Iranian Journal of Operations Research
Vol. 17, No. 1, 2026, pp. 102-119

Inverse Median Hyperplane Location Problem in Two-Dimensional Space

Mehdi Golpayegani[1], Jafar Fathali[2,*]

Abstract: *The line location problem, which represents a specific case within the broader class of hyperplane location problems, has attracted considerable research focus location theory. This investigation addresses locating lines from a location science perspective. Given n points situated in the plane, each assigned a positive weight that reflects its relative importance, the median line is defined as the line minimizing the total sum of these weighted distances. Our study is, to our knowledge, the first to examine the inverse median line location problem in the plane under both the Euclidean and rectilinear distance norms. Specifically, when a line L is fixed, the goal is to determine the Minimum-cost modifications to the problem parameters—either the demand point weights or their spatial coordinates—such that L becomes the globally optimal median line. We proceed by developing and analyzing mathematical models that characterize this inverse problem across the two norm settings. We demonstrate that the inverse model, when demand weights are the variables, can be precisely formulated and solved via linear programming. Conversely, for the instance involving necessary modifications to the coordinates, an effective greedy algorithm is proposed for solution approximation. The practical application and performance of this developed methodology are subsequently illustrated through a set of computational experiments.*

Keywords: *median line location, inverse location, continuous facility location, rectilinear/Euclidean norms.*

1. Introduction

Finding the optimal location of hyperplanes—with lines representing a specific instance—that best fit a defined set of existing facilities has recently attracted significant interest across location theory. These problems are relevant to a multitude of theoretical and practical fields where the ideal placement of geometric entities relative to discrete data points profoundly affects key metrics like efficiency, cost, and performance. This paper specifically focuses on developing novel methodologies for line placement, framed within the principles of location theory. In this context, the line location problem is modeled as an extension of the traditional facility location problem, where the facility to be positioned is a straight line rather than a single point. Let n points, representing the location of existing facilities, be given in R^2. Each existing facility has a positive weight that signifies its importance. This weight indicates that interaction between the line and the existing facility is desired. Consequently, the decision-maker seeks to position the line such that its proximity to the demand points is maximized. The primary objective functions in line location involve either minimizing the sum of weighted distances or minimizing the maximum weighted distance between the line and the demand points, which define the median line and the center line, respectively. Fundamentally, the

* Corresponding Author.
[1] Department of Industrial Engineering, K. N. Toosi Univercity of Technology, Tehran, Iran.
Email: mehdi.golpayegani84@gmail.com.
[2] Faculty of Mathematical Sciences, Shahrood University of Technology, Shahrood, Iran.
Email: fathali@shahroodut.ac.ir

classical line location problem aims to either enhance accessibility or reduce transportation expenses between the newly located line and the fixed demand points.

The classic line location problem carries substantial real-world applicability. Its primary function is to realize gains in accessibility or achieve reductions in logistical expenditure across diverse operational contexts. The scope of practical instances includes the optimal planning of utility networks, such as pipelines for gas or water; the structural alignment of canals for drainage and irrigation; the trajectory determination of belt conveyors in industrial environments; and the strategic siting of major transport infrastructure, like rail links or highways connecting population centers, where town demographics typically serve as the weighting factor [1]. Furthermore, these location models have been effectively leveraged in the realm of transportation planning for tasks like setting the most advantageous route for mass transit or railway lines to serve concentrated population groups [1]. Additionally, the theoretical underpinnings are mirrored in machine learning, notably in the Support Vector Machine (SVM) methodology, which operates by identifying a separating hyperplane through the minimization of specific distance functions relative to the input data set (refer to [2], [3], and [4]).

The paradigm of inverse optimization has solidified its position as an influential and growing trajectory of research within the field of mathematical programming in recent decades. In contrast to standard forward problems, inverse optimization endeavors to ascertain the minimum-effort alterations necessary for the existing problem parameters to ensure a pre-specified, candidate solution achieves optimality. Within the specific context of location theory, this often translates to modifying attributes of the demand points—such as their associated importance weights or their physical coordinates—so that a permanently situated facility (either a point or a line) satisfies the optimality criteria of the objective function. Such parameter adjustments are commonly governed by two modeling approaches: a cost-minimization framework or constraints imposed by a limited budget [5]. These approaches delineate two separate aims: achieving service provider (facility) optimization at the lowest possible expenditure (inverse optimization), or upgrading service provision using restricted fiscal resources (reverse optimization) [6].

Despite extensive examination of inverse models applied to conventional point-based facility location problems, the reciprocal scenario involving line-based models—particularly under median criteria—has received scant systematic attention. This deficiency is highly significant given the inherent practical complications and substantial economic outlay associated with altering linear infrastructure post-construction, such as rail systems, pipelines, or industrial transport conveyors. Due to the fact that relocating or physically modifying these established systems is frequently either cost-prohibitive or technically impractical, many real-world situations favor adjusting peripheral elements. This involves altering adjacent or related parameters—for instance, the service priority or the geographical arrangement of demand points—rather than moving the line itself. Illustrative cases include network planning where adjusting the service intensity or coordinates of local stations can optimize pipeline performance, or urban planning where repositioning residential centers near an existing railway might lower community access costs. Similarly, in industrial production, machinery positioning around a fixed conveyor can minimize material handling effort. These examples collectively underscore the critical practical need and inherent theoretical merit of investigating inverse line location problems.

This paper investigates the inverse median line location problem, focusing on determining the minimum-cost adjustments to the problem's fundamental parameters, such as the demand point weights and spatial coordinates, such that a designated, pre-specified line achieves optimality. To capture this problem, two established distance metrics are utilized for model formulation: the rectilinear and the Euclidean norms. Subsequently, based on the characteristics and models derived, we employ linear optimization methods to address the resulting formulations.

The principal achievements of this research are enumerated below:

1. Inverse Formulations: This study introduces the inverse mathematical formulations for the median line location problem in a two-dimensional plane. We specifically model two distinct scenarios: one involving the modification of point weights and the other involving the modification of point coordinates, both designed to ensure the candidate line is established as the median line.
2. Model Properties and Linearity: We establish several intrinsic properties of the problem under both the Euclidean and rectilinear norms. Critically, we demonstrate that when the weights of the demand vertices are the parameters subject to modification, the resulting optimization model admits a linear programming structure.
3. Coordinate Modification Algorithm: For the inverse problem requiring adjustments to the coordinates of the demand points, an efficient greedy algorithm is developed and proposed for obtaining approximate solutions.

This paper proceeds with the following organization to progressively introduce and resolve the studied problem: Section 2 first furnishes a necessary literature review, synthesizing previous research on both inverse facility location models and classical line location problem. Section 3 then establishes the foundational mathematical model for the forward line location problem. The core theoretical contribution begins in Section 4, where we present the comprehensive model for the inverse median facility location problem when demand point weights are variable. Subsequently, Section 5 shifts focus to the scenario involving variable point coordinates, detailing the proposed solution methodology and presenting corresponding numerical examples. Finally, Section 6 offers the conclusion, summarizing the key findings of this study.

2. Literature Review

The foundational work on the classic line location model was established in 1975 by Wesolowsky [7], who investigated the median case utilizing the L_2 norm and introduced the first exact algorithm for its resolution. Subsequent research by Morris and Norback [8] explored a graphical representation of the median line problem for smaller instances, offering exact solution algorithms and proving that a median line necessarily intersects at least two demand points. They later extended this median problem using the rectilinear norm and identified key problem properties [9]. They also introduced the minimax model, defining the center facility line location problem. Lee and Wu [10] further advanced this area by analyzing several location problems, including the center line problem, and improved upon Morris and Norback's algorithm. They proved specific properties for the center line location problems, in the cases that the same or different weights have been assigned to the given points, presenting algorithms with complexities of $O(n \log n)$ and $O(n^2 \log n)$, respectively. Megiddo and Tamir [11] considered the minisum criterion for n weighted demand points under both rectilinear and Euclidean norms, yielding efficient time algorithms of $O(n\log^2 n)$ and $O(n^2 \log n)$, respectively. Furthermore, Lee and Ching [12] provided an $O(n^2)$ time algorithm for the median version under the Euclidean norm. Korneenko and Martini [13] studied the Euclidean median line problem and demonstrated that the median line passes through a minimum of one demand point. Schöbel [14] provided a comprehensive study on both center and median line location problems for all norms, establishing various general properties. She later extended this work to the case that the line is forbidden from passing through certain planar regions [15], initially for the vertical distance, and then generalized this restriction for arbitrary norms. Schöbel [16] also examined the problem using block norms, demonstrating that it is solvable with an algorithm having an $O(n)$ time complexity. Extending the spatial dimension, Brimberg et al. [17] introduced the problem in three-dimensional space (R^3) using the l_p norm, alongside specific variants of Euclidean and rectilinear norms, proposing heuristic solution algorithms. A practical illustration of the line location problem arises in mining operations, where the objective is to minimize the annual transportation costs of moving minerals by digging a main shaft and accessing deposits through tunnels, offering an alternative to individual vertical boreholes to various underground sites. The location of obnoxious lines—facilities to be kept as far as possible from demand points—has also been a significant focus. Drezner and Wesolowski [18] first studied the undesirable path location problem, which is maximizing the minimum weighted distance from the path to demand points. This concept was extended to three dimensions by locating an undesirable plane that maximizes its distance from the convex hull of demand points, solvable in $O(n^3)$ time [19]. Diaz-Báñez et al. [20] studied locating an obnoxious line constrained to pass through the convex hull of given points, solving the weighted maximin criterion model using arbitrary norms.

Research also branched into the case that the line is semi-obnoxious. Golpayegani et al. [21] provided the first study on the minisum semi-obnoxious problem under L_2 norm, employing the Particle Swarm Optimization (PSO) method and demonstrating its efficiency. They later addressed the semi-obnoxious median line problem with the rectilinear norm, proving key properties and comparing the performance of PSO against a Genetic Algorithm (GA) in [22] and [23]. Furthermore, attention has been given to locating half lines and line segments. Lee and Wu [10] investigated the center half line problem, deriving properties and presenting an $O(n \log n)$ time algorithm. An improved algorithm was later introduced for the center line segment location problem, solvable in quadratic time [24]. Efrat and Sharir [25] further refined this center line segment algorithm with considering the Euclidean norm, achieving an approximate time complexity of $O(n^{1+\epsilon})$, where $\epsilon > 0$. Barcia et al. [26] examined the problem of locating obnoxious line segment with fixed-length under the minimax criterion, proposing an $O(n \log n)$ time algorithm. Recently, Golpayegani and Fathali [27] investigated the convexity and sensitivity analysis of the median line location problem.

The concept of inverse location problems has been a longstanding focus within academic literature. Berman et al. [28] conducted research into the efficacy of altering the structure of a transportation network—specifically through the reduction or addition of links—to achieve the most efficient improvement in the known placement of existing facilities. Cai et al. [29] demonstrated that the polynomial-time solvability of a classical (forward) problem does not guarantee the same complexity for its inverse counterpart. They illustrated this principle by considering the inverse center facility location problem, proving that this problem is NP-hard even though its original formulation can be solved in polynomial time. Furthermore, Zhang et al. [30] investigated the related reverse location problem utilizing a minimax objective function within a network setting, successfully proposing a polynomial-time algorithm for its resolution on tree graphs. Alizadeh et al. [31] examined the inverse of the 1-center location problem focusing on edge length modifications, successfully devising an algorithm with a time complexity of $O(n \log n)$ for its resolution. Building on this work, Alizadeh and Burkard [32] studied the inverse 1-center location problems specifically on tree networks. They proposed distinct combinatorial algorithms applicable to scenarios where the network topology is allowed to change and where topology changes are strictly prohibited. The same authors later addressed the inverse obnoxious center location problem on trees, considering modifications to the edge lengths, and introduced a linear time algorithm with a complexity of $O(n)$ for solving this specific case. Subsequently, Jana et al. [33] analyzed the inverse 1-center location problem on a tree, where the weights of edges were constrained to be bounded, providing an optimal algorithm with $O(n)$ time complexity. Furthermore, Nguyen [34] focused on a reverse 1-center problem on a tree, presenting an $O(n^2)$ algorithm for its solution. Research into both reverse and inverse location problems remains an active field of study in recent years. For instance, Golpayegani et al. [35] pioneered the inverse formulation for the semi-obnoxious case of the median line location problem, analyzing the model under both L_1 and L_2 norms. The undesirable (obnoxious) scenario of the inverse center location problem on graphs has been investigated by Alizadeh and Etemad [36]. They focus on modifying vertex weights, and successfully provided exact algorithms for both reduction and augmentation the length of edges. Keshtkar and Ghiyasvand [37] studied the inverse fastest center location problem on trees, considering alterations to edge capacities. They presented $O(n)$ time complexity algorithms for its solution. Omidi et al. [38] contributed to multi-facility location by studying the inverse and reverse vertex 2-center location problem, devising an $O(n\,log\,n)$ algorithm to tackle their model. Furthermore, Gholami and Fathali [39], [40] and [41] explored the inverse circle location problem, focusing on changing the coordinates and weights of existing facilities subject to a minimum total cost. Mohammadi et al. [42] addressed the inverse p-facility maxian location problem on a network, where demand points were fixed at nodes and the goal was to minimize the cost of changing both edge lengths and vertex weights such that a pre-specified facility placement attained optimality for the p-facility maxian problem. The inverse single facility location problem with variable edge length in the cases that the length of edges are bounded or not on a tree, have been investigated by Omidi and Fathali [43]. They proposed algorithms with complexities of $O(n^2)$ and $O(n\,log\,n)$, respectively, for these scenarios. Fathali [44] studied the generalized inverse facility location problem in the plane and introduced a row generation method as a means to solve the case involving variable weights. Most recently, Golpayegani and Fathali [45] developed some greedy algorithms for solving inverse center line location problem with variable weights and coordinates.

2.1. Research gap

The inverse case of the median line location problem has not been considered by any author. Based on this gap in this paper we study this problem. The central objective is to identify the minimum-cost adjustments to the problem's core parameters—specifically the importance weights or the spatial coordinates of the demand points—necessary to ensure a designated, pre-specified line achieves optimal facility placement. For the purpose of mathematical modeling and analysis, this inverse problem is examined under two established distance metrics: the rectilinear and the Euclidean norms.

3. Median line location problems

Let M= $\{A_1, A_2, \dots, A_n\}$ be the set of given existing facilities (demand points). Moreover, assume that $A_i = (a_{i1}.\, a_{i2})$ and w_i, for $i = 1, \dots, n$, denote coordination and weight of each demand point, respectively. A basic model of the line location problem is median line location that can be stated as follows:

$$\min F_1(L) = \sum_{i=1}^{n} w_i d(A_i, L), \qquad (1)$$

where $L = \{(x,y) | y = b + sx,\, x \in \mathbb{R}\}$ is the line facility, and $d(A_i, L) = min_{p \in L} d(A_i, p)$, for $i = 1, \dots, n$, is the distance between demand point A_i and the facility line L. The optimal solution of model (1) is called median line.

The following important property holds for the median line location problem with Euclidean and rectilinear norms.

Property 3.1 [14] The median line passes through at least two of the demand points.

Let $W = (w_1, \dots, w_n)$. We define the following objective function, which is depended to M, W and L.

$$F(L,\mathrm{M},\mathrm{W}) = \sum_{i=1}^{n} w_i d(A_i, L).$$

In the following sections, we consider the inverse median line location problem with modifying coordinates and weights of demand points using Euclidean and rectilinear norms.

4. Inverse model with changing the weights of points

Let the line $L^* = \{(x,y)|y=s^*x+b^*,\ s^*,b^* \in R\}$ be a given line in R^2. For $i = 1, \dots, n$, let c_i be the cost of changing per unit of the weight w_i. In the inverse median line location problem with modifying weights, the aim is finding the values of increasing and decreasing of w_i with minimum cost such L^* becomes the optimal solution of problem (1). To model this problem let r_i^+ and r_i^- be the variables of increasing and decreasing of w_i, respectively. Then we should determine these variables such that the value of the objective function in model (1) respect to L^* is less than what obtained with correspond to any other line L in R^2. So, the mathematical linear model of the inverse median line location problem with variable weights of points can be written as follows:

$$Min \quad \sum_{i=1}^{n} c_i(r_i^+ + r_i^-) \tag{2}$$

s.t.

$$F(L^*,M,\widetilde{W}) \le F(L,M,\widetilde{W}) \qquad \forall L \subseteq R^2 \tag{3}$$

$$\widetilde{w_i} = w_i + r_i^+ - r_i^-, \qquad i = 1, \dots, n \tag{4}$$

$$r_i^+, r_i^- \ge 0. \qquad i = 1, \dots, n \tag{5}$$

Note that, since in this case distances of the points and the lines are not changed, the presented model holds for any norm.

Using Property 3.1, the following lemmas can be considered for the inverse median model.

Lemma 4.1 In the inverse median line location problem with variable weights, if the given line L^* passes through less than two demand points, the inverse problem with Euclidean and rectilinear norms has no optimal solution.

Proof. Assume that L^* does not pass through any demand point or at most one point lie on L^*. It is clear that in this situation modifying the weights of demand points to any arbitrary amount does not change the position of the demand points with respect to the line L^*. This means that any change is made in the weights of demand points, the mentioned line will still pass through less than two points.

On the other hand, as stated in property 3.1, for the Euclidean and rectilinear norms, the median line passes through at least two demand points. Therefore, if all lines passing through two demand points are taken into account as the feasible space of the inverse median line facility models, considering any value for the weight of the demand points there exist at least one line like L' passing through two demand points which minimizes the median line objective function and $F(L', M, \widetilde{W}) < F(L^*, M, \widetilde{W})$. Therefore, considering L^* passing through less than two demand points, the inverse median models with modifying weight of demand points does not have optimal solution. □

Observation 4.2 The given specification along with property 3.1 shows that there is no need to consider whole lines in the plane when we deal with constraint (3). Particularly, the mentioned characteristics of the inverse models lead to the fact that models for the inverse problem with modifying weights of demand points reduce to a linear programming model with $O(n^2)$ constraints, in which each constraint corresponds to the line that passes through two demand points.

5. Inverse models with changing the coordinates of points

Again, suppose L^* be the given line in the plane. The goal is changing the coordination of the given points at minimum cost such that L^* becomes the optimal solution of model (1). To model this aim, for $i = 1, \dots, n$, let k_{i1} and k_{i2} be the values of increasing and decreasing of the horizontal coordinate of the given point A_i, respectively, and t_{i1} and t_{i2} be the values of increasing and decreasing of the vertical coordinate of the given point A_i, respectively. Moreover, suppose c_i is the cost of changing the coordinate of given point A_i. Then the inverse median line location problem can be modeled as follows:

$$Min \quad \text{TC} = \sum_{i=1}^{n} c_i\,(k_{i1} + k_{i2} + t_{i1} + t_{i2}) \tag{6}$$

s.t.

$$F(L^*,\widetilde{M},W) \le F(L,\widetilde{M},W) \qquad \forall L \in R^2 \tag{7}$$

$$\tilde{a}_{i1} = a_{i1} + k_{i1} - k_{i2}, \qquad i = 1,\dots,n \tag{8}$$

$$\tilde{a}_{i2} = a_{i2} + t_{i1} - t_{i2}, \qquad i = 1,\dots,n \tag{9}$$

$$k_{i1}, k_{i2}, t_{i1}, t_{i2} \ge 0, \qquad i = 1,\dots,n \tag{10}$$

Observation 5.1 The inverse version of the median line location problem with modifying coordinates is feasible for any given line in R^2.

Proof. Note that in the case that the given line does not pass through any demand point, then considering the definition of inverse model, the coordinates of the demand points can be changed with minimum cost such that the candidate line passes through at least two demand points and its objective function is the minimum with respect to the points with the new coordinates. □

If L^2 norm is used, then the shortest distance between a line $L = \{(x,y)|y = b + sx, x \in \mathbb{R}\}$ and any point $A_i \in M$, can be obtained by the following formulation:

$$d(A_i, L) = \frac{|a_{i2} - sa_{i1} - b|}{\sqrt{1+s^2}}. \tag{11}$$

In the case L_1 norm, the shortest distance between line L and the point A_i is calculated as follows:

$$d(A_i, L) = min\{d_{ver}(A_i, L), d_{hor}(A_i, L)\}, \tag{12}$$

in which

$$d_{ver}(A_i, L) = |sa_{i1} - a_{i2} + b|, \tag{13}$$

and

$$d_{hor}(A_i, L) = \frac{1}{|s|}|sa_{i1} - a_{i2} + b|. \tag{14}$$

Considering these distances and other L_p norms, it is observed that the presented model for inverse median line location problem with variable coordinates is a nonlinear programming with infinite constraints. Thus, it is hard to solve. In the following section, we propose an efficient algorithm to solve the mentioned problem.

5.1. The algorithm

Let $L^*: y = s^*x + b^*$ be the given line, then to make it an optimal solution to the median line location problem, we should change the coordinates of the given points such that

$$F(L^*,\widetilde{M},W) \le F(L,\widetilde{M},W), \qquad \text{for each line } L \text{ passing through pair of current points.} \tag{15}$$

The idea of the algorithm is to transfer the points to the line L^* one by one such that (15) is met. Thus, in each iteration, the point with maximum reduction on $F(L^*,\widetilde{M},W)$ should be considered for transferring. On the other hand, minimizing the cost of transferred points should be also considered.

Let the point $A_i = (a_{i1}, a_{i2})$ be transferred to the line $L^*: y = s^*x + b^*$. Then, $F(L^*,\widetilde{M},W)$ will be reduced by $\Delta_i = w_i d(L^*, \widetilde{A}_i)$.

In the case that distances are measured by Euclidean norm, the new coordinate of A_i can be found using the following lemma.

Lemma 5.2 In the Euclidean norm the projection of the point $A_i = (a_{i1}, a_{i2})$ on the line L: $y = sx + b$ is calculated as follows:

$$\tilde{a}_{i1} = \frac{sa_{i2} + a_{i1} - sb}{s^2 + 1}, \qquad \tilde{a}_{i2} = s\tilde{a}_{i1} + b.$$

Proof. To find the projection of A_i on L, we need to find the intersection of L and the line L' which passes through A_i perpendicular to L. The line L' is $y = -\frac{1}{s}(x - a_{i1}) + a_{i2}$. Thus, the coordinate of intersection of L and L' is obtained by the following:

$$x = \frac{sa_{i2} + a_{i1} - sb}{s^2 + 1}, \quad y = s\tilde{a}_{i1} + b. \quad \square$$

The cost of transferring A_i to L^* is

$$TC_i = c_i(|a_{i1} - \tilde{a}_{i1}| + |a_{i2} - \tilde{a}_{i2}|).$$

Therefore, to accomplish both aims of minimizing the transfer costs and maximizing the reduction of the objective function, we transfer the point with the minimum value of $r_i = \frac{TC_i}{\Delta_i}$ during each iteration.

These ideas lead us to the following algorithm.

Algorithm 1.

Input: $L^*: y = s^*x + b^*$, the given line, $A_i = (a_{i1}, a_{i2})$ the coordinates of the given points, w_i the weights of given points, and c_i the cost of modifying coordinates, for $i = 1, \dots, n$.

Output: The modified coordinates of demand points and TC, the total cost of changing coordinates of given points.

---

1. **Set** TC=0.

2. **Calculate** $F(L^*, M, W)$.

3. **For** each line L passing through pairs of points A_i, calculate $F(L, M, W)$ and find

$$f_m = \min F(L, M, W).$$

4. **If** $F(L^*, M, W) \le f_m$, **then** stop, L^* is the optimal line.

5. **For** i=1,…,n **do**

5.1. **Calculate** Δ_i, the value of reduction $F(L^*, M, W)$ by transferring A_i to L^*, as follows:

$$\Delta_i = w_i d(L^*, A_i)$$

5.2. **Find** the coordinate of projection of point A_i on L^* as follows:

$$\tilde{a}_{i1} = \frac{s^* a_{i2} + a_{i1} - b^* s^*}{(s^*)^2 + 1}, \quad \tilde{a}_{i2} = s^* \tilde{a}_{i1} + b^*$$

5.3. **Calculate** TC_i, the cost of transferring A_i to L^*, as follows:

$$\mathrm{TC}_i = c_i(|a_{i1} - \tilde{a}_{i1}| + |a_{i2} - \tilde{a}_{i2}|)$$

5.4. **End for**

6. **Set** $r_j = \frac{TC_i}{\Delta_i}$, for $j \in \{1, \dots, n\}$, such that $r_j \le r_k$ for any $k > j$. (i.e. for $i = 1, \dots, n$, sort r_i in ascending order.)

7. **Set** $j = 1$.

8. **While** $F(L^*, M, W) > f_m$ **do**

8.1. **Set** $F(L^*, M, W) = F(L^*, M, W) - \Delta_j$.

8.2. **Set** $TC = TC + TC_j$.

8.3. **Update** the modified demand point j as $A_j = (\tilde{a}_{j1}, \tilde{a}_{j2})$.

8.4. **For** each line L_{ji} passing through A_j and any other point A_i calculate

$$f_j = \min_{i=1,\dots,j,i\ne j} F(L_{ij}, M, W)$$

8.5. **Set** $f_m = \min(f_m, f_j)$

8.6. **Set** $j = j + 1$.

8.7. **End while**
End

Considering step 8 of Algorithm 1, the following property is concluded.

Lemma 5.3 Algorithm 1 transfer the points such that for any line L passing through two points respect to new coordinates, we have

$$F(L^*, \widetilde{M}, W) \leq F(L, \widetilde{M}, W).$$

To calculate the time complexity of Algorithm 1, note that the number lines passe through two existing points is $O(n^2)$. In Step 3, for each of these lines $F(L, M, W)$ should be compute which needs $O(n)$ time. Thus Step 3, needs $O(n^3)$ time. Step 5 needs $O(n)$ time and the values of r_i for $i = 1, \ldots, n$, can be computed in $O(nlogn)$ time in Step 6. Step 8 terminated at most in N iterations (i.e. when all the points lie on the line L^*). In Step 8.4 the value of $F(L, M, W)$ should be calculated for N lines. Thus, this step needs $O(n^2)$ time. Therefore, Step 8 needs $O(n^3)$ time. So, the time complexity of Algorithm 1 is $O(n^3)$.

In the case L_1 norm, if $|s^*| \leq 1$, then the distance of any point $A_i = (a_{i1}, a_{i2})$ to the line L^* is obtained by the vertical distance $d_{ver}(A_i, L^*)$ and the projection of A_i and L^* is obtained as the following.

$$\tilde{A}_i = (a_{i1}, s^* a_{i1} + b^*). \tag{16}$$

If $|s^*| > 1$, then the distance of $A_i = (a_{i1}, a_{i2})$ and line L^* is obtained by the horizontal distance $d_{hor}(A_i, L^*)$ and the projection of E_i and L^* is obtained by

$$\tilde{A}_i = \left(\frac{1}{s^*}(a_{i2} - b^*), a_{i2}\right). \tag{17}$$

Thus, for the rectilinear norm, the Step 5.2 of Algorithm 1 should be replaced by (16) or (17), corresponds to the value of s^*.

5.2. Computational results

We test our proposed algorithm for the inverse line median problem on several benchmark and random instances. The implementation in MATLAB R2014b was run on a PC with an AMD Ryzen 5 processor (2.10 GHz) and 8 GB of RAM. The test set includes three standard benchmarks from Beasley [44] (Ruspini, Bongartz, TSPLIB) and ten randomly generated instances. All weights and costs were randomly assigned, with their specific ranges detailed in Table 1.

Table 1. The range of the coordinates, weights and changing costs of the existing points.

test	n	Range of points	Interval of weights	Interval of costs
Ruspini	75	(4,4) to (117,156)	[1,10]	[1,10]
Bongarts	287	(5, 5) to (48, 48)	[1,10]	[1,10]
TSPLIB	654	(1000, 1000) to (5000, 5000)	[1,10]	[1,10]
Test 1	100	(0,0) to (10,10)	[1,10]	[1,10]
Test 2	200	(0,0) to (50,50)	[1,50]	[1,50]
Test 3	300	(0,0) to (50,50)	[1,50]	[1,50]
Test 4	400	(0,0) to (50,50)	[1,50]	[1,50]
Test 5	500	(0,0) to (50,50)	[1,50]	[1,50]
Test 6	600	(0,0) to (150,150)	[1,150]	[1,150]
Test 7	700	(0,0) to (150,150)	[1,150]	[1,150]
Test 8	800	(0,0) to (25,25)	[1,25]	[1,25]
Test 9	900	(0,0) to (100,100)	[1,100]	[1,100]
Test 10	1000	(0,0) to (100,100)	[1,100]	[1,100]

Table 2 contains the best lines correspond to the given initial coordinates. In this table, the pair (s, b) in the column with heading L_{best} indicates the slop and intercept of the line passing through two existing points in which its objective function of the line median problem is minimum.

The results of running Algorithm 1 on test problems with Euclidean and rectilinear norms are reported in tables 3 to 6. In these tables, the column with heading "iter." indicates the number of termination iteration. It also shows the number of transferring points to the line L^*. The results are reported for varying given lines.

Comparing the values of $F(L_{best},M,W)$ in Table 2 with $F(L^*,\widetilde{M},W)$ values in tables 3 to 6, indicate that the new obtained values of the objective line medians are less than the initial ones. Thus, since by Lemma 3 the obtained solution is a local minimum with respect to the new coordinates, by spending some cost to change the coordinates, the decision-maker can reduce the total transportation cost for all clients. Note that changing the coordinates may be done once in a period of time, but clients use the facility continuously during this period. Therefore, the transportation cost for clients is more important than the cost of changing the coordinates of the points.

Table 2. The best lines correspond to the initial given points for L_1 and L_2 norms.

Test	n	L_2 norm		L_1 norm	
		(s,b) $L_{best}: y = sx + b$	$F(L_{best},M,W)$	(s,b) $L_{best}: y = sx + b$	$F(L_{best},M,W)$
Ruspini	75	(0.1250,22.0000)	9884.7994	(1.0000,36.0000)	20407.0000
Bongarts	287	(-1.2727,60.5455)	7889.3003	(-1.0000,54.0000)	11201.0000
TSPLIB	654	(-0.3683,4887.9761)	3866036.9156	(-0.8857, 6965.7142)	7850577.5806
Test 1	100	(1.3333,-2.0000)	1085.2000	(1.0000,0.0000)	1564.0000
Test 2	200	(0.7907,5.4651)	59738.5010	(1.0000,0.0000)	85242.0000
Test 3	300	(-0.8947, 49.0526)	86321.9289	(1.0000,0.0000)	122233.0000
Test 4	400	(-0.9091,49.0909)	115277.0129	(-1.0000,52.0000)	163185.0000
Test 5	500	(0.9474,-0.4211)	145510.9111	(1.0000,-2.0000)	205912.0000
Test 6	600	(0.9286,8.4286)	1576548.1963	(1.0000,3.0000)	2233366.0000
Test 7	700	(0.8507,13.9403)	1828418.1876	(1.0000,3.0000)	2591683.0000
Test 8	800	(0.9583,0.0417)	57834.4740	(1.0000,0.0000)	81796.0000
Test 9	900	(1.0000,1.0000)	961904.2245	(1.0000,1.0000)	1360338.0000
Test 10	1000	(-1.0٤٥٥,10٥.٢٧٢٧)	1151912.9566	(-1.0000,103.0000)	1629717.0000

Table 3. The results of Algorithm 1 for Beasley [46] instances with Euclidean norm.

Test	n	(s^*, b^*) $L^*: y = s^*x + b^*$	$F(L^*,M,W)$	Iter	Total cost (TC)	$F(L^*,\widetilde{M},W)$ (Respect to new coordinates)	CPU time (in sec.)
Ruspini	75	(0.5,0.5)	27607.3896	39	13560.0000	9722.8707	0.0529
		(0.5,10)	25432.1427	38	11459.4000	9563.6627	0.0453
		(1,20)	15056.4246	20	2358.0000	9460.3816	0.0484
		(-1,20)	39683.5396	47	2839.0000	9051.6739	0.0500
		(10,0)	19963.7296	31	5285.3366	9373.3498	0.0421
Bongarts	287	(0.5,0.5)	29793.3697	163	16228.8000	7859.3317	1.8682
		(-5,10)	42420.1161	190	26599.1538	7719.3271	1.9746
		(-1,50)	9029.0464	45	550.0000	7111.3728	1.7826
		(10,20)	33935.6438	177	17695.6237	7726.6627	2.1264
		(1,10)	9921.4152	43	7696.8573	616.0000	1.61764
TSPLIB	654	(0.5,0.5)	7164521.6787	255	2058978.0000	3649113.1227	85.3954
		(-.5,4000)	5870835.6061	213	1274233.5000	3347819.7332	84.7211
		(10, 5000)	12758076.0532	344	5426641.7079	3858893.0313	87.5803
		(-1,4500)	7673066.7055	241	2257265.0000	3854972.9298	85.2048
		(5,1000)	10188858.8308	292	3711194.9990	3859918.5485	85.3886

Table 4. The results of Algorithm 1 for randomly generated instances with Euclidean norm.

Test#	n	(s^*, b^*) $L^*: y = s^*x + b^*$	$F(L^*, M, W)$	Iter	Total cost (TC)	$F(L^*, \widetilde{M}, W)$ (Respect to new coordinates)	CPU time (in sec.)
Test 1	100	(1,1)	1129.9566	3	11.0000	1057.1246	0.1051
		(3,4)	2412.1853	37	747.1999	1058.0981	0.1138
		(-1,10)	1329.3607	18	114.0000	1016.8195	0.1054
		(-3,10)	2028.2848	33	497.9999	1021.0994	0.1219
Test 2	200	(1,1)	60362.1703	6	134.0000	57973.5636	0.5034
		(2,5)	83556.4937	54	10950.0000	54510.8651	0.5906
		(-5,50)	107219.4367	74	22896.2307	56892.7024	0.6276
		(-2,40)	94388.4542	60	16834.8000	56681.1927	0.5812
Test 3	300	(1,1)	90612.1984	16	933.0000	83171.3138	1.9359
		(3,10)	155501.2092	101	28664.3999	85934.5791	2.1947
		(-2,40)	139716.6826	99	26367.0000	79690.3322	2.1915
		(-5,60)	147960.6026	96	22475.3076	86085.3736	2.2855
Test 4	400	(1,1)	118854.7504	19	814.0000	111179.1063	5.4273
		(3,10)	207407.1509	131	36680.7999	114812.1825	5.9171
		(-3,60)	164040.9400	103	17811.6000	106810.0389	5.8077
		(-6,90)	171403.3705	107	18333.5675	109265.4853	6.0949
Test 5	500	(1,1)	147055.5830	27	1439.0000	136349.2792	14.9821
		(3,10)	261173.1444	170	50240.3999	144339.0015	14.6882
		(-2,5)	407653.5280	246	176094.6000	143920.0432	15.2334
		(-5,50)	265600.0818	176	48280.6153	145353.2386	14.4278
Test 6	600	(1,1)	1580158.7959	17	2357.0000	1506448.5709	54.3061
		(3,10)	2444925.4084	174	307316.0000	1515374.5227	55.3257
		(-2,100)	2669398.2741	193	455804.4000	1519135.8376	50.8643
		(-5,150)	2708768.0216	191	400774.6153	1541206.3003	50.2928
Test 7	700	(1,1)	1833688.3968	21	3191.0000	1742788.4059	125.7363
		(2,100)	4065268.1812	281	1136751.0000	1827768.2257	129.6197
		(-1,80)	2925090.0359	208	450838.0000	1762637.6003	129.0926
		(-4,180)	2789884.8694	179	285006.1764	1798264.6270	126.0501
Test 8	800	(1,1)	58419.7480	4	37.0000	57826.4854	234.7745
		(2,-10)	60605.9392	70	1348.8000	52858.4109	232.1157
		(-2,10)	131326.5083	328	43114.2000	57599.7694	240.9410
		(-4,60)	64075.2442	61	975.2941	57370.8319	238.0559
Test 9	900	(2,2)	1255604.0461	193	106929.0000	886716.7813	408.6990
		(3,-100)	1036114.5832	103	23031.20000	878741.9381	407.5148
		(-2,100)	1296032.1551	168	86723.3999	953252.3257	414.4726
		(-4, 80)	1799997.5440	318	356416.7647	953163.7937	422.7831
Test 10	1000	(1,1)	1220889.8612	43	5633.0000	1150692.5426	619.1822
		(2,-10)	١٤١٩٢١٨,٠٣٤٤	١٨٠	١٠٦٠٤٤.6000	1059169.4992	613.7813
		(-2,40)	2530892.6459	415	848841.6000	1145946.3780	649.8552
		(-4,80)	2155745.8884	355	481536.1764	1139476.2653	647.1981

Table 5. The results of Algorithm 1 for Beasley [46] instances with rectilinear norm.

Test	n	(s^*, b^*) $L^*: y = s^*x + b^*$	$F(L^*, M, W)$	Iter	Total cost (TC)	$F(L^*, \widetilde{M}, W)$ (Respect to new coordinates)	CPU time (in sec.)
Ruspini	75	(0.5,0.5)	61732.0000	48	15884.5000	13532.0000	0.0372
		(0.5,10)	56868.0000	49	15569.5000	10659.0000	0.0378
		(1,20)	21293.0000	2	154.0000	19976.0000	0.0268
		(-1,20)	56121.0000	39	20149.0000	20286.0000	0.0348
		(10,0)	200633.0000	58	14309.6000	14643.0000	0.0391
Bongarts	287	(0.5,0.5)	66620.0000	193	17823.0000	10811.0000	2.8759
		(-5,10)	216301.0000	243	33215.5990	11065.0000	2.7088
		(-1,50)	17269.0000	28	254.0000	11108.0000	1.9840
		(10,20)	341049.0000	259	28884.2990	9268.0000	2.5937
		(-10,30)	364751.0000	259	30852.7000	9762.0000	2.3351
TSPLIB	654	(0.5,0.5)	16020357.5000	348	2888596.7500	5655212.0000	81.5013
		(-.5,4000)	60600585.0000	533	8770755.0000	3922230.0000	83.2251
		(10, 5000)	128217077.5000	581	10431639.0000	4019657.5000	84.8196
		(-1,4500)	10851355.0000	133	834585.0000	7804435.0000	80.8064
		(5,1000)	51953190.0000	524	7202316.0000	3939615.0000	82.8275

Table 6. The results of Algorithm 1 for randomly generated instances with rectilinear norm.

Test	n	(s^*, b^*) $L^*: y = s^*x + b^*$	$F(L^*, M, W)$	Iter	Total cost (TC)	$F(L^*, \widetilde{M}, W)$ (Respect to new coordinates)	CPU time (in sec.)
Test 1	100	(1,1)	1598.0000	1	4.0000	1558.0000	0.0601
		(3,4)	7628.0000	67	1514.6666	1210.0000	0.0939
		(-1,10)	1880.0000	13	69.0000	1547.0000	0.0704
		(-3,10)	6414.0000	66	1270.6666	1053.0000	0.0917
Test 2	200	(1,1)	85307.000	1	27.0000	84092.0000	0.4446
		(2,5)	186838.0000	100	35129.5000	61410.0000	0.5811
		(-5,50)	546714.0000	140	72104.8000	59379.0000	0.6408
		(-2,40)	211059.0000	108	47953.0000	55731.0000	0.5950
Test 3	300	(1,1)	128145.0000	10	418.0000	121062.0000	1.8727
		(3,10)	491738.0000	187	85320.0000	87617.0000	2.3506
		(-2,40)	312416.0000	167	68128.0000	78287.0000	2.2729
		(-5,60)	754454.0000	207	91275.2000	90871.0000	2.3908
Test 4	400	(1,1)	168086.0000	11	318.0000	161620.0000	5.4832
		(3,10)	655879.0000	249	112512.3333	118262.0000	6.2804
		(-3,60)	518743.0000	241	81428.0000	109247.0000	6.2998
		(-6,90)	1042606.0000	281	106104.8333	117928.0000	6.4222
Test 5	500	(1,1)	207968.0000	5	79.0000	205479.0000	13.0368
		(3,10)	825902.0000	316	145445.3333	146546.0000	15.2783
		(-2,5)	911541.0000	340	2597773.5000	134785.0000	15.3447
		(-5,50)	1354300.0000	358	169352.8000	148438.0000	15.9268
Test 6	600	(1,1)	2234682.0000	1	36.0000	2229318.0000	46.6109
		(3,10)	7731533.0000	386	1246786.6666	1406379.0000	51.4448
		(-2,100)	5968956.0000	349	1206383.5000	1477796.0000	49.4403
		(-5,150)	13812061.0000	428	1638800.4000	1553338.0000	51.4012
Test 7	700	(1,1)	2593227.0000	1	36.0000	2587863.0000	123.7259
		(2,100)	9090216.0000	451	2236253.0000	1590247.0000	130.5354
		(-1,80)	4136702.0000	196	396427.0000	2590897.0000	127.6060
		(-4,180)	11502990.0000	468	1522739.7500	1744686.0000	132.4554

Test 8	800	(1,1)	82618.0000	4	37.0000	81779.0000	231.2939
		(2,-10)	135519.0000	342	19534.5000	51678.0000	246.2092
		(-2,10)	293655.0000	489	72232.5000	56725.0000	246.3501
		(-4,60)	264189.0000	461	30344.2500	56858.0000	245.7794
Test 9	900	(2,2)	2807616.0000	454	486468.0000	928289.0000	415.1788
		(3,-100)	3276482.0000	509	468106.3333	877335.0000	409.0081
		(-2,100)	2898016.0000	461	497564.0000	957975.0000	408.2063
		(-4, 80)	7421580.0000	623	1083496.7500	1003237.0000	412.4093
Test 10	1000	(1,1)	176599.0000	43	5633.0000	1627325.0000	633.9228
		(2,-10)	3173468.0000	503	594102.5000	992648.0000	637.9739
		(-2,40)	5659248.0000	630	1525060.0000	1073947.0000	626.8892
		(-4,80)	8888368.0000	678	1388494.5000	1159789.0000	633.6660

In Figure 1, the diagram depicts the changes in the value of the median line objective function and total cost for the Bongartz instance in the Euclidean norm case. The specific parameters used for this illustration are s*=10 and b*=20. The graph visually represents how these values fluctuate over a given range or period, providing insight into the relationship between the median line objective function and total cost in this particular scenario.

The results indicate that the proposed method efficiently solve the inverse median line location problem with contrast to solving nonlinear programming model with infinite constraints.

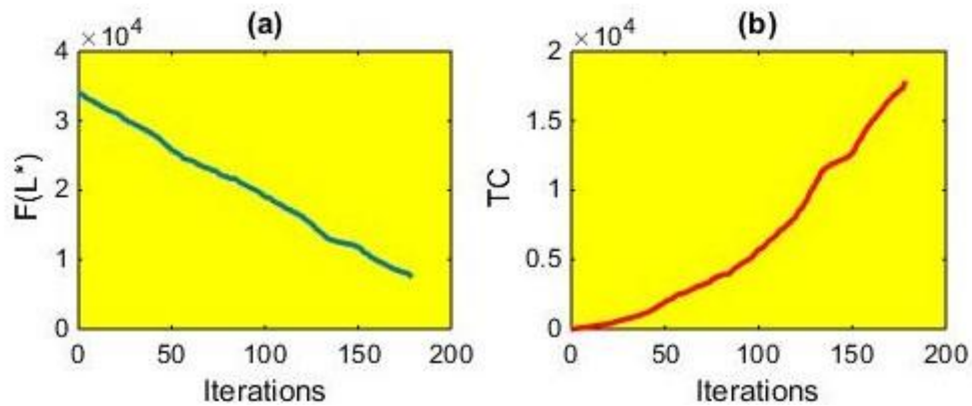

Figure 1. Diagrams for Bongartz instance with $s^* = 10, b^* = 20$ and Euclidean norm. (a) Changes of $F(L^*, \widetilde{M}, W)$ over iterations. (b) Total cost changes over iterations.

5.2.1 Sensitivity analysis

We investigate the sensitivity of the runtime of the algorithm as well as the termination iteration to the choice of L^* by evaluating some different lines for each instance. As shown in Tables 3–6, selecting L^* close to L_{best} (see Table 2) decreases the number of iterations and the runtime, although it influences the coordinate-change cost. In contrast, the final objective function value, $F(L^*, \widetilde{M}, W)$, is largely unaffected by this selection.

For example, consider Figures 2 and 3 which depict the lines $L^*$and L_{best}, and the location of points in initial form and after modifying for the Bongartz instance with L_2 norm, for the cases $(s^*, b^*) = (1,10)$ and $(s^*, b^*) = (10,20)$, respectively. In the first case, line L^* passes through the middle of existing points, indicating that there is no need to transfer many points. As a result, the algorithm terminated after a few iterations (as shown in Table 3, with a total of 43 terminated iterations). In the case where $(s^*, b^*) = (10,20)$, line L^* lies outside of the existing points. In order to make L^* a median line in this case, many points should be transferred. Table 3 indicates that in this scenario, the algorithm terminated after 177 iterations, meaning that 177 out of 287 points needed to be transferred. Thus, after transferring more than half of the points are located on line L^*.

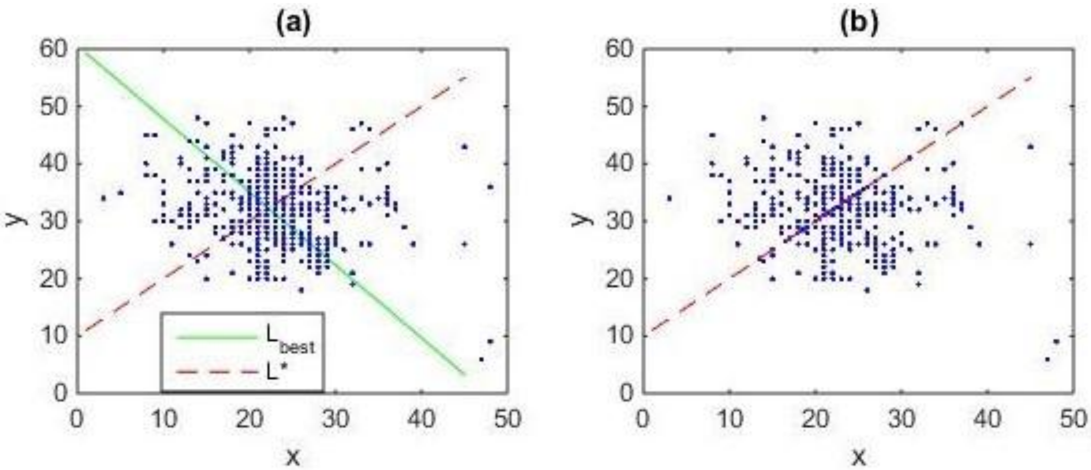

Figure 2. The coordinates of points, best line L_{best} and given line L^* for Bongartz instance with $s^* = 1,\ b^* = 10$ and Euclidean norm. (a) The initial coordinates. (b) The modified coordinates.

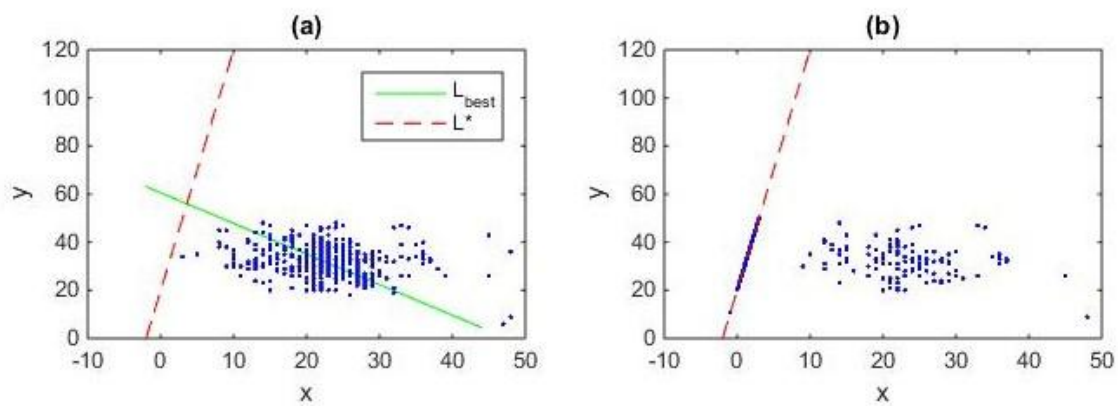

Figure 3. The coordinates of points, best line L_{best} and given line L^* for Bongartz instance with $s^* = 10, b^* = 20$ and Euclidean norm. (a) The initial coordinates. (b) The modified coordinates.

6. Conclusion

The line location problem, recognized as a specialized instance of the broader class of hyperplane location problems, continues to be a central topic of interest across location theory. This research concentrated on introducing novel methodologies for identifying optimal lines from the standpoint of location theory. Within the classical line location paradigm, a set of planar demand points, each assigned a non-negative weight indicative of its relative importance, is a fixed input. Given that decisions concerning line facility placement typically carry substantial long-term ramifications, the physical relocation of these lines often proves prohibitively expensive or practically infeasible. Consequently, in numerous practical settings, the pragmatic necessity arises to adjust associated problem parameters in order to optimize a pre-existing, fixed line relative to the surrounding demand points. Modifying these parameters is generally a more cost-effective and viable strategy than physically altering the already installed line infrastructure.

In the present study, we addressed the inverse median line location problem by analyzing two distinct scenarios: modifying the importance weights of the demand points and adjusting their spatial coordinates. The models were developed using both the rectilinear and Euclidean norms. The proposed formulations are designed to ascertain the desired weights or coordinates that minimize the total modification cost while ensuring the specified line achieves the status of the optimal median line facility. Although the initial mathematical constructions involved an ostensibly infinite number of constraints, we successfully exploited specific problem properties and structural features to reformulate the variable-weight case as a finite-dimensional linear optimization problem. For the more complex scenario of coordinate modification, we devised a dedicated algorithm structured to effectively manage the inherent computational complexities. The validity of the proposed solution methods was substantiated through several illustrative numerical examples.

The results obtained definitively confirm that the developed algorithm constitutes an effective and practical tool for solving inverse median line location problems, thereby enriching the methodological resources available to decision-makers managing fixed-line infrastructures. This research offers both theoretical and practical contributions by establishing new inverse optimization frameworks and corresponding solution strategies applicable to real-world facility location challenges.

Future scholarly endeavors could logically pursue the extension of these models to higher dimensions, regions with block norms (see e.g. [47]), the incorporation of uncertainty into the parameter modifications, goal Weber facility location model (see e.g. [48]), balancing location model (see e.g. [49]) or the investigation of inverse formulations pertaining to other geometric location problem classes.

Conflict of interest The authors declare that they have no conflict of interest.

Funding No funding was received for this research.

Data availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] I. Espejo and A. M. Rodríguez-Chía, "Simultaneous location of a service facility and a rapid transit line," *Computers and Operations Research,* vol. 39, pp. 2899-2903, 2012.

[2] K. P. Bennett and O. L. Mangasaria, "Robust linear programming discrimination of two linearly inseparable sets," *Optimization Methods and Software,* vol. 1, pp. 23-34, 1992.

[3] O. Mangasarian, "Arbitrary-norm separating plane," *Operations Research Letters,* vol. 24, pp. 15-23, 1999.

[4] V. Blanco, A. Japón, D. Ponce and P. J., "On the multisource hyperplanes location problem to fitting set of points," *Computers & Operations Research,* vol. 128, pp. 105-124, 2021.

[5] F. Baroughi Bonab, R. E. Burkard and E. Gassner, "Inverse p-median problems with variable edge lengths," *Mathematical methods of operations research,* vol. 73, no. 2, pp. 263-280, 2011.

[6] R. E. Burkard, E. Gassner and J. Hatzl, "A linear time algorithm for the reverse 1-median problem on a cycle," *Networks: An International Journal,* vol. 48, no. 1, pp. 16-23, 2006.

[7] G. Wesolowsky, "Location of the median line for weighted points," *Environment and Planning A,* vol. 7, pp. 163-170, 1975.

[8] J. G. Morris and J. P. Norback, "A simple approach to linear facility location," *Transportation Science,* vol. 14, no. 1, pp. 1-8, 1980.

[9] J. Morris and J. Norback, "Linear facility location-solving extensions on the basic problems," *European Journal of Operations Research,* vol. 12, no. 1, pp. 90-94, 1983.

[10] D. T. Lee and Y. F. Wu, "Geometric complexity of some location problems," *Algorithmica,* vol. 1, no. 1-4, pp. 193-211, 1986.

[11] N. Megiddo and A. Tamir, "Finding least-distances lines," *SIAM Journal on Algebraic Discrete Methods,* vol. 4, no. 2, pp. 207-211, 1983.

[12] D. T. Lee and Y. T. Ching, "The power of geometric duality revisited," *Information Processing Letters,* vol. 21, no. 3, pp. 117-122, 1985.

[13] N. M. Korneenko and H. Martini, "Approximating finite weighted point sets by hyperplanes," *Lecture Notes in Computer Science,* vol. 447, pp. 276-286, 1990.

[14] A. Schöbel, "Locating least-distant lines in the plane," *European Journal of Operational Research,* vol. 106, no. 1, pp. 152-159, 1998.

[15] A. Schöbel, "Solving restricted line location problems via a dual interpretation," *Discrete applied mathematics,* vol. 93, no. 1, pp. 109-125, 1999.

[16] A. Schöbel, Locating lines and hyperplanes: theory and algorithms, vol. 25, Springer Science & Business Media, 2013.

[17] J. Brimberg, H. Juel and A. Schoebel, "Linear facility location in three dimensions- models and solution methods," *Operations Research,* vol. 50, pp. 1050-1057, 2002.

[18] Z. Drezner and G. Wesolowsky, "Location of an obnoxious route," *Journal of Operational Research Society,* vol. 40, pp. 1011-1018, 1989.

[19] J. M. Díaz-Báñez, M. A. López and J. A. Sellares, "Locating an obnoxious plane," *European journal of operational research,* vol. 173, no. 2, pp. 556-564, 2006.

[20] J. M. Díaz-Báñez, P. A. Ramos and P. Sabariego, "The maximin line problem with regional demand," *European journal of operational research,* vol. 181, no. 1, pp. 20-29, 2007.

[21] M. Golpayegani, J. Fathali and E. Khosravian, "Median line location problem with positive and negative weights and Euclidean norm," *Neural Computing and Applications,* vol. 24, pp. 613-619, 2014.

[22] M. Golpayegani, J. Fathali and H. Moradi, "A particle swarm optimization method for semi-obnoxious line location problem with rectilinear norm," *Computers & Industrial Engineering,* vol. 109, pp. 71-78, 2017.

[23] M. Golpayegani, F. J. and H. Moradi, "Bi-objective location of a semi-desirable line facility and evolutionary solution algorithms," *Evolutionary Intelligence,* vol. 19, no. 2, p. 33, 2026.

[24] P. Agarwal, A. Efrat, M. Sharir and S. Toledo, "Computing a segment center for a planar point set," *Journal of Algorithms,* vol. 15, no. 2, pp. 314-323, 1993.

[25] A. Efrat and M. Sharir, "A near-linear algorithm for the planar segment-center problem," *Discrete & Computational Geometry,* vol. 16, no. 3, pp. 239-257, 1996.

[26] J. A. Barcia, J. M. Dıaz-Bánez, A. J. Lozano and I. Ventura, "Computing an obnoxious anchored segment," *Operations Research Letters,* vol. 31, no. 4, pp. 293-300, 2003.

[27] M. Golpayegani and F. J., "Convexity and sensitivity analysis of the median line location problem," *International Journal of Systems Science: Operations & Logistics,* vol. 12, no. 1, p. 252996, 2025.

[28] O. Berman, D. I. Ingco and A. Odoni, "Improving the location of minimax facilities through network modification.," *Networks,* vol. 24, no. 1, pp. 31-41, 1994.

[29] M. C. Cai, X. G. Yang and J. Z. Zhang, "The complexity analysis of the inverse center location problem," *Journal of Global Optimization,* vol. 15, no. 2, pp. 213-218, 1999.

[30] J. Zhang, Z. Liu and Z. Ma, "Some reverse location problems," *European Journal of Operational Research,* vol. 124, no. 1, pp. 77-88, 2000.

[31] B. Alizadeh, R. E. Burkard and U. Pferschy, "Inverse 1-center location problems with edge length augmentation on trees," *Computing,* vol. 86, no. 4, pp. 331-343, 2009.

[32] B. Alizadeh and R. E. Burkard, "Combinatorial algorithms for inverse absolute and vertex 1-center location problems on trees," *Networks,* vol. 58, no. 3, pp. 190-200, 2011.

[33] B. Jana, S. Mondal and M. Pal, "Computation of inverse 1-center location problem on the weighted trees," *CiiT htemational Joumal ofNetworking and communication Engineering,* vol. 4, no. 2, pp. 70-75, 2012.

[34] K. T. Nguyen, "Reverse 1-center problem on weighted trees," *Optimization,* vol. 65, no. 1, pp. 253-264, 2016.

[35] M. Golpayegani, H. Moradi and R. Tavakkoli-Moghaddam, "Mathematical Modelling for a Semi-obnoxious Inverse Line Location Problem," in *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM*, Bangkok, Thiland, 2018.

[36] B. Alizadeh and R. Etemad, "Optimal algorithms for inverse vertex obnoxious center location problems on graphs," *Theoretical Computer Science,* vol. 707, pp. 36-45, 2018.

[37] I. Keshtkar and M. Ghiyasvand, "Inverse quickest center location problem on a tree," *Discrete Applied Mathematics,* vol. 260, pp. 188-202, 2019.

[38] S. Omidi, J. Fathali and M. Nazari, "Inverse and reverse balanced facility location problems with variable edge lengths on trees," *Opsearch,* vol. 57, pp. 261-273, 2020.

[39] M. Gholami and J. Fathali, "Inverse minimax circle location problem with variable coordinates," *AUT Journal of Mathematics and Computing,* vol. 3, pp. 137-146, 2023.

[40] M. Gholami and J. Fathali, "The inverse minisum circle location problem," *Yugoslav Journal of Operations Research,* vol. 32, pp. 153-165, 2022.

[41] M. Gholami and J. Fathali, "Mathematical models for the variable weights version of the inverse minimax circle location problem," *Journal of Mathematical Modeling,* vol. 9, no. 1, pp. 137-144, 2021.

[42] S. Mohammadi, B. Alizadeh, F. Baroughi and E. Afrashteh, "A modified directional bat algorithm for extensive inverse p-facility maxian location problems on networks," *Soft Computing,* vol. 26, no. 4, pp. 1941-1959, 2022.

[43] S. Omidi and J. Fathali, "Inverse single facility location problem on a tree with balancing on the distance of server to clients," *Journal of Industrial and Management Optimization,* vol. 18, no. 2, pp. 1247-1259, 2022.

[44] J. Fathali, "A row generation method for the inverse continuous facility location problems," *Computers & Industrial Engineering,* vol. 171, p. 108482, 2022.

[45] M. Golpayegani and F. J., "Greedy algorithms for the inverse center line location problem," *Expert Systems with Applications,* vol. 296, p. 129064, 2025.

[46] J. Beasley, "OR-Library: distributing test problems by electronic mail," *Journal of the Operational Research Society,* vol. 41, pp. 1069-1072, 1990.

[47] J. Fathali and M. Zaferanieh, "Location problems in regions with Lp and block norms," *Iranian Journal of Operations Research,* pp. 72-87, 2011.

[48] A. Soleimani Kourandeh, J. Fathali and S. Taherifard, "Solving single facility goal Weber location problem using stochastic optimization methods," *Iranian Journal of Operations Research,* pp. 1-19, 2021.

[49] T. Sayar, J. Fathali and M. Ghiyasi, "The problem of balancing allocation with regard to the efficiency of servers," *Iranian Journal of Operations Research,* pp. 29-47, 2018.