FRB).

Keywords. Facility location, Euclidean distance, forbidden region, reformulation, divide and conquer, nested partitioning.

[1] Department of Statistics and Operations Research, College of Science, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait.
*Corresponding author: intesarm@hotmail.com

In this paper, we tackle FRB, a single facility location problem with one (non-)convex polygonal forbidden region R_F and barrier to travel, where we measure distances using the Euclidean norm. Given a continuous two-dimensional space and a set of existing facilities characterized by their positive weights, we search for a new facility's location that minimizes the total weighted Euclidean distance between the new facility and existing ones. The new facility can't be located in $\text{Int}(R_F)$, the interior of R_F. Similarly, travel across R_F is prohibited. For all practical purposes, a facility is mapped to a point on the plane as its size is negligible.

Despite its relevance, FRB is difficult. It is non-differentiable and non-convex due to the Euclidean distance measure. It is generally approached heuristically, with emphasis on identifying local optima using an iterative procedure that solves a series of unconstrained problems [6] or on finding "efficient" bounds [13]. Some of these bounds are based on the relaxation of the location and travel constraints; relaxations that result in convex but non-differentiable problems.

In this paper, we propose an approach that optimizes FRB while overcoming FRB's non-differentiability. Using this approach, we reduce the search space by excluding any region that can't contain the optimum; thus, we decrease the computational effort, in particular when R_F is non-convex. We further divide the reduced search space into a finite set of convex sub-regions (candidate domains). For each distinct polygonal sub-region, we rewrite FRB as an equivalent differentiable convex reformulation (RFRB), with tighter constraints, a linear objective, but both convex and non-convex constraints. In addition, for any candidate domain, we transform RFRB into a tight mixed integer convex program with a linear objective function, and smoothen its binary non-differentiable relaxation using the differentiable, convex reformulation of the unrestricted Euclidean multi-facility location problem [25].

This convex reformulation has a linear function and both linear and quadratic constraints. It maps a sub-region into a convex set by recognizing that the optimal distance of a point X (within the sub-region) to an existing facility that is invisible to X is the shortest of two dominant paths. The length of either path is the sum of a variable term corresponding to the Euclidean distance between X to a vertex of R_F, and a constant term representing the length of the shortest distance from this vertex to the existing facility [6]. The two vertices of R_F are those that delimit the R_F path blocking the visibility of X. The length of the dominant path is therefore convex: It is the sum of a convex function and a constant. In addition, this reformulation overcomes the non-differentiability of the distance expression when the location of the new facility coincides with a vertex of R_F.

In summary, we map FRB to a series of differentiable, convex mixed integer programming (MINLP) sub-problems, with linear objectives, convex constraints, and convex regions whose points share identical visibility features with respect to existing facilities and to R_F's vertices. We solve these sub-problems using an MINLP solver. We use their solution values to derive two lower and four upper bounds for FRB. The two lower and two upper bounds are the result of FRB's relaxations, which drop the location and travelling constraints. The third upper bound is the result of a nested partitioning (NP) based heuristic. NP converges to (near-)global optima [27]. It maintains both a global and a local view on FRB's domain. It partitions the feasible region and investigates each region, but backtracks when it deems that it is no longer searching in the most promising region. The fourth upper bound is the result of an exact divide and conquer (DC) technique. In DC, we only solve non-dominated RFRB sub-problems; *i.e.*, its least cost solution is a global optimum for FRB. When we halt DC before exploring all regions, DC returns a tight upper bound.

We explore these aforementioned bounds in a bounding-based algorithm (BA). We benefit BA from DC's exact search while avoiding the computational effort of determining the expressions of the linear boundaries of each sub-region. We use DC as a local search in the neighborhood of the incumbent. That is, we curtail DC's search to the most promising regions; *i.e.*, to those sharing a border with the incumbent. Our application of BA to new and benchmark instances supports the good quality of the obtained solutions and the tightness of the bounds. BA improves two existing upper bounds and matches another.

In Section 2, we review pertinent literature on FR, FRB, and NP. In Section 3, we define and model the problem. In Section 4, we describe the proposed solution approach, which converts the non-convex non-differentiable FRB into a finite set of mixed-integer differentiable convex programs that are solved using off-the-shelf software.

In Section 5, we report the computational results, which highlight the tightness of the lower and upper bounds. Finally, in Section 6, we summarize the work and present possible extensions.

2. Literature review

In Sections 2.1 and 2.2, we preview the literature on the Euclidean unrestricted and restricted single facility location problems. In Section 2.3, we present NP. Finally, in Section 2.4, we highlight our contribution.

2.1. On Euclidean single facility location problems

The earliest version of the single facility location problem is the classic Weber problem (WP). It searches for a new facility's location that minimizes the sum of the weighted Euclidean distances between the new facility and existing ones. Kuhn [19] proves that the optimal location must lie in the convex hull of existing facilities. The convex hull of a set of vertices is the smallest convex polygon that contains them. For WP, Weiszfeld [34] proposes a fixed-point iterative approach, known as Weisfeld's algorithm. Katz and Cooper [16] outperform Weiszfeld's algorithm using a gradient method with an inexact quadratic fit based line-search. Drezner [9] presents a sensitivity analysis for WP. Love *et al.* [20] prove the convexity of WP's objective function. Convexity insures that any local optimum is a global optimum.

Several approaches address the non-differentiability of the Euclidean norm function. Francis *et al.* [10] add an epsilon term to the distance function and develop a hyperploid approximation that converges to the optimum *via* a steepest descent. Sherali and Al-Loughani [25] present two equivalent convex differentiable reformulations that can be solved using standard nonlinear solvers. Their first reformulation has a linear objective function and both linear and quadratic constraints. Even though the quadratic constraints are nonconvex, the overall feasible region is a convex set. Their second reformulation is based on Lagrangian duality. Sherali and Al-Loughani [26] propose a conjugate sub-gradient algorithm with line search strategies embedded within the variable target value method.

Torres [32] considers a WP constrained to a closed convex set. He combines the iterative Weiszfeld's function with an iterative orthogonal projection over the feasible set. Ghaderi *et al.* [11] address the uncapacitated location allocation WP heuristically. Kazakovtsev [17] offers a heuristic for the single-facility constrained WP with the connected feasible region bounded by arcs of equal radii. The heuristic augments Weiszfeld's algorithm with a procedure that determines the closest feasible solution to a point.

2.2. On the restricted single facility location problem

Herein, we classify the literature according to the problem type: FR and FRB. Aneja and Parlar [2] stipulate that if X^*_{WP}, the optimum of the unrestricted WP, lies in the interior of R_F, then FR's optimum lies on the R_F's boundary that is closest to X^*_{WP}. They then propose an approximate algorithm for FR when R_F is a convex polygon. They further provide an exact algorithm when R_F consists of non-convex polygons. Butt and Cavalier [7] study the rectilinear unweighted multiple facility FR in the presence of congested non-intersecting convex polygonal regions and no existing facility is in R_F. They transform the problem into an unconstrained network problem that is further reduced *via* a combinatorial search.

Klamroth [18] offers an exhaustive survey of FRB approaches and applications. Katz and Cooper [14–16] address FRB for three types of R_Fs: a single circle, non-intersecting circles, and a single convex polygon. They prove that FRB is non-convex non-linear, and has a discontinuous objective function, and a non-convex constraint set. They infer that the optimum lies within the convex hull of existing facilities and vertices of R_F. They employ constrained optimization for the first two cases and a discrete search for the third case. The search chooses the least cost location among points of a grid. Butt [5] shows that a non-convex R_F can be replaced by its convex hull Conv(R_F) as long as no existing facility lies in the interior Conv(R_F). Aneja and Parlar [2] apply simulated annealing to FRB under general *lp*-metric distances and (non-)convex polygonal R_Fs. They assimilate the problem to a network that is represented *via* a visibility graph.

A visibility graph is an undirected graph $G(N, E)$ whose set of nodes $N = V \cup P$ where $P = P_1, \ldots, P_n$ is the set of existing facilities and V is the set of vertices of R_F. Its set of arcs E includes all arcs that connect any two visible nodes of N, where a node may be an existing facility $P_i \in P$ or a vertex $v \in V$. If the direct line segment connecting two locations does not pass through $\text{Int}(R_F)$, then an arc connects them in the graph. It follows that E connects pairs of nodes of N without intersecting $\text{Int}(R_F)$ and $G(N, E)$ is a graph of inter-visible locations for a set of points and obstacles. $G(N, E)$ can be constructed using the method of Wangdahl *et al.* [33] or any of the algorithms mentioned in Ghosh and Mount [12]. When R_F is non-convex and none of the existing facilities lies in the interior of $\text{Conv}(R_F)$, we ease the computational burden by using V', the set of vertices of $\text{Conv}(V)$, in lieu of V. Indeed, V' has fewer vertices than V. Evidently, when R_F is convex, $V' = V$. In Algorithm 1, we describe how we generate FRB's visibility graph.

Algorithm 1. Visibility graph for FRB.

1: Input

- V: set of vertices $v_f \in R_F$, $f = 1, \ldots, F$;
- P: set of existing facilities $P_i \in P$, $i = 1, \ldots, n$;
- X^*_{WP}: relaxed solution of WP;

2: Let $N = V \cup P$ and $E = \emptyset$;
3: Construct the set O of all pairs of points (τ, π), such that $\pi \in P$, $\tau \in N$, and $\tau \neq \pi$;
4: Select one pair (τ, π) from O;
5: If the line joining π to τ doesn't intersect $\text{Int}(R_F)$, then append arc (τ, π) to E;
6: Discard pairs (τ, π) and (π, τ) from O;
7: If $O \neq \emptyset$, go to Line 4;
8: For each arc $(\tau, \pi) \in G(N, E)$, compute the Euclidean distance between τ and π;
9: **return** $G(N, E)$: Visibility graph;

In Figure 1, we give the visibility graph of an instance with three existing facilities $P = \{P_1, P_2, P_3\}$ and with a rectangular forbidden region defined by $V = \{v_1, v_2, v_3, v_4\}$. The visibility graph is delimited by $\text{Conv}(P)$; *i.e.*, by the triangle (P_1, P_2, P_3). The set of nodes $N = \{P_1, P_2, P_3, v_1, v_2, v_3, v_4\}$. The set E of arcs consists of the arcs connecting every pair of visible nodes; *e.g.*, P_1 is visible to v_1; thus an arc connects P_1 to v_1. But, P_1 is invisible to v_3; thus no arc connects P_1 to v_3.

Aneja and Parlar [2] employ Dijkstra's algorithm on the visibility graph to find the shortest path between any candidate location and existing facilities. They reduce the set of candidate solutions and infer an important result when X^*_{WP}, the optimum of WP, lies in R_F (WP is a relaxation of FRB.) When R_F is convex, FRB's optimum lies on the nearest boundary of R_F to X^*_{WP}; otherwise, it may only lie on the boundary points that may be joined to X^*_{WP} by a line segment that is embedded in R_F.

Butt and Cavalier [6] consider FRB with convex polygonal R_Fs. They decompose the solution space into regions such that the shortest feasible path from any point of the region to an invisible existing facility is unique and passes through a fixed visible intermediate vertex of R_F. For each region, they formulate FRB as an MINLP and solve it. They then choose the best solution among all regions. Nevertheless, not all boundaries are linear; consequently, the convexity of the regions is not guaranteed. In addition, determining the mathematical expression of every boundary is difficult. Consequently, they propose a heuristic FORBID, which starts with a good solution and iteratively solves a series of unconstrained WPs.

Bischoff and Klamroth [3] improve FORBID. They slightly modify the partitioning of FRB's domain. They develop a visibility gridline's partitioning that generates convex sub-regions with linear boundaries, denoted candidate domains. All points of a candidate domain share the same visibility properties with respect to existing facilities and vertices of R_Fs. The authors reduce FRB to a finite series of NLMIPs; each defined on a specific candidate domain and having a nonlinear objective function. The NLMIPs use binary variables that choose the correct distance function of a point to an existing facility, where only dominant paths are considered. Because of the complexity of the sub-problems, the authors apply a genetic algorithm.

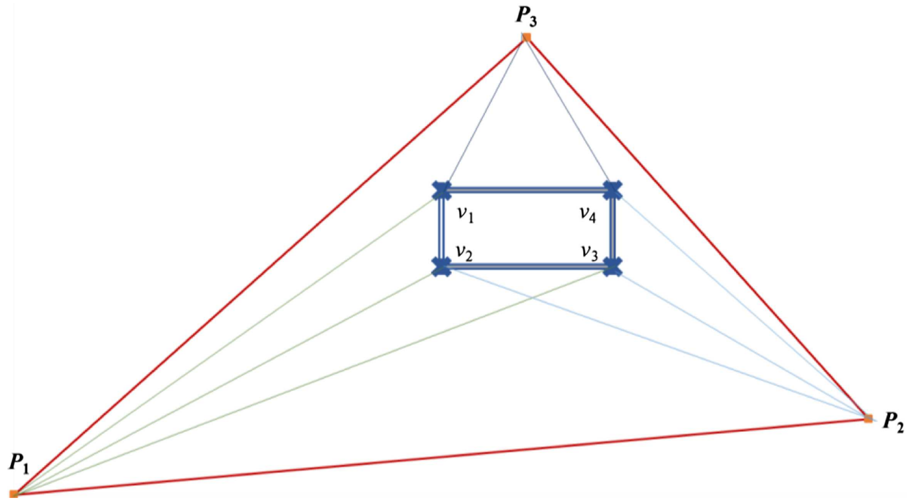

FIGURE 1. Visibility graph of an instance with three existing facilities and a rectangular forbidden region.

For FRB with non-intersecting convex polygonal R_Fs and no existing facility located within any R_F, McGarvey and Cavalier (2003) use an exact procedure that divides the plane into discrete regions. Satyanarayana *et al.* [23] apply a sloping line search to FRB. Prakash *et al.* [22] propose a reduced search for FRB with polygonal and elliptical R_Fs.

2.3. On nested partitioning

NP is a randomized search for the global optimization of difficult large-scale combinatorial problems [27, 28, 30, 31]. It imitates human's tackling of complicated problems [28]. When the search space is large, humans divide it into regions, and investigate each region separately [1]. They use sampled information to decide whether to further investigate a particular region or to consider an alternative one. Similarly, NP focuses on the most promising region of the search space while maintaining a global perspective. It either shrinks the most promising region or backtracks when it deems that it may have missed the region containing the global optimum. The most promising region is determined in each iteration based on information obtained from random sampling of the entire feasible region and from local search. In each iteration, NP performs four steps.

(1) Partitioning divides the feasible region into mutually disjoint $M+1$ sub-regions, whose union covers the feasible search space and whose intersection is nil. The partitioning is not uniform: It favors the most promising region by partitioning it into M sub-regions, and aggregates all other regions into region $M+1$.
(2) Random sampling selects arbitrarily a point from each of the $M+1$ sub-regions. It can be uniform or biased towards solutions that are more likely to lead to the optimum. Based on the $M+1$ random samples, NP assesses the potential of each region to contain the global optimum.
(3) Selecting the most promising region is based on a comparison of the values of the sampled points.
(4) Finally, NP chooses either further partitioning or backtracking. If the most promising region is one of the M sub-regions, then NP further partitions this sub-region. Otherwise, it backtracks to region $M+1$.

The four steps may be implemented generically or adapted to the structure of the problem. Despite its wide spectrum of applications in planning and scheduling, logistics and transportation, supply chain design, data mining, and health care, NP hasn't been applied to facility location problems.

TABLE 1. Related algorithms in the literature.

	Existing literature	Proposed algorithm	
	Description	Similarity	Difference
[2]	When the solution of the unrestricted problem WP lies in R_F, the optimal location of the new facility lies on a specific boundary of R_F.	=	
[5]	If no existing facility lies in Conv(R_F), R_F can be replaced by Conv(R_F).	=	
[25]	A differentiable convex reformulation for the unrestricted multi-facility location problem in the primal space.	Incorporates the reformulation for the visible existing facilities.	The presence of an R_F.
[6]	Used a visibility graph. R_F is convex.	=	R_F may be non-convex.
	Partitioned the problem's domain into mutually exclusive regions whose borders may be nonlinear and non-convex. In each sub-region, the path to an invisible existing facility is unique.	Decomposes the solution space into mutually exclusive regions.	Larger sub-regions that are convex and whose linear boundaries are based on the visibility grids of [3].
	Distance function based on the visible vertices of R_F. Decomposed the shortest travelling feasible path to an invisible existing facility into a variable and a constant part.	=	
	Equivalent reformulation: MIP with nonlinear objective function.		Tighter NLP with a linear objective function and (non-)linear differentiable convex constraints and non-linear non-convex constraints.
	Problem decomposed into a finite series of convex unconstrained sub-problems that are non-differentiable at the visible vertices of R_F.	=	Tighter MINLP sub-problems with linear objective function and (non-)linear constraints. Its binary relaxation is smooth and differentiable.
	Iterative solution procedure: FORBID: Solves a series of unconstrained WPs. Requires a very good starting solution. Terminates at a local optimum of FRB.		Bounding Algorithm uses two lower and four upper bounds sequentially to improve the incumbent. Does not require a starting solution. Uses NP only when needed. DC applied only on the most promising regions. DC solves the smooth reformulation.
	A local optimum.		A near-global optimum is guaranteed. When an optimum is sought, BA applies DC.
[18]	Developed relaxation based lower and upper bounds.	=	Bounds based on the relaxation of the equivalent reformulation.
[23]	Applied a sloping line search to FRB.	(Non-)convex R_F. Two test problems.	Improved solution for the two test problems.
[3]	A convex R_F. Proposed visibility grids partitioning. Developed visibility properties. Introduced the notion of dominant paths and the touching point property.	=	Proved that the number of feasible dominating paths is two.
	Iterative solution approach. Used genetic algorithm to select the sub-problems. Solved a series of convex MIP sub-problems with nonlinear objective function.	Applied heuristics.	NP is applied to improve the incumbent.

2.4. Contribution

The surveyed literature reflects the difficulty of FRB. It lacks an efficient exact algorithm for FRB, whose objective is non-convex and non-differentiable. In Table 1, we list FRB related approaches.

In Table 1, we ordered the related literature by publication year and specified their similarities/differences with respect to the proposed approach. Based on this comparison, we explore, in this paper, the results of Aneja and Parlar [2], Butt [5], and Sherali and Al-Loughani [25]. We follow the streamline of Butt and Cavalier [6] and Bischoff and Klamroth [3] except that we consider a non-convex R_F as well. We avoid the non-differentiability of the reformulation when the optimal location of the new facility coincides with one of the visible vertices of R_F. Furthermore, our equivalent reformulation is tighter and has a linear objective. The relaxation of the

MINLP sub-problems is convex and smooth. Every sub-problem corresponds to a convex region obtained from the visibility grid lines partitioning. Its distance constraints consider only the two dominant feasible paths to the invisible existing facility. We obtain the global minimum by solving all sub-problems.

In this paper, our two lower bounds and two upper bounds are the result of relaxations of the equivalent RFRB and not of FRB as in Klamroth [18]. Our third upper bound is an NP's near-optimum. Our fourth upper bound is the result of DC, which solves the smooth sub-problems on every region generated from the visibility grid lines. We incorporate the bounds in a bounding algorithm, which applies DC to the most promising regions only; *i.e.*, those that share borders with the current incumbent.

3. Model formulation

In Section 3.1, we describe FRB, its reduced solution space, and the shortest travelling distance between two points under the visibility concept. In Section 3.2, we present the differentiable convex reformulation of the FRB sub-problems, and give related algorithms.

3.1. Problem definition

Herein, we consider the Euclidean distance single facility location problem in the presence of a forbidden region R_F and travel barriers. Let existing facility $P_i,\ i = 1, \ldots, n$, be defined by its coordinates (a_i, b_i), and by its strictly positive weight w_i. Let R_F be a (non-)convex polygonal set defined by its set V of F vertices. Travelling through and locating the new facility $X^* \in \text{Int}(R_F)$ are forbidden. The goal is to find the coordinates $(x^*, y^*) \in \mathbb{R}^2 \setminus \text{Int}(R_F)$ of X^* such that (x^*, y^*) minimizes the total weighted Euclidean paths to all existing facilities:

$$FRB \quad \min_{X \in \mathbb{R}^2 \setminus \text{Int}(R_F)} \quad f(X) = \sum_{i=1}^{n} w_i d(X, P_i) \tag{3.1}$$

where $d(X, P_i)$ is the traveling distance between P_i and X. In the absence of a forbidden region, $d(X, P_i)$ is the Euclidean distance:

$$d(X, P_i) = \sqrt{(x - a_i)^2 + (y - b_i)^2},$$

and FRB reduces to WP, whose optimum lies in the convex hull of existing facilities. In the presence of a forbidden region, R_F may block the visibility of X to P_i. Therefore, $d(X, P_i)$ is the actual Euclidean travel distance between P_i and X. It is the length of the shortest path that links P_i and X without crossing $\text{Int}(R_F)$. For example, consider Figure 2, where R_F is the triangle (v_1, v_2, v_3), and the dotted lines represent the feasible Euclidean paths from X to P_1 and P_2. The point X is visible to P_1; thus,

$$d(X, P_1) = \sqrt{(x - a_1)^2 + (y - b_1)^2}.$$

On the other hand, X is invisible to P_2; therefore $d(X, P_2)$ is the minimal length of two paths: X, v_2, P_2 and X, v_3, P_2. It follows that

$$d(X, P_2) = \min\{d(X, v_2) + d(v_2, P_2), d(X, v_3) + d(v_3, P_2)\}.$$

In fact, thanks to the "Barrier Touching Property" [3], the shortest feasible path connecting a point X to an invisible P_i is a piecewise linear path with breaking points in vertices v of R_F. We determine the length of the shortest path between every P_i and every R_F's vertex that is visible to P_i using a visibility graph.

The feasible region for FRB is $\mathbb{R}^2 \setminus \text{Int}(R_F)$. However, the master candidate domain MC is a limited subset of the points of $\mathbb{R}^2 \setminus \text{Int}(R_F)$. FRB's optimum lies within $\text{Conv}(P \cup V')$. Hence, any FRB solution procedure should restrict its search to MC.

MC depends on the convexity of R_F and on the position of X^*_{WP}, the optimum of WP.

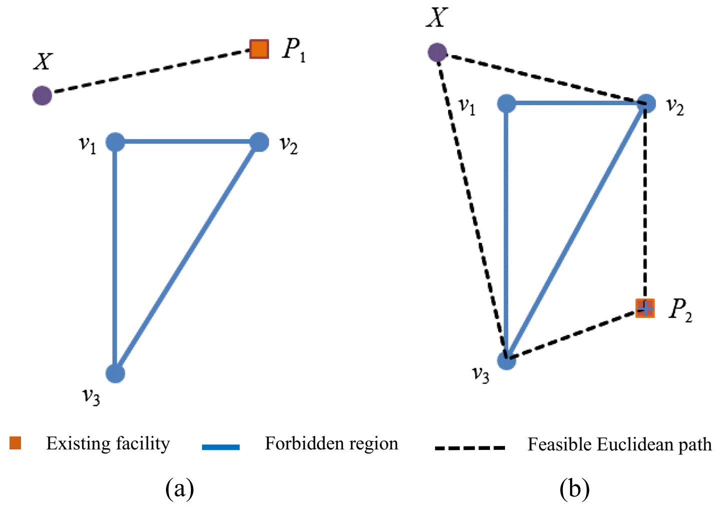

FIGURE 2. Feasible Euclidean paths between X and existing facilities P_1 and P_2. (a) P_1 is visible to X. (b) P_2 is invisible to X.

– When R_F is convex, two cases arise.
  - $X^*_{WP} \in \mathrm{Int}(R_F)$; thus, FRB's optimal solution lies on the nearest boundary of R_F. Subsequently, $\mathrm{MC} = \mathrm{MC_C} = \mathrm{MC_{C1}}$ where $\mathrm{MC_{C1}}$ is the set of points that are on R_F's boundary nearest to X^*_{WP}.
  - $X^*_{WP} \notin \mathrm{Int}(R_F)$; thus, $\mathrm{MC}{=}\mathrm{MC_C} = \mathrm{MC_{C2}} = \mathrm{Conv}(P \cup V) \setminus \mathrm{Int}(R_F)$.

– When R_F is non-convex, two cases also arise.
  - $X^*_{WP} \in \mathrm{Int}(R_F)$; thus, $\mathrm{MC} = \mathrm{MC_{NC}} = \mathrm{MC_{NC1}}$, where $\mathrm{MC_{NC1}}$ is the set of R_F boundary points which can be joined to X^*_{WP} by a line segment that lies entirely in R_F.
  - $X^*_{WP} \notin \mathrm{Int}(R_F)$, two further cases arise.
    - If no existing facility lies in $\mathrm{Int}(\mathrm{Conv}(R_F))$, then $\mathrm{MC_{NC}} = \mathrm{MC_{NC2}}{=}\mathrm{Conv}(P \cup V') \setminus \mathrm{Int}(\mathrm{Conv}(V'))$;
    - otherwise, $\mathrm{MC_{NC}}{=}\ \mathrm{MC_{NC3}}{=}\mathrm{Conv}(P \cup V) \setminus \mathrm{Int}(R_F)$.

Evidently, unless WP is solved, it is impossible to know whether $X^*_{WP} \in \mathrm{Int}(R_F)$.

3.2. Equivalent differentiable reformulation of FRB

The FRB equivalent reformulation partitions MC into mutually exclusive convex candidate domains R_j whose points share the same visibility property and whose boundaries are linear. It defines the length of the shortest distance d_i between an existing facility $P_i,\ i = 1, \ldots, n$, and a point $X \in R_j$ as

$$d_i = \begin{cases} \sqrt{(x-a_i)^2 + (y-b_i)^2}, & \forall P_i \in K \\ \min\limits_{k \in K \cap V} \{s(P_i, k) + d(k, X)\}, & \forall P_i \notin K \end{cases} \tag{3.2}$$

where K is the candidate set; *i.e.*, K contains every node $k \in P \cap V$ that is visible to X. $s(P_i, k)$ is the length of the shortest feasible path from P_i to a vertex $k \in K \cap V$. Naturally,

$$d(X, P_i) \leq d(X, k) + d(k, P_i), \quad \forall P_i \in P,\ \forall k \in K.$$

RFRB reformulates FRB using decision variables $d_i,\ i = 1, \ldots, n$, and (x, y).

$$\text{RFRB} \qquad \min f_{RFRB}(X) = \sum_{i=1}^{n} w_i d_i \tag{3.3}$$

$$\begin{aligned} \textit{subjectto} \qquad d_i^2 i &\geq (x - a_i)^2 + (y - b_i)^2, \quad i = 1, \ldots, n \\ d_i &\geq (x - a_i), \quad i = 1, \ldots, n \\ d_i &\geq (y - b_i), \quad i = 1, \ldots, n \\ d_i &\geq (a_i - x), \quad i = 1, \ldots, n \\ d_i &\geq (b_i - y), \quad i = 1, \ldots, n \end{aligned} \tag{3.4}$$

$$d_i = \min_{\forall k \in K \cap V} \{s(P_i, k) + d(k, X)\} \quad \forall P_i \notin K \tag{3.5}$$

Equation (3.3) minimizes the sum of the weighted distances. Equation (3.4) impose lower bounds on the distances between a feasible location X and a facility P_i. The distance must be at least as large as the Euclidean distance, and as the absolute values of the horizontal and vertical distances that separate X and P_i. Equation (3.5) chooses, among all candidate paths joining X to an invisible existing facility P_i, the smallest length path. This path must pass through an R_F vertex that is visible to P_i.

Theorem 3.1. *RFRB is a convex equivalent restatement of FRB.*

Proof. Suppose $\bar{X} = (\bar{x}, \bar{y})$ solves FRB. When $\bar{d}$ is computed *via* equation (3.2), $(\bar{x}, \bar{y})$ solves RFRB. Because it is linear, the objective function of RFRB is convex and differentiable. Constraints (3.5) define a non-convex set [6,18]. RFRB without equation (3.5) reduces to the unrestricted problem of Sherali and Al-Loughani [25], who proved that despite the non-convexity of the quadratic constraints (3.4), the overall feasible region is a convex set. In fact, the set of points

$$S_i = \left\{(x, y, d_i) : (x - a_i)^2 + (y - b_i)^2 \leq d_i^2,\ d_i \geq 0\right\}$$

can be equivalently represented as

$$S_i = \left\{(x, y, d_i) : |\sqrt{(x - a_i)^2 + (y - b_i)^2}| \leq d_i\right\},$$

which is a convex set. Subsequently, RFRB without (3.5) is a convex program. □

Solving RFRB faces three challenges:

(1) Defining the constraints of RFRB depends on the visibility property of X. It requires defining the candidate set K, which depends on the location of X with respect to the existing facilities and R_F.
(2) The number of feasible paths that must be considered for constraint (3.5) increases as the number of visible vertices of R_F and the number of invisible existing facilities increase.
(3) The non-convexity and the non-differentiability of constraint (3.5) prevent optimization solvers from reaching a global optimum.

These difficulties are handled as follows.

3.2.1. Determining K

First, we decompose the candidate domain MC into J convex regions $R_j,\ j = 1, \ldots, J$, such that all points of a region R_j share the same visibility property but no two regions have identical visibility properties. For this decomposition, we use the visibility grid line's partitioning of Algorithm 2. We divide MC according to the boundaries of P_i's shadow, which is the set of feasible points that are invisible to P_i. When no $P_i \in P$ lies in Conv(V) and R_F is non-convex, we extend the facets of Conv(V) to bisect Conv$(P \cup V)$; otherwise, we extend

Algorithm 2. Partitioning scheme.

1: Input

- V: set of vertices $v_f \in R_F$, $f = 1, \ldots, F$;
- P: set of existing facilities $P_i \in P$, $i = 1, \ldots, n$;
- V': set of vertices of $\text{Conv}(R_F)$;

2: **if** (R_F is convex) or (R_F is non-convex but no existing facility lies in $\text{Int}(R_F)$) **then**
3:     Bisect $\text{Conv}(P)$ with the boundaries of the shadow of each facility P_i, $i = 1, \ldots, n$, with respect to the vertices of V';
4:     Extend each facet of $\text{Conv}(R_F)$ until the borders of $\text{Conv}(P)$;
5: **else**
6:     Bisect $\text{Conv}(P)$ with the boundaries of the shadow of each facility P_i, $i = 1, \ldots, n$, with respect to the vertices of V;
7:     Extend each facet of R_F until the borders of $\text{Conv}(P)$;
8: **end if**
9: **return** R_j, $j = 1, \ldots, J$: Candidate domains of the solution space;

the facets of R_F. In the former case, we partition MC_{NC} instead of partitioning FRB's solution space because $\text{Conv}(V)$ has fewer facets than R_F. In the following, we present the pseudocode of Algorithm 2, which is the partitioning algorithm of [3]. Evidently, when R_F is convex, $\text{Conv}(R_F) = R_F$.

With MC partitioned, we determine using Algorithm 3, for each region R_j, the candidate set K_j of all existing facilities and R_F's vertices that are visible to R_j's points. We choose a point $r \in R_j$ and identify the existing facilities and R_F's vertices that are visible to r. When the line joining r to an existing facility P_i (or to a vertex $v \in V$) intersects $\text{Int}(R_F)$, then P_i (or v) is invisible to r; otherwise it is visible.

Algorithm 3. Constructing the candidate set K_j.

1: Input

- V: set of vertices $v_f \in R_F$, $f = 1, \ldots, F$;
- P: set of existing facilities $P_i \in P$, $i = 1, \ldots, n$;
- R_j, $j = 1, \ldots, J$: Candidate domains of the solution space;

2: Set $K_j = \emptyset$;
3: Choose a random point $r \in R_j$;
4: **for** $P_i \in P$ **do**
5:     **if** the line joining r to P_i intersects $\text{Int}(R_F)$ **then**
6:         P_i is invisible to R_j;
7:     **else**
8:         add P_i to K_j;
9:     **end if**
10: **end for**
11: **for** $v \in V$ **do**
12:     **if** the line joining r to v intersects $\text{Int}(R_F)$ **then**
13:         v is invisible to R_j;
14:     **else**
15:         add v to K_j;
16:     **end if**
17: **end for**
18: **return** K_j, $j = 1, \ldots, J$: candidate set of region R_j;

For illustrative purposes, consider the example of Figure 3a. This instance has four existing facilities: $P = \{P_1, P_2, P_3, P_4\}$, and a triangular R_F with $V = \{v_1, v_2, v_3\}$. In Line 3 of Algorithm 3, we bisect the boundary of the shadow of each existing facility with the borders of $\text{Conv}(P)$. In Line 4, we extend the three facets $\overline{v_1 v_2}$, $\overline{v_2 v_3}$, and $\overline{v_3 v_1}$ to reach the convex hull's border as we show in Figure 3b. Based on these bisections, we divide MC_{C} into 19 candidate domains, where each candidate domain is convex with linear boundaries. Using Algorithm 3, we determine the visibility properties of adjacent candidate domains R_1 and R_2. Candidate sets $K_1 = \{P_1, P_2, P_4, v_1, v_2, v_3\}$ for R_1 and $K_2 = \{P_1, P_2, P_4, v_1, v_2\}$ for R_2 are different.

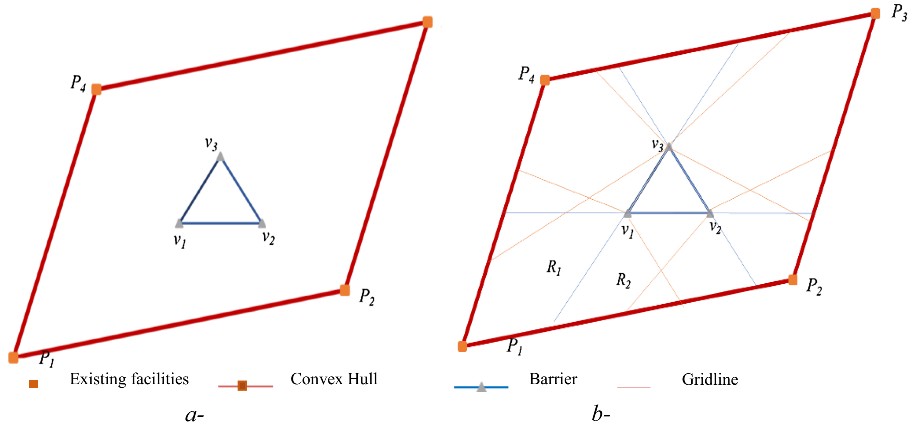

FIGURE 3. Partitioning the solution space. (a) Convex hull. (b) Division steps.

For each region R_j, $j = 1, 2, \ldots, J$, characterized by its candidate set K_j, equation (3.5) is equivalent to

$$d_i = \min_{\forall k \in K_j \cap V} \{s(P_i, k) + d(k, X)\}, \quad j = 1, 2, \ldots, J. \tag{3.6}$$

In this way, our decomposition handles the first difficulty associated with RFRB.

Second, with K_j, $j = 1, \ldots, J$, at hand, we solve RFRB for each region R_j and obtain its solution value $f^*_{SRFRB}(R_j)$. Because (i) R_j is the intersection of visibility grid lines, (ii) any visibility grid line is convex, (iii) the intersection of convex sets is convex, and (iv) the solution space is bounded by the convex hull whose boundaries are linear, R_j is convex with linear boundaries:

$$R_j = \{X \in \mathbb{R}^2 : g_{jh}(X) \leq 0,\ h = 1, \ldots, m_j\}, \quad j = 1, 2, \ldots, J \tag{3.7}$$

where $g_{jh} : \mathbb{R}^2 \longrightarrow \mathbb{R}$ for $h = 1, \ldots, m_j$, and m_j is the number of constraints defining the boundaries of R_j. Therefore, sub-problem SRFRB(R_j) is differentiable and convex.

Third and last, we deduce the global minimal cost of f_{RFRB}:

$$f^*_{RFRB} = \min_{j=1,\ldots,J} \{f^*_{SRFRB}(R_j)\}. \tag{3.8}$$

3.2.2. Reducing the number of feasible paths

The number of feasible paths that satisfy the barrier touching property (BT) is large. When candidate domain R_j has several feasible paths to an invisible facility P_i, Bischoff and Klamroth [3] suggest deleting dominated ones; *i.e.*, paths having at least two intermediate breaking points in K_j. Deleting dominated paths reduces the set $K_j \cap V$ of equation (3.6) to a set RK_{ji} that includes only vertices in $K_j \cap V$. RK_{ji} consists of the last visible breaking points in the shortest feasible paths that satisfy BT from some invisible P_i to the points of R_j. We summarize the construction of RK_{ji} in Algorithm 4.

To illustrate the application of Algorithm 4, we consider the candidate domain R_1 of Figure 3b. P_3 is invisible to any point $X \in R_1$. There are four paths that satisfy BT and that connect P_3 to any point $X \in R_1$. These paths go through the vertices v_1, v_2, v_3; *i.e.*, initially, $RK_{13} = K_1 \cap V = \{v_1, v_2, v_3\}$, as in Line 3. These paths are $X - v_1 - v_3 - P_3$, $Xv_1 - v_2 - P_3$, $X - v_2 - P_3$, and $X - v_3 - P_3$. For the first (resp. second) path, nodes v_1 and v_3 (resp. v_1 and v_2) are breaking points despite being members of candidate set K_1. These two paths are longer than the paths passing through $v_2 - P_3$, or $v_3 - P_3$, respectively. We therefore delete them as per Line 6. Subsequently, v_2 and v_3 are the appropriate intermediate points on the path from any point $X \in R_1$ to facility P_3. Consequently, $RK_{13} = K_1 \cap V = \{v_2, v_3\}$. For any invisible $P_i \in P$ of a candidate domain R_j, set RK_{ji} has 2 vertices, as follows.

Algorithm 4. Constructing RK_{ji} for a candidate domain R_j.

Input

– $R_j,\ j = 1,\ldots,J$: Candidate domains of the solution space;
– $K_j,\ j = 1,\ldots,J$: Candidate sets associated to R_j;

for $P_i \notin K_j$ **do**
  Define set $RK_{ji} = K_j \cap V$;
  Choose a random point $r \in R_j$;
  if a path from r_j to P_i shares at least two vertices with RK_{ji} **then**
    delete these vertices from RK_{ji};
  end if
end for
return $RK_{ji},\ j = 1,\ldots,J,\ i = 1,\ldots,n$: Reduced candidate sets RK_{ji} with respect to $P_i \notin K_j$;

Lemma 3.2. *Let K_j be the candidate set for a candidate domain R_j, then the reduced candidate set RK_{ji} with respect to an existing facility P_i includes only two intermediate points.*

Proof. Take a vertex $v \in RK_{ji}$. By definition, v is an intermediate point; *i.e.*, v is the last visible breaking point for any shortest path satisfying BT from any invisible P_i to the points of R_j. Hence v is the only visible vertex in RK_{ji} that belongs to one side of RF. Because FRB is solved in a two-dimensional space, only two sides of RF are considered. The second point of RK_{ji} has the same features but is on the other side of R_F. Thus RK_{ji} has two points. □

Using Lemma 3.2, there are only two dominant paths from any invisible existing facility P_i to the points of a candidate domain R_j, where each path passes through an intermediate point of RK_{ji}. This reduces the computational time for solving the RFRB sub-problems whose equations (3.6) are replaced by:

$$d_i = \min_{\forall k_s \in RK_{ji},\ s=1,2} \{c_{k_s i} + d_{Xk_s}\}, \quad \forall P_i \notin K_j,\ j = 1,\ldots,J. \tag{3.9}$$

Decision variable d_i is the minimal length of two feasible paths from points of R_j to an invisible existing facility P_i. Each path passes through a visible vertex k_s that belongs to RK_{ji}. The cost of a path is the sum of a variable component d_{Xk_s} and a fixed component $c_{k_s i}$. d_{Xk_s} is the Euclidean distance from $X \in R_j$ to k_s, whereas $c_{k_s i}$ is the length of the feasible shortest path from k_s to P_i. Both d_{Xk_s} and $c_{k_s i}$ are computed *via* the visibility graph and Dijkstra's Algorithm. $c_{k_s i}$ is constant for a given set RK_{ji}.

To choose the dominant path from the two alternative paths connecting $P_i \notin K_j$ to $X \in R_j$, we solve an assignment problem whose binary variables $l_{ji} = 1$ if the first intermediate point $k_1 \in RK_{ji}$ yields the shortest path, and 0 otherwise. Using l_{ji}, we rewrite equation (3.9) as

$$\begin{aligned} &d_i \le d_{Xk_s} + c_{k_s i} \\ &d_i \ge d_{Xk_s} + c_{k_s i} - L * l_{ji} \\ &d_i \ge d_{Xk_s} + c_{k_s i} - L * (1 - l_{ji}) \quad s = 1,2, \forall P_i \notin K_j,\ j = 1,\ldots,J, \end{aligned} \tag{3.10}$$

where L is a large positive number that is larger than $d_{Xk_s} + c_{k_s i}$ for any feasible solution X. For example, we can set L to the Euclidean distance between the farthest extreme points of Conv(P). In the first constraint of equation (3.10), we set an upper bound on d_i. In the second and third constraints, we choose the shortest path and subsequently set the optimal value of d_i. Because $c_{k_s i},\ s = 1,2$, is constant, minimizing d_i is equivalent to minimizing the Euclidean distance d_{Xk}. The smallest $d_{Xk} = 0$ when $X = k_s$; *i.e.*, when the location of the new facility coincides with vertex $k_s = (a_{k_s}, b_{k_s})$. Therefore, d_{Xk} is convex but non-differentiable when $X = k_s$. To avoid this non-differentiability, we treat d_{Xk} as in equation (3.5). Let $k_1 = (a_{k_1}, b_{k_1}) \in RK_{ji}$ and

$k_2 = (a_{k_2}, b_{k_2}) \in RK_{ji}$, then we rewrite constraint (3.9) as:

$$
\begin{aligned}
&d^2_{Xk_s} \geq (x - a_{k_s})^2 + (y - b_{k_s})^2, && \forall P_i \notin K_j, && (3.11)\\
&d_{Xk_s} \geq (x - a_{k_s}) && \forall P_i \notin K_j, &&\\
&d_{Xk_s} \geq (y - b_{k_s}) && \forall P_i \notin K_j, &&\\
&d_{Xk_s} \geq (a_{k_s} - x) && \forall P_i \notin K_j, &&\\
&d_{Xk_s} \geq (b_{k_s} - y) && \forall P_i \notin K_j, &&\\
&\quad d_i \leq d_{Xk_s} + c_{k_s i} && s = 1,2, \quad \forall P_i \notin K_j, \quad \forall R_j, && (3.12)\\
&\quad d_i \geq d_{Xk_s} + c_{k_s i} - L * l_{ji} && s = 1,2, \quad \forall P_i \notin K_j, \quad \forall R_j &&\\
&\quad d_i \geq d_{Xk_s} + c_{k_s i} - L * (1 - l_{ji}) && s = 1,2, \quad \forall P_i \notin K_j, \quad \forall R_j && (3.13)
\end{aligned}
$$

where decision variable $l_{ji} \in \{0, 1\}$. Therefore, we reformulate a sub-problem of RFRB on R_j as:

$$
\begin{aligned}
\text{SRFRB}(R_j) \qquad & \min f_{SRFRB}(R_j) = \sum_{i=1}^{n} w_i d_i && (3.14)\\
\textit{subjectto} \qquad & d_i^2 \geq (x - a_i)^2 + (y - b_i)^2, \quad \forall P_i \notin K_j\\
& d_i \geq (x - a_i), \quad \forall P_i \notin K_j\\
& d_i \geq (y - b_i), \quad \forall P_i \notin K_j\\
& d_i \geq (a_i - x), \quad \forall P_i \notin K_j\\
& d_i \geq (b_i - y), \quad \forall P_i \notin K_j\\
& d_i \geq 0, \quad \forall P_i \notin K_j\\
& d_i = \min_{\forall k \in K \cap V} \{s(P_i, k) + d(k, X)\} \quad \forall P_i \notin K\\
& d^2_{Xk_s} \geq (x - a_{k_s})^2 + (y - b_{k_s})^2, \quad \forall P_i \notin K_j,\ s = 1,2\\
& d_{Xk_s} \geq (x - a_{k_s}) \quad \forall P_i \notin K_j,\ s = 1,2\\
& d_{Xk_s} \geq (y - b_{k_s}) \quad \forall P_i \notin K_j,\ s = 1,2\\
& d_{Xk_s} \geq (a_{k_s} - x) \quad \forall P_i \notin K_j,\ s = 1,2\\
& d_{Xk_s} \geq (b_{k_s} - y) \quad \forall P_i \notin K_j,\ s = 1,2\\
& d_i \leq d_{Xk_s} + c_{k_s i} \quad s = 1,2, \quad \forall P_i \notin K_j, \quad \forall R_j,\\
& d_i \geq d_{Xk_s} + c_{k_s i} - L * l_{ji} \quad s = 1,2, \quad \forall P_i \notin K_j, \quad \forall R_j\\
& d_i \geq d_{Xk_s} + c_{k_s i} - L * (1 - l_{ji}) \quad s = 1,2, \quad \forall P_i \notin K_j, \quad \forall R_j\\
& g_{jh}(X) \leq 0, \quad h = 1, \ldots, m\\
& l_{ji} \in \{0, 1\} \quad j = 1, \ldots, J,\ i = 1, \ldots, n
\end{aligned}
$$

$f_{SRFRB}(R_j)$ is a convex MINLP defined on convex region R_j whose boundaries are linear. It has a linear objective function but includes both linear and non-linear constraints. Its binary relaxation is differentiable; therefore, we apply standard procedures for solving differentiable MINLPs. Subsequently, we obtain the global optimum of FRB by solving a finite series of smooth linear relaxations of sub-problems SRFRB(R_j).

3.2.3. Finding the global optimum

To solve SRFRB(R_j) for every candidate domain R_j, and deduce the global optimum of FRB, we determine (i) the candidate set K_j of every R_j, (ii) its reduced candidate set RK_{ji}, for all $P_i \notin K_j$, and (iii) the linear expressions of its boundaries. Even though this is theoretically possible, it is practically difficult as the number of regions increases with the increase of the number of existing facilities and of vertices of R_F. To overcome this difficulty, we apply BA, which employs a set of developed bounds on f^*_{RFRB}. We stop BA once the search obtains a good solution; thus, we avoid the computational complexity of an exact search.

4. Proposed approach

In Sections 4.1 and 4.2, we propose, respectively, lower and upper bounds. In Section 4.3, we describe the bounding based algorithm and explain how it explores the aforementioned bounds.

4.1. Lower bounds

FRB has two restrictions: (i) The new location can't lie in $\mathrm{Int}(R_F)$, and (ii) traveling through $\mathrm{Int}(R_F)$ is prohibited. Relaxing either restriction and solving the relaxed problem exactly, we obtain a lower bound for FRB. Herein, we consider two relaxations: WP, which drops both restrictions; and FR, which drops the second restriction.

Because WP is non-differentiable, we use its equivalent reformulation RWP, given by equations (3.3) and (3.4). RWP has no visibility issue. In addition, RWP is convex and differentiable. Let X^*_{RWP} be an optimum of RWP and $f_{RWP}\left(X^*_{RWP}\right)$ be its objective value. Then $f_{RWP}\left(X^*_{RWP}\right)$ is a valid lower bound for FRB [18]; that is, $f_{RWP}\left(X^*_{RWP}\right) \leq f^*_{FRB}$, where f^*_{FRB} is FRB's optimal objective value. Evidently, when feasible to FRB, X^*_{RWP} is FRB's global optimum; *i.e.*, $f_{RWP}\left(X^*_{RWP}\right) = f^*_{FRB}$.

Similarly, because FR is non-differentiable, we use its equivalent reformulation RFR, given by equations (3.3)–(3.4) and $X \notin \mathrm{Int}(R_F)$. Let X^*_{RFR} be an optimum of RFR and $f_{RFR}(X^*_{RFR})$ its value. Then $f_{RFR}(X^*_{RFR})$ is a valid lower bound for FR [18]: $f_{RFR}(X^*_{RFR}) \leq f^*_{RFR}$. Evidently, when feasible to FRB, X^*_{RFR} is FRB's global optimum; *i.e.*, $f_{RFR}(X^*_{RFR}) = f^*_{FRB}$. In general, the optimal solution X^*_{RFR} of RFR yields a tighter lower bound to RFRB than $X^*_{RWP}:\ f_{RWP}\left(X^*_{RWP}\right) \leq f_{RFR}(X^*_{RFR}) \leq f^*_{FRB}$. Let the lower bound $\mathrm{LB} = f_{RFR}(X_{FR})$.

4.2. Upper Bounds

When not in $\mathrm{Int}(R_F)$, X^*_{RWP} serves as the initial incumbent for FRB. Its associated FRB's cost $f_{RFRB}\left(X^*_{RWP}\right)$ is a valid upper bound to FRB: $f^*_{FRB} \leq f_{RFRB}\left(X^*_{RWP}\right)$. Of course, evaluating $f_{RFRB}(X)$ of any point X requires defining the visibility properties, the visibility graph and applying Djikstra's algorithm. X^*_{RFR} is always outside $\mathrm{Int}(R_F)$. Thus, X^*_{RFR} serves an initial incumbent to FRB. Its associated FRB's cost $f_{RFRB}(X^*_{RFR})$ is a valid upper bound to FRB: $f^*_{FRB} \leq f_{RFRB}(X^*_{RFR})$.

4.2.1. An NP-based upper bound

In Algorithm 5, we detail NP's steps. Ideally, the initial partitioning should mimic Butt's [5] division of the convex hull of FRB's feasible region. However, the boundaries of those sub-regions are not well behaved. In addition, their number becomes large as n increases. Instead, Lines 3–9 of Algorithm 5, we use an adapted initial partitioning. When R_F is non-convex and $\mathrm{MC} = \mathrm{MC}_{\mathrm{NC2}}$, we draw (i) a vertical line passing through X^*_{RWP} such that it bisects the borders of the convex hull, and (ii) $M-1$ arbitrary vertical lines dividing domain MC; thus, we obtain $M+1$ regions. Otherwise, we select an edge of MC and divide it with arbitrary vertical lines. For the random sampling (*cf.* Lines 10–13), we choose an arbitrary point $r_j \in R_j,\ j = 1, \ldots, M+1$. For every r_j, we evaluate RFRB's objective function value. (For the initial iteration, we use X^*_{RWP} as the sampled point of the first region.) In Line 15 of Algorithm 5, we select the most promising region; *i.e.*, the region whose sampled point has the least FRB's cost. In Line 16, we update UB whenever we encounter an improving solution. In Lines 17–19, we stop NP when the stopping criterion holds; specifically, when $\mathrm{UB} - \mathrm{LB} < 0.10$ or three consecutive iterations do not improve UB. In Lines 20–23, we either further partition the current region or backtrack to the aggregated region. Regardless, we divide the selected region into M sub-regions and aggregate the others into region $M+1$. In Line 24, we iterate the above steps.

Algorithm 5. Nested partitioning.

1: Input

- $(a_i, b_i),\ i = 1, \ldots, n$: position of existing facility P_i;
- $w_i,\ i = 1, \ldots, n$: weight of existing facility P_i;
- $V = \{v_1, \ldots, v_F\}$: vertices of forbidden region R_F;
- LB: current lower bound;
- UB: current upper bound;
- X_{inc}: incumbent solution;
- M: number of partitions of NP;
- Q: number of iterations of NP;

2: Set Counter = 0;
3: **if** R_F is non-convex, MC = MC_{NC2}, and X_{inc} lies in Conv(V) **then**
4: Partition MC into $M+1$ regions using M vertical lines;
5: Bisect the lines with the borders of Conv$(P \cup V)$;
6: **else**
7: Partition MC into $M+1$ regions by M arbitrary vertical lines with one line passing through X_{inc};
8: Bisect the lines with the borders of Conv$(P \cup V)$;
9: **end if**
10: **for** $R_j,\ j = 1, \ldots, M+1$ **do**
11: take a random point rj;
12: determine its visibility properties, its K_j, and its FRB's cost;
13: **end for**
14: choose the region R_p whose random point r_p has the minimum cost;
15: set the current region to R_p.
16: update UB and X_{inc} if the minimum cost is less than UB;
17: **if** the stopping criterion is satisfied **then**
18: stop
19: **end if**
20: **if** r_p belongs to any of the first M regions **then**
21: further partition this region and aggregate the other M regions;
22: **else**
23: partition the aggregated region and aggregate the M regions;
24: set Counter = Counter + 1 and go to 10.
25: **end if**
26: **return** Upper bound UB for RFRB and incumbent solution X_{inc};

4.2.2. Divide and conquer

In Algorithm 6, we detail the two steps of DC: division and conquering. In the division step, we partition (as in Algorithm 2) the master candidate domain MC into J finite convex candidate domains $R_j,\ j = 1, \ldots, J$, whose boundaries are linear and whose points share identical visibility properties K_j and identical distance functions. For the conquering step, we select the set SR of most promising domains. SR consists of the domain that includes X_{inc} and its neighboring domains. For every domain $R_j \in \text{SR}$, we identify the visibility properties K_j, the reduced candidate sets RK_{ji} with respect to every invisible P_i, and all the linear constraints defining the boundaries. We then solve SRFRB(R_j) *via* an MINLP solver. When we encounter an improving solution, we update UB, X_{inc}, and SR. An improved UB suggests that X_{inc} might belong to a new domain. The new SR includes all the unvisited domains that are adjacent to the region of the new X_{inc}. Before conquering a domain of SR, we check the satisfaction of its stopping criterion; we stop when $\text{UB} - \text{LB} < 0.1$. DC is therefore a heuristic when we stop before conquering all regions of MC, and is an exact method otherwise.

4.3. BA

In Algorithm 7, we explain how BA combines the obtained lower and upper bounds. First, we solve RWP.

- When $X^*_{RWP} \in \text{Int}(R_F)$, we construct MC depending on the convexity of R_F.
  - When R_F is convex, we set MC=MC_{C1} to R_F's edge nearest to X^*_{RWP}, and force FRB's solution to lie on this edge.

Algorithm 6. Divide and conquer.

```
Input
    – (a_i, b_i), i = 1,...,n: position of existing facility P_i;
    – w_i, i = 1,...,n: weight of existing facility P_i;
    – V = {v_1,...,v_F}: vertices of forbidden region R_F;
    – LB: current lower bound;
    – UB: current upper bound;
    – X_inc: incumbent solution;
    – MC: master candidate domain;
    – Q: number of iterations of NP;
                                                            # Division steps
Partition MC into candidate regions R_j, j = 1,...,J (as in Algorithm 2);
Determine the region R_1 of MC that includes X_inc;
Determine SR, the set of regions that are adjacent to R_1 including R_1.
                                                            # Conquering steps
for R_j ∈ SR do
  if R_j has been investigated then
    go to 15;
  else
    determine its K_j, and RK_ji for any invisible P_i;
    determine the linear constraints representing the boundaries of R_j;
    define and solve SRFRB(R_j) using an MINLP solver;
    if f*_SRFRB < UB then
      set X_inc = X*_SRFRB, UB = f*_SRFRB, and redefine SR;
    end if
    if stopping criterion is satisfied then
      stop;
    end if
  end if
end for
return Upper bound UB for RFRB and incumbent solution X_inc;
```

- When R_F is non-convex, we set MC=MC_{NC1} to the edges that can be joined to X^*_{RWP} with line segments that lie entirely in R_F. We solve as many relaxed FR problems as there are edges. Each relaxed FR is augmented with linear constraints forcing the solution to be on a specific edge. We choose the least cost solution as FR's global minimum X^*_{FR}. Because $X^*_{FR} \notin \text{Int}(R_F)$, we set $X_{\text{inc}} = X^*_{FR}$, $\text{UB} = f^*_{FRB}(X^*_{FR})$, and $\text{LB} = f^*_{RWP}(X^*_{FR})$. To improve UB, we perform a search on X_{inc}'s edge. We update X_{inc} whenever we improve UB. We stop when the stopping criterion is satisfied; specifically, when $\text{UB} - \text{LB} < 0.1$.

– When $X^*_{RWP} \notin \text{Int}(R_F)$, we proceed depending on the visibility of existing facilities to X^*_{RWP}.

- When visible to all existing facilities, X^*_{RWP} is feasible to RFRB, and $f^*_{RFRB} = f^*_{RWP}$. Hence, we stop and return X^*_{RWP} as FRB's global minimum and f^*_{RWP} as its minimal cost.
- When X^*_{RWP} is invisible to at least one existing facility, we create MC. If R_F is convex, we set MC=MC_{C}. When R_F is non-convex and no $P_i \in \text{Conv}(R_F)$, we set MC=MC_{NC2}; otherwise we set MC=MC_{NC3}. We set $\text{LB} = f^*_{RWP}$, $\text{UB} = f^*_{RFRB}(X^*_{RWP})$, and $X_{\text{inc}} = X^*_{RWP}$.

We employ NP with a maximum number of iterations $Q = 5$, and $M = 2$ partitions. When NP's incumbent is unsatisfactory, we apply DC. When the stopping criterion is satisfied, we stop at a near optimum: $X^* = X^*_{RWP}$.

5. Computational results

In this section, we assess BA's performance. Because the literature lacks instances of single facility problems with one polygonal forbidden region under Euclidean distances with best known solutions, we use seven FRB problems. The first four are new. Instances 1–3 have identical parameters but different R_F locations. Instance 4 has 3 existing facilities and a squared R_F. Instances 5 and 6 are those of Satyanarayana *et al.* [23]. Their solution $X^*_{\text{literature}}$ is very close the border of R_F. Instance 7 is due to Aneja and Parlar [2]. We detail all the instances

Algorithm 7. Bounding based algorithm.

Input

- $(a_i, b_i),\ i = 1, \ldots, n$: position of existing facility P_i;
- $w_i,\ i = 1, \ldots, n$: weight of existing facility P_i;
- $V = \{v_1, \ldots, v_F\}$: vertices of forbidden region R_F;

Solve RWP and set LB $= f^*_{RWP}\left(X^*_{RWP}\right)$;
if $X^*_{RWP} \notin \text{Int}(R_F)$ and X^*_{RWP} is visible to all existing facilities **then**
    Set UB $= f^*_{RWP},\ X* = X^*_{RWP},\ z* = f^*_{RWP}$, and stop.
else
    Construct RFRB's visibility graph, and set $X* = X^*_{RWP}$ and UB $= f^*_{RWP}\left(X^*_{RWP}\right)$;
end if
if $X^*_{RWP} \in \text{Int}(R_F)$ **then**
    if R_F is convex **then**
        construct MC_{C1}, the set of R_F's boundaries that are nearest to X^*_{RWP}.
    else
        Construct MC_{NC1}, the set R_F's boundaries whose points can be joined to X^*_{RWP} with line segments that lie entirely in $\text{Int}(R_F)$;
    end if
    For every boundary in MC, augment F_R with a linear constraint representing the boundary and solve the augmented problem.
    Set X_{inc} to the minimal F_R's cost solution X^*_{FR} among all candidate solutions and set UB $= f^*_{RFRB}\left(X^*_{FR}\right)$;
    if the stopping criterion is satisfied **then**
        set $X* = X^*_{FR}$, $z* =$ UB, and stop;
    else
        on the edge containing X^*_{FR}, perform a line search, update X_{inc} and UB; stop;
    end if
end if
if $X^*_{RWP} \notin \text{Int}(R_F)$ and X^*_{RWP} is invisible to at least one existing facility **then**
    if the stopping criterion is satisfied **then**
        $X^* = X_{\text{inc}}$ and $z^* =$ UB; stop;
    else
        if R_F is convex **then**
            set MC $= \text{MC}_{\text{C2}} = \text{Conv}(P \cup V) \setminus \text{Int}(R_F)$;
        else
            if no existing facility lies in the $\text{Conv}(V)$ **then**
                set MC $= \text{MC}_{\text{NC2}} = \text{Conv}(P \cup V) \setminus \text{Int}(\text{Conv}(V)$;
            else
                set MC $= \text{MC}_{\text{NC3}} = \text{Conv}(P \cup V) \setminus \text{Int}(R_F)$;
                apply NP;
                apply DC;
            end if
        end if
    end if
end if
set $X^* = X_{\text{inc}}$ and $z^* =$ UB, and stop;
return Best solution $X^* = (x^*, y^*)$ of the new facility, and z^* its objective function value;

in Table 2. In Column 1, we indicate the instance "Ins". In Columns 2 and 3, we specify the number of existing facilities and their coordinates. In Columns 4–6, we report the number and coordinates of the vertices of the forbidden region R_F. Finally, in Column 6, we indicate whether R_F is convex. In Figure 4, we present the instances along with their relaxed optima X^*_{RWP}.

We summarize the results in Table 3. In Column 1, we indicate the instance. In Columns 2 and 3, we report the ifeasibility of X^*_{RWP} to the problem. In Columns 4 and 5, we display f^*_{RWP} and f^*_{FR}. The lower bound LB equals f^*_{RWP} when $X^*_{RWP} \notin \text{Int}(R_F)$, and equals f^*_{FR} otherwise. For the latter case (*i.e.*, when $X^*_{RWP} \in \text{Int}(R_F)$), the proposed approach constructs MC from the boundaries of R_F, and solves as many problems as MC has edges. It retains the least cost solution displayed in Column 6. It then undertakes a line search along the most promising edge in MC and reports the best solution in Column 7. In the former case (*i.e.*, when $X^*_{RWP} \notin \text{Int}(R_F)$), it assesses the true cost of X^*_{RWP} given by $f_{RFRB}\left(X^*_{RWP}\right)$ as displayed in Column 8. It next applies NP and DC

TABLE 2. Characteristics of the seven tested instances.

Ins	n	P	F	R_F	R_F convex?
1	4	(0, 0), (1, 4), (5, 5), (4, 1)	3	(2, 2), (2.5, 3), (3, 2)	Yes
2	4	(0, 0), (1, 4), (5, 5), (4, 1)	3	(1, 2), (1.5, 3), (2, 2)	Yes
3	4	(0, 0), (1, 4), (5, 5), (4, 1)	3	(.5, 1), (1, 2), (1.5, 1)	Yes
4	3	(0, 0), (1, 4), (5, 5), (4, 1)	4	(2.5, 3), (2.5, 4), (3.5, 4), (3.5, 3)	Yes
5	9	(1, 8), (7, 16), (18, 18), (23, 15), (22, 11), (19, 6), (18, 2), (14, 3), (7, 6)	5	(3, 10), (4, 13), (7, 14), (9, 12), (9, 9), (5, 7)	Yes
6	8	(3, 9), (6, 12), (8, 15), (17, 16), (16, 13), (21, 13) (19, 7), (9, 5)	6	(8, 8), (8, 11), (11, 14), (13, 12), (17, 10), (13, 7)	No
7	4	(0, −10), (0, 11.6), (11, 11.6), (11, −10)	21	(1, 3), (2, 4), (2, 5), (4, 4), (3, 8), (4, 7), (5.5, 9), (7, 7), (8, 8), (7, 4), (9, 5), (9, 4), (10, 3), (9, 2), (9, 1) (6, 2), (7, 0), (4, 0), (5, 2), (2, 1), (2, 2)	No

whose best solution values $f^*_{RFRB-NP}$ and $f^*_{RFRB-DC}$ are given in Columns 9 and 10, respectively. Finally, in Column 11, we report the optimality gap UB − LB. The superscript † indicates that UB = z^*, the proven optimal solution value.

For Instance 1, X^*_{RWP} is infeasible to FRB; *i.e.*, $X^*_{RWP} = (2.5, 2.5) \in \mathrm{Int}(R_F)$ with $f^*_{RWP} = 11.314$. Because all weights are equal, any of R_F's edges might include X^*_{FRB}. Therefore, we set MC = $\mathrm{MC_{C1}}$ to include the three boundaries of R_F. To obtain X^*_{FR} for a given boundary, we augment RWP by the linear constraint representing the boundary. Among the three solutions, we retain the least cost one $X^*_{FR} = (2.313, 2.626)$, and set LB = $f^*_{FR} = 11.328$ and UB = $f_{RFRB}(X^*_{FR}) = 11.954$. Finally, we undertake a line search, on the edge containing X^*_{FR}, and update UB to 11.409 for $X_{\mathrm{inc}} = (2.5, 3)$; thus, we reduce UB − LB to 0.081. We can prove that $X_{\mathrm{inc}} = (2.5, 3)$ is the global optimum by applying DC as an exact method. For this, we partition MC = $\mathrm{MC_{C2}}$ into 19 convex candidate domains, as in Figure 5, solve SRFRB(R_j), $j = 1, \ldots, 19$, and retain the least cost 11.409.

For Instance 2, $X^*_{RWP} = (2.5, 2.5)$ is outside R_F and is visible to all existing facilities, as we show in Figure 6. Thus, LB = $f^*_{RWP} = 11.314$, and UB = $f^*_{FRB} = 11.314$. In fact, X^*_{RWP} is the global optimum.

For Instance 3, $X^*_{RWP} = (2.5, 2.5)$, which is outside R_F but is invisible to existing facility (0, 0), as we show in Figure 7. Thus, LB = $f^*_{RWP} = 11.314$ and UB = $f_{FRB}(X^*_{RWP}) = 11.382$. Neither NP nor DC further improve this solution, which can be proven to be the global optimum.

For Instance 4, $X^*_{RWP} = (3.150, 1.987)$, which lies outside R_F but is invisible to facility (3, 6) as we show in Figure 8. Thus, LB = $f^*_{RWP} = 9.837$ and UB = $f_{RFRB}(X^*_{RWP}) = 9.953$. Applying NP doesn't improve UB. Applying DC, we partition MC = $\mathrm{MC_{C2}} = \mathrm{Conv}(P \cup V) \setminus \mathrm{Int}(R_F)$ into 19 candidate domains R_j, $j = 1, \ldots, 19$, but solve SRFRB(R_j) only on the most promising regions. We thus obtain $X^* = (3.479, 1.919) \in R_1$ with $f^*_{RFRB-DC} = 9.892$ as its best solution. We can prove that $f^*_{RFRB-DC} = 9.892$ is the global optimum by solving SRFRB(R_j), $j = 1, \ldots, 19$. For this instance, LB is quite tight.

For Instance 5, $X^*_{RWP} = (15.75, 8.245) \notin R_F$, but is invisible to facility (1,8). Thus, LB = $f^*_{RWP} = 78.345$ and UB = $f_{RFRB}(X^*_{RWP}) = 78.538$. For BA, we set MC = $\mathrm{MC_{C2}} = \mathrm{Conv}(P \cup V) \setminus \mathrm{Int}(R_F)$, and apply NP obtaining the near optimum (15.75, 8.245) whose cost is 78.538. Were we to apply DC, we would have partitioned MC into 40 convex regions. Herein, we only investigate the set SR of most promising regions, displayed in Figure 9. SR = $\{R_1, R_2, R_3, R_4, R_5\}$, where R_1 is the incumbent's region and R_2, R_3, R_4, and R_5 share borders with R_1. We solve SRFRB(R_j), $j = 1, \ldots, 5$, and obtain the near-global optimum $X_{\mathrm{inc}} = (15.758, 8.09)$, located in R_1 and whose cost is 78.530.

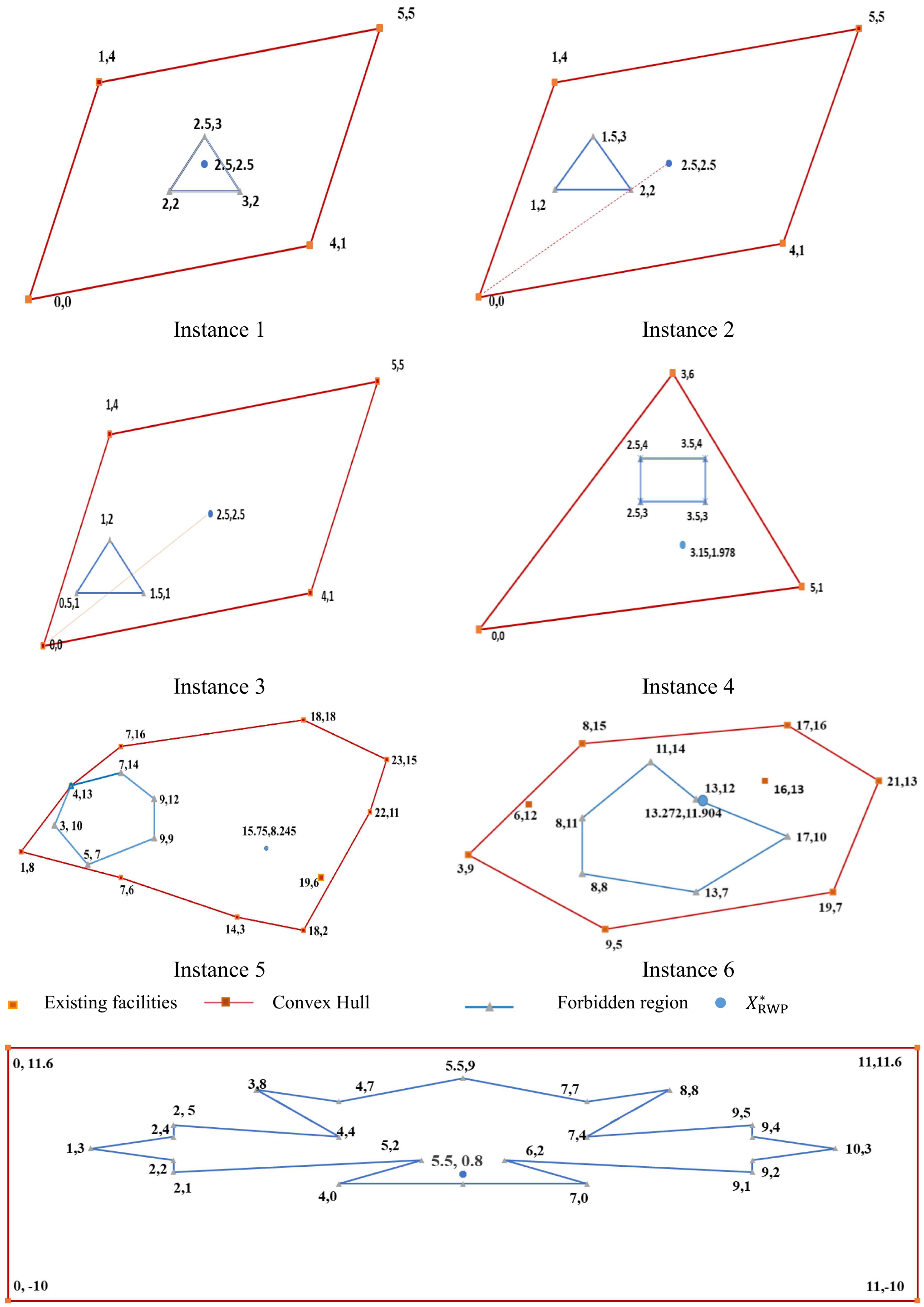

FIGURE 4. Relaxed solutions of the seven test instances.

TABLE 3. Results.

Ins	X^*_{RWP}		LB		UB					UB − LB
	$\in$ Int (R_F)?	Visible to $\forall P_i$?	f^*_{RWP}	f^*_{FR}	$f_{RFRB}\,(X^*_{FR})$	Line search	$f_{RFRB}\,(X^*_{RWP})$	$f^*_{RFRB-NP}$	$f^*_{RFRB-DC}$	
1	Yes	NA	11.314	11.328	11.954	†11.409	NA	NA	NA	0.081
2	No	Yes	11.314	NA	NA	NA	† 11.314	NA	NA	0.000
3	No	No	11.314	NA	NA	NA	†11.382	11.382	11.382	0.068
4	No	No	9.837	NA	NA	NA	9.953	9.953	†9.892	0.061
5	No	No	78.345	NA	NA	NA	78.538	78.538	78.530	0.185
6	No	No	56.004	NA	NA	NA	64.988	60.959	60.676	4.672
7	Yes	NA	48.479	48.501	52.082	52.082	NA	NA	NA	3.581

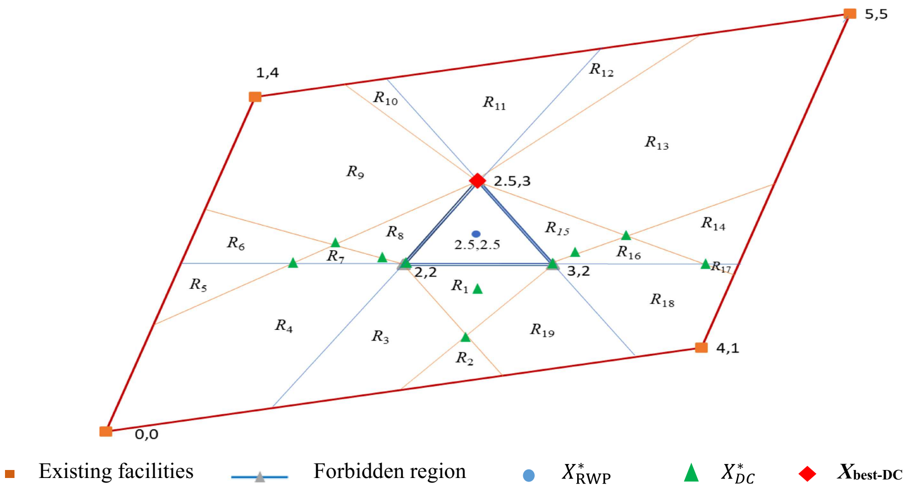

FIGURE 5. Dispersion of DC's local optima for Instance 1.

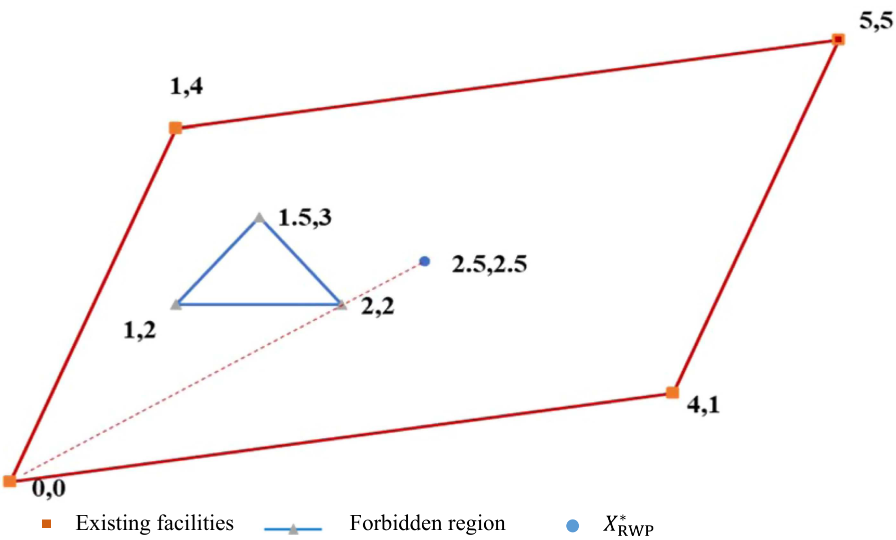

FIGURE 6. Optimal solution of Instance 2.

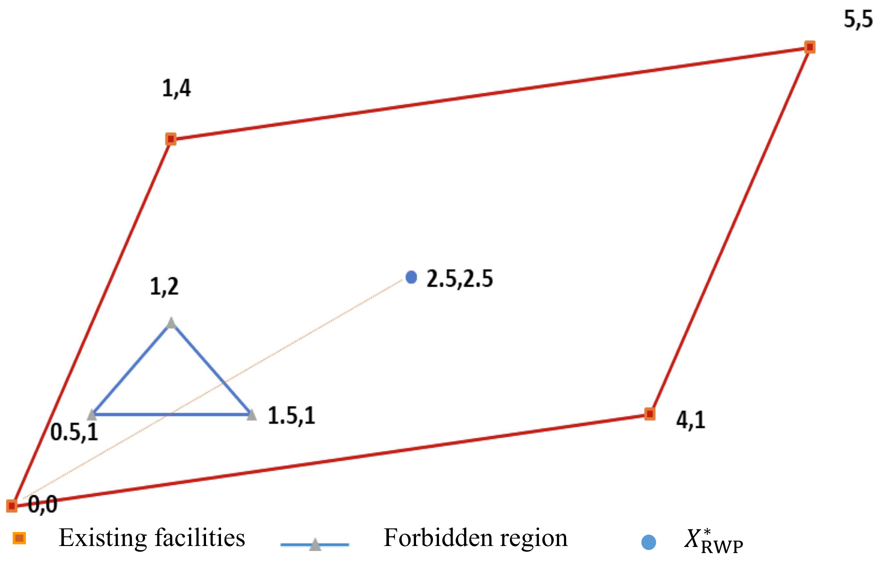

FIGURE 7. Relaxed solution for Instance 3.

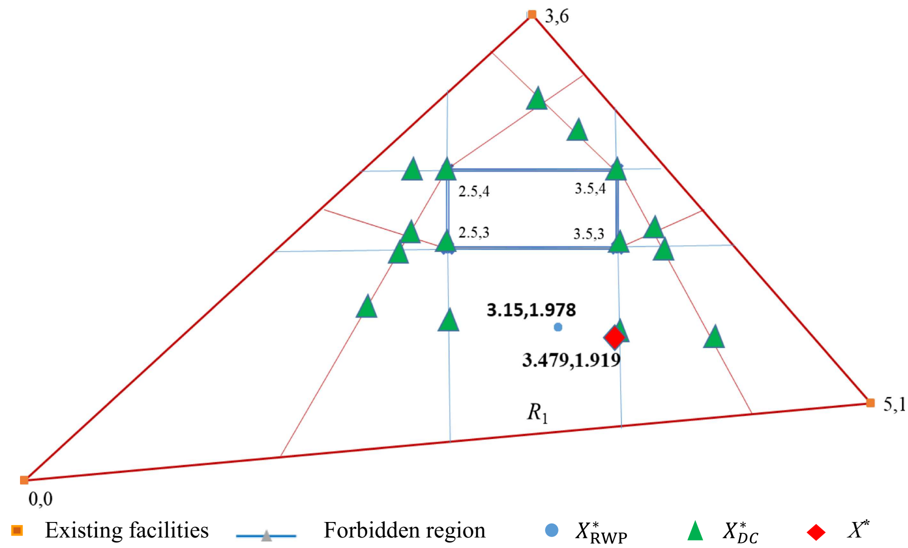

FIGURE 8. Solutions of SRFRB(R_j), $j = 1, \dots, 19$, for Instance 4.

The reported solution $X_{\text{literature}} = (15.759, 8.236)$ whose reported cost is $f_{\text{literature}} = 78.345$. $f_{\text{literature}}$ is most likely erroneous. It equals our lower bound LB, which we computed based on X^*_{RWP}. Our computation shows that $f_{FRB}\left(X^*_{\text{literature}}\right) = 78.537$, which exceeds the reported value by 0.192. This implies that BA improves the best known solution of this instance by 0.0073.

For Instance 6, which has a non-convex polygonal R_F, $X^*_{RWP} = (13.272, 11.904)$. Even though outside R_F, X^*_{RWP} is invisible to five existing facilities. Therefore, LB $= f^*_{RWP} = 56.004$, UB $= f_{RFRB}\left(X^*_{RWP}\right) = 64.988$, and MC $=$ MC$_{\text{NC2}}$ $= \text{Conv}(P \cup V) \setminus \text{Int}(\text{Conv}(V))$, as we illustrate in Figure 10. Using NP, we reduce UB

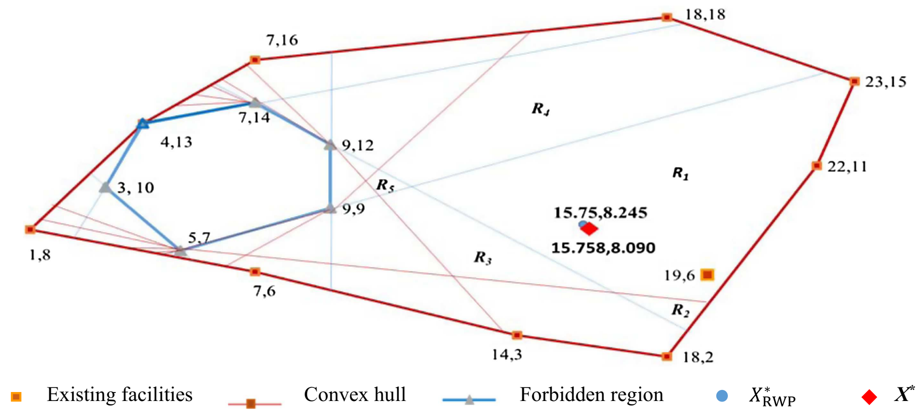

FIGURE 9. Solutions of SRFRB(R_j), $j = 1, \ldots, 19$, for Instance 5.

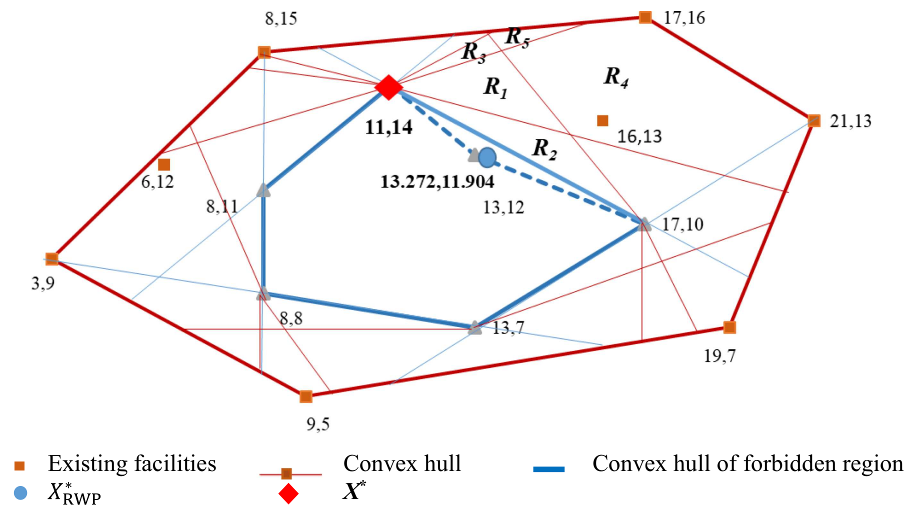

FIGURE 10. MC and SR for Instance 6.

to 60.959; thus, we improve the best known bound $f_{FRB}\left(X^*_{\text{literature}}\right) = 61.434$ corresponding to $X^*_{\text{literature}} = (14.777, 12.898)$.

Applying DC, we partition MC into 38 candidate domains but solve SRFRB(R_j), $j = 1, \ldots, 5$, only on the five most promising set of regions. We thus obtain the near-global minimum $X_{\text{inc}} = (11, 14)$ and UB $= 60.674$. This upper bound is obviously tighter than $f_{FRB}\left(X^*_{\text{literature}}\right)$. $X_{\text{inc}} = (11, 14)$ lies on the boundary of Conv(V). We suspect X_{inc} to be the global optimum.

For Instance 7, $X^*_{RWP} = (5.5, 0.8)$ with $f^*_{RWP} = 48.479$. As we show in Figure 11, $X^*_{RWP} \in \text{Int}(R_F)$; thus, is infeasible to FRB. Therefore, FRB's optimum must lie on one of R_F's boundaries. To obtain X^*_{FR}, we construct MC = MC$_{\text{NC1}}$, which consists of the nine boundaries of R_F. We show these boundaries in bold in Figure 11. For every boundary, we solve RWP augmented by the linear constraint representing the boundary. Among all solutions, we retain the least cost one: $X^*_{FR} = (5.5, 0)$, located on the boundary connecting (4, 0) and (7, 0), with $f^*_{FR} = 48.501$. Thus, LB $= 48.501$ and UB $= f_{RFRB}\left(X^*_{FR}\right) = 52.082$. We do not further improve UB when we apply a line search on the segment connecting (4, 0) and (7, 0). In fact, UB matches the best known bound from the literature.

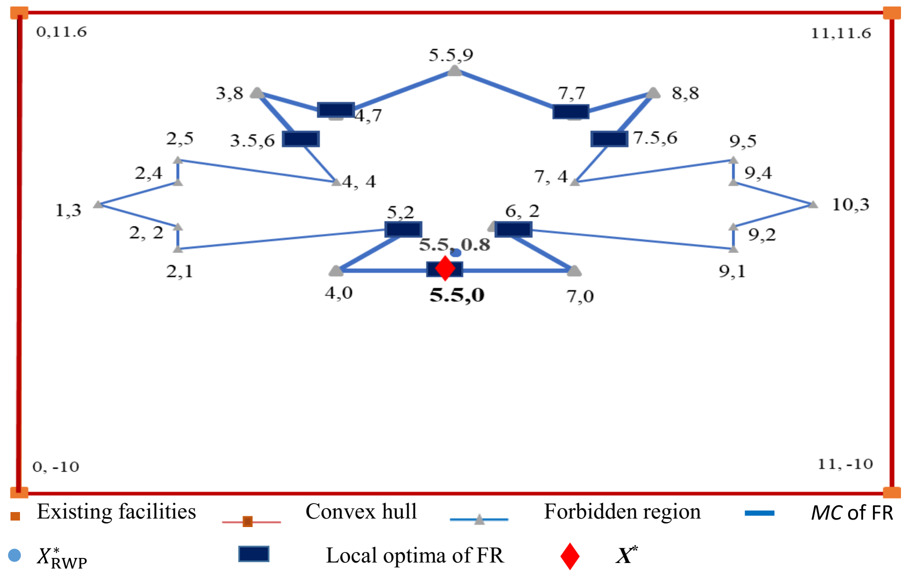

FIGURE 11. Solutions of Instance 7.

In summary, we infer the following.

– The tightness of the lower bound depends on the instance and on the shape of the forbidden region.
– The solution of the relaxed problem may be optimal.
– NP and DC improve the solution of the relaxed problem in many instances and improve best known solutions.
– Applying a stopping criterion based on the absolute gap allows the early stopping of the proposed approach. For example, for Instance 1, the approach will omit the line search; similarly, for Instance 3, the approach will omit executing NP and DC.

The runtime of FRB for the above instances is relatively small for all practical reasons. It is of the order of few minutes for all tested instances. In general, the runtime of FRB should not be an issue. FRB is usually solved only at the design stage of a system. It is neither solved repetitively nor on-line. It is considered at the strategic management level. Therefore, the solution time is not critical. The improvement in solution cost justifies a high CPU time. Regardless, MINLP instances are solved using the mixed integer non-linear programming solver CONOPT3, which uses CPLEX12.8 as the integer linear programming solver and DICOPT as the non-linear programming solver. All these solvers are evoked from GAMS25.0, which is in turn evoked from FORTRAN. The runtime of any of the MINLP problems did not exceed few seconds. Therefore, the runtime of any instance is determined by the algorithmic set up; specifically, by the time needed to generate the visibility graph, which is known to be at worst $O\left((F+n)^3\right)$.

6. Conclusion

In this paper, we focused on solving the single facility Euclidean location problem in the presence of a (non-) convex polygonal forbidden region where location and travel are not permitted. This optimization problem is non-differentiable and non-convex. We reduced the problem's feasible domain by introducing a master candidate domain that excludes dominated regions. To overcome the non-convexity and non-differentiability of the problem's objective function, we proposed an equivalent tight reformulation. The reformulation has a linear objective function, linear and convex quadratic constraints as well as non-convex, non-differentiable constraint sets. To handle the non-differentiability and non-convexity of the reformulation, we decomposed it into convex sub-problems whose objective functions are linear and which are defined on convex sub-regions, obtained by dividing the master candidate domain using a visibility grid lines partitioning. The binary relaxation of each of these sub-problems is differentiable. Subsequently, we reduced the original problem to solving a finite series of convex and differentiable sub-problems, that can be solved by mixed integer non-linear programming optimization solvers.

In addition, we developed two lower and four upper bounds. The first lower bound is the solution value of a relaxed version of the Weber problem where both location and travelling are permitted. The second, which is tighter, is the solution value of the relaxed problem where location in the forbidden region is prohibited but travel is permitted. The four upper bounds are FRB's cost of RWP, FR, nested partitioning and divide and conquer near-optima. The divide and conquer heuristic solves a series of RFRB sub-problems on the most promising domains, which are convex and which have linear boundaries. We incorporated these bounds within a bounding-based algorithm to approximately solve FRB. We presented our computational results and highlighted the advantages of the proposed approach. In particular, the proposed methodology improves two best known solutions and matches one.

We can extend the proposed method to larger sized Euclidean location problems with multiple new/existing facilities, and one or more polygonal forbidden regions. We can modify the nested partitioning making it a stand alone heuristic for larger instances. Finally, we can extend the proposed method to instances with different distance metrics, stochastic weights, three-dimensional space, etc.

References

[1] I. Al-Mudahka, M. Hifi and R. M'Hallah, Packing circles in the smallest circle: an adaptive hybrid algorithm. *J. Oper. Res. Soc.* **62** (2011) 1917–1930.

[2] Y.P. Aneja and M. Parlar, Algorithms for Weber facility location in the presence of forbidden regions and/or barriers to travel. *Transp. Sci.* **28** (1994) 70–76.

[3] M. Bischoff and K. Klamroth, An efficient solution method for Weber problems with barriers based on genetic algorithms. *Eur. J. Oper. Res.* **177** (2007) 22–41.

[4] R. Bosch and M. Trick, Integer programming, edited by E.K. Burke and G. Kendall. In: Search Methodologies: Introductory Tutorial in Optimization and Decision Support Techniques, Chapter 3. Springer, US (2005) 69–95.

[5] S.E. Butt, *Facility location in the presence of forbidden regions.* Ph.D. thesis, Pennsylvania State University, University Park, PA (1994).

[6] S.E. Butt and T.M. Cavalier, An efficient algorithm for facility locations in the presence of forbidden regions. *Eur. J. Oper. Res.* **90** (1996) 56–70.

[7] S.E. Butt and T.M. Cavalier, Facility location in the presence of congested regions with the rectilinear distance metric. *Socio Econ. Plan. Sci.* **31** (1997) 103–113.

[8] E.W. Dijkstra, A note on two problems in connection with graphs. *Numer. Math.* **1** (1959) 269–271.

[9] Z. Drezner, Sensitivity analysis of the optimal location of a facility. *Nav. Res. Log. Q.* **32** (1985) 209–224.

[10] R.L. Francis, L.F. McGinnis and J.A. White, Facility Layout and Location: An Analytical Approach. Prentice Hall, New York (1992).

[11] A. Ghaderi, M.S. Jabalameli, F. Barzinpour and R. Rahmaniani, An efficient hybrid particle swarm optimization algorithm for solving the uncapacitated continuous location-allocation problem. *Netw. Spat. Econ.* **12** (2012) 421–439.

[12] S.K. Ghosh and D.M. Mount, An output-sensitive algorithm for computing visibility graphs. *SIAM J. Comput.* **20** (1991) 888–910.

[13] H.W. Hamacher and K. Klamroth, Planar Weber location problems with barriers and block norms. *Ann. Oper. Res.* **96** (2000) 191–208.

[14] I.N. Katz and L. Cooper, Facility location in the presence of forbidden regions, II: Euclidean distance and several forbidden circles. Report OREM 79006, Department of Operations Research and Engineering Management, Southern Methodist University, Dallas, TX (1979).

[15] I.N. Katz and L. Cooper, Facility location in the presence of forbidden regions, III: lp distance and polygonal forbidden regions. Report OREM 79006, Department of Operations Research and Engineering Management, Southern Methodist University, Dallas, TX (1979).

[16] I.N. Katz and L. Cooper, Facility location in the presence of forbidden regions, I: formulation and the case of Euclidean distance with one forbidden circle. *Eur. J. Oper. Res.* **6** (1981) 166–173.

[17] L.A. Kazakovtsev, Algorithm for constrained Weber problem with feasible region bounded by arcs. *Facta Univ. Ser.: Math. Inf.* **28** (2013) 271–284.

[18] K. Klamroth, Single-Facility Location Problems with Barriers. *Springer Series in Operations Research.* Springer, Berlin (2002).

[19] H. Kuhn, A note on Fermat's problem. *Math. Program.* **4** (1973) 98–107.

[20] R.F. Love, J.G. Morris and G.O. Wesolowsky, Facilities Location: Models and Methods. Elsevier, New York (1988).

[21] R.G. McGarvey and T.M. Cavalier, A global optimal approach to facility location in the presence of forbidden regions. *Comput. Ind. Eng.* **45** (2003) 1–15.

[22] M.A. Prakash, K.V.L. Raju and V.R. Raju, Facility location problems in the presence of mixed forbidden regions. *Int. J. Appl. Eng. Res.* **13** (2018) 91–97.

[23] B. Satyanarayana, K.V.L. Raju and K.V.V. Mohan, Facility location problems in the presence of single convex/non-convex polygonal barrier/forbidden region. *Opsearch* **41** (2004) 87–105.

[24] R. Sedgewick and J.S. Vitter, Shortest paths in Euclidean graphs. *Algorithmica* **1** (1986) 31–48.

[25] H.D. Sherali and I. Al-Loughani, Equivalent primal and dual differentiable reformulations of the Euclidean multifacility location problem. *IIE Trans.* **30** (1998) 1065–1074.

[26] H.D. Sherali and I. Al-Loughani, Solving Euclidean distance multi-facility location problems using conjugate subgradient and line-search methods. *Comput. Optim. App.* **14** (1999) 275–291.

[27] L. Shi and C.H. Chen, A new algorithm for stochastic discrete resource allocation optimization. *Disc. Event Dyn. Syst.* **10** (2000) 271–294.

[28] L. Shi and S. Men, Optimal buffer allocation in production lines. *IIE Trans.* **35** (2003) 1–10.

[29] L. Shi and S. Olafsson, Nested partitions method for global optimization. *Oper. Res.* **48** (2000) 390–407.

[30] L. Shi, S. Olafsson and Q. Chen, A new hybrid optimization algorithm. *Comput. Ind. Eng.* **36** (1999) 409–426.

[31] L. Shi, S. Olafsson and N. Sun, New parallel randomized algorithms for the traveling salesman problem. *Comput. Oper. Res.* **26** (1999) 371–394.

[32] G.A. Torres, A Weiszfeld-like algorithm for a weber location problem constrained to a closed and convex set. Preprint arXiv:1204.1087 (2012).

[33] G. Wangdahl, S. Pollock and J. Woodward, Minimum trajectory pipe routing. *J. Ship Res.* **18** (1974) 46–49.

[34] E. Weiszfeld, Sur le point par lequel la somme des distances de n points donnés est minimum. *Tohoku Math. J.* **43** (1937) 355–386.