Computational and Applied Mathematics (2022) 41:407
https://doi.org/10.1007/s40314-022-02128-2

New approaches to the robust 1-center location problems on tree networks

Soudabeh Seyyedi Ghomi[1] · Fahimeh Baroughi[1]

Received: 27 June 2021 / Revised: 13 November 2022 / Accepted: 15 November 2022 /
Published online: 23 November 2022

Abstract
This paper is concerned with the robust 1-center location problems on the tree networks under minmax and minmax regret criteria, where the vertex weights and edge lengths of the underlying tree are considered as dynamic data or discrete set of scenarios. In the problem under minmax criterion, the aim is to find a point on the tree such that it minimizes the maximum cost. The problem under minmax regret asks to find a point on the tree such that minimizes the maximum regret. We develop the first optimal solution algorithms with polynomial time complexities for the problems under investigation.

Keywords Minmax regret · Robust optimization · Center problem · Location problem

Mathematics Subject Classification 90C27 · 90C17

1 Introduction

Location problems have received strong theoretical interest due to their relevance in practice. In a location problem on the network, the aim is to find the best locations for establishing one or several facilities such that these facilities serve all clients on the network. One of the most important location problems is the p-center location problem, which has many applications in practice. In this problem the task is to find p locations for facilities which minimize the maximum of weighted distances of each customer to its closest facility.
The classical p-center location problem was first investigated by Hakimi (1965). Hakimi showed that this problem on general networks can not be solved in a polynomial time. Kariv and Hakimi (1979) proposed a combinatorial algorithm with $O(mn \log n)$ time complexity for the 1-center location problem on general graphs with m edges and n vertices. For the case that the underlying graph is a tree, they reduced this time to $O(n \log n)$. Megiddo designed

Communicated by Carlos Hoppen.

✉ Fahimeh Baroughi
baroughi@sut.ac.ir

[1] Department of Applied Mathematics, Faculty of Basic Sciences, Sahand University of Technology, Tabriz, Iran

a linear time algorithm for solving the 1-center location problem on trees (Megiddo 1983). Megiddo and Tamir also considered this problem in Megiddo and Tamir (1983) and proposed an $O(n \log^3 n)$ time algorithm for the continuous p-center location problem on tree networks. Also, they presented an $O(n \log^2 n \log \log n)$ time algorithm for a weighted discrete p-center location problem.

Kouvelis et al. introduced the idea of minmax and minmax regret approaches to obtain a robust solution of the 1-median problem (Kouvelis et al. 1993). These methods minimize the maximum cost of decision maker's decision and maximum difference between the cost of decision-maker's decision and the cost of optimal decision, respectively. In the minmax and minmax regret criteria on networks, vertex weights and edge lengths are uncertain, and this uncertainty can be consider as interval, discrete set of scenarios, dynamic, polyhedral, etc.

Averbakh showed that the minmax regret 1-center location problem on general graphs with interval edge lengths is NP-hard (Averbakh 2003). This problem on general graphs with uncertain vertex weights can be solved in a polynomial time. Averbakh and Berman considered the minmax regret 1-center location problem with interval vertex weights on general graphs and tree networks and presented combinatorial algorithms with $O(m^2 n \log n)$ and $O(n^2)$ running times, respectively (Averbakh and Berman 1997). Lin et al. improved these times to $O(mn \log n)$ and $O(n \log^2 n)$, respectively (Lin et al. 2006). The robust 1-center location problem on tree networks with interval vertex weights and edge lengths was solved in $O(n^6)$ time by Averbakh and Berman (2000). They reduced this time to $O(n^2 \log n)$ on un-weighted trees. Burkard and Dollani considered this problem and proposed combinatorial algorithms for the problem with $O(n^3 \log n)$ and $O(n \log n)$ times on weighted trees and un-weighted trees, respectively (Burkard and Dollani 2002). Aloulou et al. (2005) presented an algorithm with $O(mn(\log n + |S|)|S|)$ time complexity for the robust 1-center location problem with uncertain vertex weights on general graphs and an algorithm with $O(n(n + \log n)|S|)$ time complexity with uncertain vertex weights on tree graphs, where the type of uncertainty is the discrete set of scenarios and S is the set of scenarios.

Bhattacharya et al. proposed algorithms with $O(n)$, $O(n \log n)$ and $O(n \log^2 n)$ times for the minmax regret 1-center location problem on path, cycle and tree networks with interval vertex weights, respectively (Bhattacharya et al. 2012). Also, Bhattacharya et al. considered the minmax regret 1-center location problem on tree, unicycle and cactus graphs with interval vertex weights and obtained algorithms with $O(n \log n)$, $O(n \log n)$ and $O(n \log^2 n)$ on these graphs, respectively (Bhattacharya et al. 2015). This problem on cactus networks with c cycles solved in Bhattacharya et al. (2014).

In this paper, we consider the robust vertex and absolute 1-center location problems on the tree networks. We use the minmax and minmax regret criteria for obtaining the robust solution. The vertex weights and the edge lengths of the tree network are considered as dynamic data and discrete set of scenarios. These problems have not been investigated up to now with these types of data for the vertex weights and edge lengths. This paper is organized as follows. Section 2 introduces the basic definitions, preliminaries and reviews some facts that are used throughout the paper. In Sect. 3, we consider the robust vertex 1-center location problem on dynamic tree networks with minmax criterion and present a polynomial time algorithm for this problem. In Sect. 4, we solve this problem with minmax regret criterion in a polynomial time. In Sects. 5 and 6, we consider the robust absolute 1-center location problem on the dynamic tree networks and solve this problem under minmax and minmax regret criteria, respectively. Section 7 considers the robust 1-center location problem on tree networks with discrete set of scenarios for vertex weights and edge lengths. Finally, we present the concludes of paper in Sect. 8.

2 Problem statement

Let $T = (V, E)$ be a tree network with vertex set $V = \{v_1, \ldots, v_n\}$ and edge set E. Each vertex $v \in V$ is associated with a weight w_i and each edge e is associated with a non-negative length l_e. Notice that the negative weights are allowed for vertices of tree T. For all $x, y \in T$, let $d(x, y)$ be the length of shortest path between x, y. The classical absolute (vertex) 1-center location problem wants to find $x \in T$ $(x \in V)$ such that it minimizes the following objective function:

$$f(x) = \max_{i=1,\ldots,n} w_i d(v_i, x) \tag{1}$$

in Kariv and Hakimi (1979) showed that $d(v_i, x), x \in e$ on general graphs is a linear function or contains two linear functions. Therefore $f(x)$ is a piecewise linear and continuous function on each edge e on the graph. They used this property and solved the classical 1-center location problem on general graphs and tree networks. The following Lemma is a conclusion of their article.

Lemma 2.1 (Kariv and Hakimi 1979) *An optimal solution of the classical absolute 1-center location problem on the tree network is obtained in* $O(n \log n)$ *time.*

Dearing and Francis considered the center location problem on networks and obtained some properties of this problem (Dearing and Francis 1974). We use one of these results, i.e., Lemma 2.1 in this paper.

Lemma 2.2 (Dearing and Francis 1974) *The optimal value of the classical absolute 1-center location problem equals to*

$$\max_{1 \le i,j \le n} \frac{w_i w_j d(v_i, v_j)}{w_i + w_j}.$$

Also, if

$$\frac{w_{i^*} w_{j^*} d(v_{i^*}, v_{j^*})}{w_{i^*} + w_{j^*}} = \max_{1 \le i,j \le n} \frac{w_i w_j d(v_i, v_j)}{w_i + w_j},$$

then x^* *is an absolute 1-center location, where*

$$d(x^*, v_{i^*}) = \frac{w_{j^*} d(v_{i^*}, v_{j^*})}{w_{i^*} + w_{j^*}}.$$

In the real-life problems, the input data are not characterized precisely and many factors have effect on these parameters. Therefore, the input data of problems may be uncertain. In the robust optimization problems, this uncertainty is considered as interval data, discrete set of scenarios, dynamic data and etc.

In the next sections, the vertex and absolute 1-center location problems are investigated such that these problems contains dynamic data and discrete set of scenarios. First we introduce the robust 1-center location problem under general uncertainty. Consider tree network T. Suppose that the weight of vertices and length of edges in T are uncertain. Let S represents the set of all realizations of the vertex weights and edge lengths. This set is called the set of scenarios and each $s \in S$ is a scenario. The classical 1-center location problem under scenario $s \in S$ is defined as

$$\min_{x \in T} (\min_{x \in V}) f(x, s),$$

in which

$$f(x,s) = \max_{i=1,\ldots,n} w_i^s d^s(v_i, x).$$

The robust 1-center location problem on uncertain trees can be consider under two criteria, i.e. minmax criterion (worst case criterion) and minmax regret criterion (robust deviation and relative robustness). The first one selects a point on tree T such that it minimizes the maximum value of the objective function on all scenarios. If we use this criterion for absolute (vertex) 1-center location problem, then we have the following formulation:

$$\min_{x\in T}(\min_{x\in V}) \max_{s\in S} f(x,s). \tag{2}$$

The optimal solution of this problem is called absolute robust solution. If the difference between $f(x,s)$ and $f(x^*,s)$ is considered by $R(x,s)$ for all $x \in T$ and $s \in S$, i.e. $R(x,s) = f(x,s) - f(x^*,s)$, then the robust deviation criterion for the absolute (vertex) 1-center location problem is defined as follows:

$$\min_{x\in T}(\min_{x\in V}) \max_{s\in S} R(x,s). \tag{3}$$

Furthermore, the relative robustness criterion for the described problem is defined as

$$\min_{x\in T}(\min_{x\in V}) \max_{s\in S} \frac{R(x,s)}{f(x^*,s)}. \tag{4}$$

In the deviation and relative robustness the regret for a specific location is computed by calculating either the difference or the ratio between the maximum transportation cost of the specific location under a scenario and the maximum transportation cost of the optimal location for this scenario.

3 Dynamic robust vertex 1-center location problem with minmax criterion

Suppose that the vertex weights and the edge lengths of tree T are linear functions of parameter t and t varying in the interval $[l, u]$. Assume that the weight of vertex v_i equals to $w_i(t) = a_i t + b_i$, $1 \le i \le n$, and the length of edge e equals to $l_e(t) = a_e t + b_e$, $e \in E$ with $l_e(t) > 0$ for any value of t on interval $[l, u]$. Notice that the vertex weights are allowed to become negative. The dynamic robust vertex 1-center location problem under the minmax criterion is formulated as follows:

$$\min_{x\in V} \max_{t\in[l,u]} f(x,t), \tag{5}$$

where

$$f(x,t) = \max_{i=1,\ldots,n} w_i(t) d^t(x, v_i).$$

According to the definition of vertex weights and edges lengths, we can rewrite $f(x,t)$ at given vertex x as follows:

$$\begin{aligned} f(x,t) &= \max_{i=1,\ldots,n} w_i(t) d^t(x, v_i) \\ &= \max_{i=1,\ldots,n} (a_i t + b_i)\left(\left(\sum_{e\in E(P_i)} a_e\right) t + \sum_{e\in E(P_i)} b_e\right) \end{aligned}$$

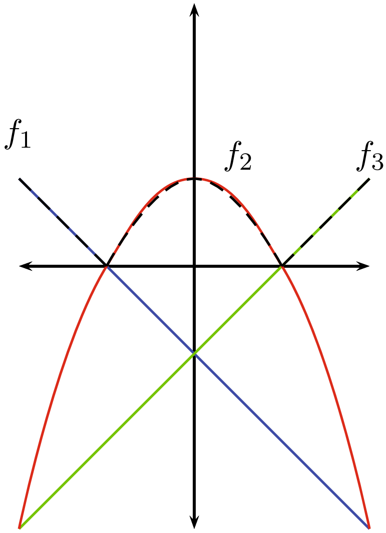

Fig. 1 The upper envelope of functions f_1, f_2, f_3

$$= \max_{i=1,\ldots,n} \left(A_i t^2 + B_i t + C_i\right),$$

where P_i is the unique path between x and v_i, $A_i = a_i(\sum_{e\in E(P_i)} a_e)$, $B_i = (a_i(\sum_{e\in E(P_i)} b_e) + b_i(\sum_{e\in E(P_i)} a_e))$ and $C_i = b_i(\sum_{e\in E(P_i)} b_e)$. If we define $f_i(t) = A_i t^2 + B_i t + C_i$, $i = 1, \ldots\ldots, n$, then

$$f(x,t) = \max_{i=1,\ldots,n} f_i(t).$$

The functions $f_i(t)$, $i = 1, \ldots, n$ have the following properties:

1. $f_i(t)$ is a continuous function on interval $[l, u]$.
2. $f_i(t)$ is a totally defined on interval $[l, u]$.
3. $f_i(t)$ is a univariate function. Also, for each $1 \le i \ne j \le n$, $f_i(t)$ and $f_j(t)$ intersect in at most two points.

Consequently, for all $x \in V$, $f(x,t)$ is the upper envelope of n continuous, totally defined and univariate functions which each pair of them intersect at most two points. Therefore, $f(x,t)$ is a continuous piecewise function such that each piece of it is either a quadratic function or linear function (see Fig. 1). If we want to obtain an optimal solution for (5) first, we must compute $f(x,t)$ on interval $[l, u]$, for every vertex $x \in V$. For this purpose, we can use the following Lemma, which it has been proven in Agarwal et al. (1996), because $f_i(t)$, $1 \le i \le n$ satisfy in the conditions of Lemma 3.1.

Lemma 3.1 (Agarwal et al. 1996) *Let $\mathcal{F} = \{f_1, f_2, \ldots, f_n\}$ be a collection of n continuous, totally defined and univariate functions that each pair of them intersect in at most s points. Then, the total pieces of upper envelope of these n functions are equal to $\lambda_s(n)$.*

The value of $\lambda_s(n)$ can be calculated using the following theorem, which is described in Agarwal et al. (1996).

Theorem 3.2 (Agarwal et al. 1996) *Let n be the number of univariate and continuous functions. Then we have*

i. $\lambda_1(n) = n$,
ii. $\lambda_2(n) = 2n - 1$,
iii. $\lambda_3(n) = O(n\alpha(n))$

in which $\alpha(n)$ is the inverse of Ackerman's function.

For more information about Ackerman's function, see Ackermann (1928). According to Lemma 3.1 and Theorem 3.2, for each vertex $v \in V$, the number of pieces for the piecewise function $f(x,t)$ on the interval $[l,u]$ is $O(\lambda_2(n)) = O(n)$. Suppose that this number is denoted by m. In Agarwal et al. (1996), a method is presented that explains how to calculate these pieces and their corresponding intervals with required time complexity. The important results of this method are mentioned in the following two theorems.

Theorem 3.3 (Agarwal et al. 1996) *The upper envelope of an arbitrary collection contains n continuous, totally defined and univariate functions that each pair of them intersect in at most s points, can be constructed in $O(\lambda_s(n)\log n)$ time.*

We use from Theorem 3.3 and the proposed method presented in its proof to compute the pieces of function $f(x,t)$ as follows. For a given vertex $x \in V$, let $\{i_1, \ldots, i_m\}$ be a permutation of the index of $\{f_1(t), \ldots, f_n(t)\}$ such that

$$\begin{aligned} f(x,t) =& f_{i_1}(t), \qquad \forall t \in I_1 = [t_0, t_1], \\ f(x,t) =& f_{i_2}(t), \qquad \forall t \in I_2 = [t_1, t_2], \\ &\vdots \\ f(x,t) =& f_{i_m}, \qquad \forall t \in I_m = [t_{m-1}, t_m], \end{aligned}$$

where $[l,u] = \bigcup_{j=1}^{m} I_j$. The used process is described in Example 3.6. After calculating the upper envelope and breakpoints, we must find the points that maximize the each piece of upper envelope.

If the first derivative of the function f_{i_j} is denoted by $f_{i_j}^{'}$, then one of the points that this function obtains its maximum, is t_j^* such that, $f_{i_j}^{'}(t_j^*) = 0$. Point t_j^* is called a critical point of $f_{i_j}^{'}(t)$. For example, if $f(t) = at^2 + bt + c$, then

$$f^{'}(t) = 0 \Rightarrow 2at + b = 0 \Rightarrow t^* = -b/2a.$$

However, we know that the maximum of $f_{i_j}(t)$ on interval I_j is achieved at $t = t_{j-1}$ or $t = t_j$. If $t_{j-1} \le t_j^* \le t_j$, then we get the following theorem.

Theorem 3.4 *For a given vertex x on tree T, the value of objective function of the robust vertex $1-$center location problem with minmax criterion is obtained as*

$$\max_{l \le t \le u} f(x,t) = \max\{f_{i_1}(x,t_0), f_{i_1}(x,t_1^*), f_{i_1}(x,t_1), \ldots, \\ f_{i_m}(x,t_{m-1}), f_{i_m}(x,t_m^*), f_{i_m}(x,t_m)\}.$$

Thus, $\max_{l \le t \le u} f(x,t)$ is equivalent to choosing the largest member of a set with $O(n)$ members. This selection is performed in linear time. Consequently, for a vertex x on tree T, $\max_{l \le t \le u} f(x,t)$ is calculated in $O(n \log n)$ time. If these operations are repeated for all vertices of tree T, then we get the following Lemma.

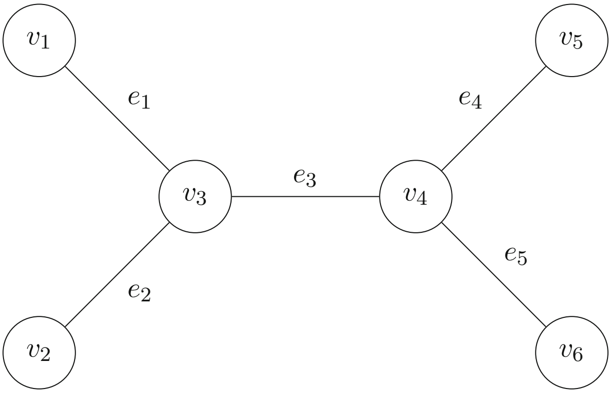

Fig. 2 The tree network of Example 3.6

Table 1 The data of dynamic tree

i	$w_i(t)$	$l_{e_i}(t)$
1	$3t+1$	$3t+3$
2	4	2
3	$t-1$	$t+1$
4	$2t+1$	1
5	3	$2t+2$
6	1	–

Lemma 3.5 *An optimal solution for the dynamic robust vertex 1-center location problem under minmax criterion on tree networks with linear vertex weights and linear positive edge lengths can be computed in $O(n^2 \log n)$ time.*

Example 3.6 Consider the given tree network in Fig. 2with dynamic edge lengths and dynamic vertex weights.

Assume that the vertex weights and edge lengths are given in Table 1. Using Theorem 3.3, we can obtain maps of $f(x,t)$ on interval $[-1,3]$ for each $x \in \{v_1, v_2, ..., v_6\}$. These computed maps are given in Table 2. For example, we explain how to calculate $f(x,t)$ for $x = v_1$ on interval $[-1,3]$. First we compute $f_i(v_1,t)$ for $i = 1, ..., 6$. We set

$$\mathcal{F} = \{0, 12t+20, 3t^2-3, 8t^2+12t+4, 12t+15, 6t+6\}.$$

Then we partition set $\mathcal{F}$ into two sets $\mathcal{F}_1$ and $\mathcal{F}_2$ such that $|\mathcal{F}_i| \le 3$, $i = 1,2$. We calculate the breakpoints of all functions in set $\mathcal{F}_i$, $i = 1,2$ and we put all these points in set $V = \{-1, 0.25, 1, \frac{\sqrt{352}}{16}, 3\}$ so that $v_i \le v_i + 1$ for $i = 1, ..., t-1$. There are functions $f_i^{(1)} \in \mathcal{F}_1$ and $f_i^{(2)} \in \mathcal{F}_2$ that have the highest value on the interval $[v_i, v_i+1]$, for $i = 1, ..., t-1$. In this example, $f_i^{(1)} = 12t+20$ and $f_i^{(2)} = 12t+15$ for $i = 1, ..., t-1$. Next, we find the roots of $f_i^{(1)} - f_i^{(2)}$ on interval $[v_i, v_i+1]$. We add these roots to set V and call the new set $V^{'} = \{v_1^{'}, ..., v_{t^{'}}^{'}\}$. Here, $V^{'} = V$. It is clear that for $i = 1, ..., t^{'}$, there is a unique function $f_i \in \mathcal{F}$ such that $f(v_1,t) = f_i$ on interval $\left[v_i^{'}, v_{i+1}^{'}\right]$ for $i = 1, ..., t^{'}$. Here, $f_i = 12t+20$, for $i = 1, ..., t-1$. Therefore, $f(v_1,t) = 12t+20$ on interval $[-1,3]$.

Table 2 The values of $f(x,t)$

x	$f(x,t)$
v_1	$12t+20,\ \ t \in [-1,3]$
v_2	$3t+12, \quad t \in \left[-1, \frac{-15+\sqrt{477}}{18}\right]$
	$9t^2+18t+5, \quad t \in \left[\frac{-15+\sqrt{477}}{18}, 3\right]$
v_3	$8 \quad t \in \left[-1, \frac{1}{3}\right]$
	$9t^2+12t+4 \quad t \in \left[\frac{1}{3}, 3\right]$
v_4	$4t+12 \quad t \in \left[-1, \frac{-3+\sqrt{33}}{6}\right]$
	$12t^2+16t+4 \quad t \in \left[\frac{-3+\sqrt{33}}{6}, 3\right]$
v_5	$4t+16 \quad t \in \left[-1, \frac{-19+\sqrt{753}}{24}\right]$
	$12t^2+19t+5\, t \in \left[\frac{-19+\sqrt{753}}{24}, 3\right]$
v_6	$12t+20 \quad t \in \left[-1, \frac{-12+\sqrt{1152}}{32}\right]$
	$18t^2+24t+6 \quad t \in \left[\frac{-12+\sqrt{1152}}{32}, 3\right]$

As a result, a robust solution of the robust vertex 1-center location problem with minmax criterion is obtained as

$$\min_{x\in\{v_1,\ldots,v_6\}} \max_{t\in[-1,3]} f(x,t) = \min\{f(v_1,-1),\ f(v_1,3),\ f(v_2,-1),\ f\left(v_2, \frac{-15+\sqrt{477}}{18}\right),$$

$$f(v_2,3),\ f(v_3,-1),\ f(v_3,\frac{1}{3}),\ f(v_3,3),\ f(v_4,-1),\ f\left(v_4, \frac{-3+\sqrt{33}}{6}\right),\ f(v_4,3),$$

$$f(v_5,-1),\ f\left(v_5, \frac{-19+\sqrt{753}}{24}\right),\ f(v_5,3),\ f(v_6,-1),$$

$$f\left(v_6, \frac{-12+\sqrt{1152}}{32}\right),\ f(v_6,3)\} = f(v_5,3).$$

Then v_5 is a location of robust vertex 1-center location on described tree.

4 Dynamic robust vertex 1-center location problem with minmax regret criterion

In this section, we use the minmax regret criterion to obtain a robust solution of the vertex 1-center location problem on tree networks with dynamic edge lengths and dynamic vertex weights. Suppose that the vertex weights and the edge lengths of tree T are the linear functions of parameter t. Using these assumptions, the minmax regret vertex 1-center location problem is written as follows:

$$\min_{x\in V} \max_{t\in[l,u]} (f(x,t) - f(x^*,t)), \tag{6}$$

where

$$f(x^*,t)=\min_{y\in V} f(y,t).$$

Therefore, we can rewrite the minmax regret vertex 1-center location problem as

$$\min_{x\in V}\max_{y\in V}\max_{t\in[l,u]}(f(x,t)-f(y,t)). \tag{7}$$

Now we wants to find a vertex such $\bar{x}$ on tree T so that it minimizes this function:

$$\max_{y\in V(T)}\max_{t\in[l,u]}(f(\bar{x},t)-f(y,t)).$$

Averbakh (2003) proved that the robust 1-center location problem with minmax regret criterion on general networks is NP-hard, if the edge lengths be as interval. Moreover, he proved that this problem can be solved on tree networks in a polynomial time (Averbakh and Berman 2000).

Lemma 4.1 *The minmax regret 1-center location problem on general networks with dynamic edge lengths is NP-hard.*

Proof The length of edges are linear functions. Therefore, the length of edge $e\in E$ is belong to interval $[l_e(l), l_e(u)]$. □

Assume that for $x\in e=(u,v)$ the ratio

$$\frac{d^t(x,v)}{l_e(t)}$$

remains constant for every $t\in[l,u]$. For every vertex x and y on tree T, we know that $f(x,t)=\max_{1\leq i\leq n} f_i(x,t)$ and $f(y,t)=\max_{1\leq i\leq n} f_i(y,t)$, where $f_i(x,t)=A_{xi}t^2+B_{xi}t+C_{xi}$ and $f_i(y,t)=A_{yi}t^2+B_{yi}t+C_{yi}$. Notice that A_{xi}, A_{yi}, B_{xi}, B_{yi} and C_{xi}, C_{yi} are defined similar to A_i, B_i and C_i introduced in Sect. 3. Now to calculate the robust solution, $f(x,t)-f(y,t)$ must be computed for each pair of vertices x and y. This is done as the following two steps.

Step 1. For every $x\in V$, compute the upper envelope of the functions

$$f_i(x,t)=A_{xi}t^2+B_{xi}t+C_{xi},\quad 1\leq i\leq n$$

using the presented method in Theorem 3.2. Notice that for each $x\in V$ and $i\in\{1,...,n\}$, $f_i(x,t)$ is an univariate, continuous and totally defined function on all points of interval $[l,u]$, also, $f_i(x,t)$ is a quadratic or linear function. Then for each pair $i,j\in\{1,...,n\}$, $i\neq j$, the graph of $f_i(x,t)$ and $f_j(x,t)$ cut each other off at most in two points. The time complexity of computing the upper envelope for a given vertex is $O(n\log n)$. Consequently, for all vertices, this time increases to $O(n^2\log n)$.

Step 2. This step computes $f(x,t)-f(y,t)$ for all $x,y\in V$. Suppose that $\{x1,x2,...,xm_x\}$ and $\left\{y1,y2,...,ym_y\right\}$ are permutations of $\{1,2,...,n\}$ such that

$$\begin{aligned} f(x,t)=&f_{x1}(x,t),\quad \forall t\in[l,t_{x1}],\\ f(x,t)=&f_{x2}(x,t),\quad \forall t\in[t_{x1},t_{x2}],\\ &\vdots \end{aligned}$$

$$f(x,t) = f_{xm_x+1}(x,t), \quad \forall t \in [t_{xm_x}, u].$$

and

$$\begin{aligned} f(y,t) =& f_{y1}(y,t), \quad \forall t \in [l, t_{y1}], \\ f(y,t) =& f_{y2}(y,t), \quad \forall t \in [t_{y1}, t_{y2}], \\ &. \\ &. \\ &. \\ f(y,t) =& f_{ym}(y,t), \quad \forall t \in [t_{ym_y}, u]. \end{aligned}$$

Notice that $m = \lambda_2(n)$. Now we want to characterize $f(x,t) - f(y,t)$ on interval $[l, u]$. Let $\bar{t}_i = \min\left\{t_{xi}, t_{yi}\right\}$, m_x be the number of breakpoints of $f(x,t)$ and m_y be the number of breakpoints of $f(y,t)$. Also, assume that $k = \min\{m_x, m_y\}$.

Lemma 4.2 *If $t \in \left[\bar{t}_0, \bar{t}_1\right]$, then $f(x,t) - f(y,t) = f_{x1}(x,t) - f_{y1}(y,t)$.*

Proof For every $t \in \left[\bar{t}_0, t_{x1}\right]$, $f(x,t) = f_{x1}(x,t)$ and for every $t \in \left[\bar{t}_0, t_{y1}\right]$, $f(y,t) = f_{y1}(y,t)$. By considering $\bar{t}_1$ is less than t_{y1}, the desired goal can be concluded. □

Lemma 4.3 *For $i = 1, ..., k-1$, we have*

1. *If $\bar{t}_i = t_{xi}$ and $t_{yi} \le \bar{t}_{i+1}$, then for every $t \in [\bar{t}_i, t_{yi}]$,*

$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi}(y,t)$$

and for every $t \in [t_{yi}, \bar{t}_{i+1}]$,

$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi+1}(y,t).$$

2. *If $\bar{t}_i = t_{xi}$ and $t_{yi} > \bar{t}_{i+1}$, then for every $t \in [\bar{t}_i, \bar{t}_{i+1}]$,*

$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi}(y,t)$$

3. *If $\bar{t}_i = t_{yi}$ and $t_{xi} \le \bar{t}_{i+1}$, then for every $t \in [\bar{t}_i, t_{xi}]$,*

$$f(x,t) - f(y,t) = f_{xi}(x,t) - f_{yi+1}(y,t)$$

and for every $t \in [t_{xi}, \bar{t}_{i+1}]$,

$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi+1}(y,t).$$

4. *If $\bar{t}_i = t_{yi}$ and $t_{xi} > \bar{t}_{i+1}$, then for every $t \in [\bar{t}_i, \bar{t}_{i+1}]$,*

$$f(x,t) - f(y,t) = f_{xi}(x,t) - f_{yi+1}(y,t).$$

Proof Case 1 can be proved as follows. For every $t \in \left[t_{xi}, t_{xi+1}\right]$, $f(x,t) = f_{xi+1}(x,t)$ and for every $t \in \left[t_{yi}, t_{yi+1}\right]$, $f(y,t) = f_{yi+1}(y,t)$. If $\bar{t}_i = t_{xi}$ and $t_{yi} \le \bar{t}_{i+1}$, then on interval $\left[\bar{t}_i, t_{yi}\right]$, $f(x,t) = f_{xi+1}(x,t)$ and $f(y,t) = f_{yi}(y,t)$. Therefore,

$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi}(y,t).$$

Also, for every $t \in \left[t_{yi}, \bar{t}_{i+1}\right]$, $f(x,t) = f_{xi+1}(x,t)$ and $f(y,t) = f_{yi+1}(y,t)$. Then

$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi+1}(y,t).$$

The other cases can be proved in an analogous way as Case 1. □

Now based on Lemma 4.2 and Lemma 4.3 we get the following algorithm.
Algorithm 1: Find $f(x,t) - f(y,t)$ for each given pairs of vertices x, y.

1. For every $t \in \left[\bar{t}_0, \bar{t}_1\right]$, set
$$f(x,t) - f(y,t) = f_{x1}(x,t) - f_{y1}(y,t).$$
2. Set $i = 1$.
3. Repeat the following steps for $1 \le i \le k$:
4. **If** $\bar{t}_i = t_{xi}$ and $t_{yi} \le \bar{t}_{i+1}$, **then** for every $t \in [\bar{t}_i, t_{yi}]$ set
$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi}(y,t),$$
and for every $t \in [t_{yi}, \bar{t}_{i+1}]$ set
$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi+1}(y,t).$$
5. **If** $\bar{t}_i = t_{xi}$ and $t_{yi} > \bar{t}_{i+1}$, **then** for every $t \in [\bar{t}_i, \bar{t}_{i+1}]$ set
$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi}(y,t),$$
6. **If** $\bar{t}_i = t_{yi}$ and $t_{xi} \le \bar{t}_{i+1}$, **then** for every $t \in [\bar{t}_i, t_{xi}]$ set
$$f(x,t) - f(y,t) = f_{xi}(x,t) - f_{yi+1}(y,t),$$
and for every $t \in [t_{xi}, \bar{t}_{i+1}]$ set
$$f(x,t) - f(y,t) = f_{xi+1}(x,t) - f_{yi+1}(y,t).$$
7. **If** $\bar{t}_i = t_{yi}$ and $t_{xi} > \bar{t}_{i+1}$, **then** for every $t \in [\bar{t}_i, \bar{t}_{i+1}]$ set
$$f(x,t) - f(y,t) = f_{xi}(x,t) - f_{yi+1}(y,t),$$
8. Set $i = i + 1$ and go to 3.

These steps are repeated until the function $f(x,t) - f(y,t)$ are specified over the entire interval $[l, u]$. In an iteration of Algorithm 1, the total time complexity is a constant value. Thus, we can conclude the following lemma.

Lemma 4.4 *For the given vertices x and y on tree T, the function $f(x,t) - f(y,t)$ on interval $[l, u]$ is computed in a linear time. Moreover, the maximum value of this function occurs at one of the breakpoints.*

Proof The functions $f(x,t)$ and $f(y,t)$ are piecewise linear or quadratic functions, where are computed in Step 1. Therefore $f(x,t) - f(y,t)$ is also a piecewise linear or quadratic function. According to Theorem 3.3 maximum and minimum point of this function is obtained in its breakpoints. Therefore,

$$\begin{aligned}
&\max_{t\in[l,u]} (f(x,t) - f(y,t)) \\
&\quad = \max_{1\le i\le m} \big\{ f_{xi-1}(x, t_{xi-1}) - f_{yi}(y, t_{xi-1}),\ f_{xi}(x, t_{xi-1}) \\
&\qquad - f_{yi-1}(y, t_{xi-1}),\ f_{xi}(x, t_{xi-1}) - f_{yi}(y, t_{xi-1}),\ f_{xi-1}(x, t_{xi-1}) \\
&\qquad - f_{yi-1}(y, t_{xi-1}),\ f_{xi-1}(x, t_{xi}) - f_{yi}(y, t_{xi}),\ f_{xi}(x, t_{xi}) - f_{yi-1}(y, t_{xi}),\ f_{xi}(x, t_{xi}) \\
&\qquad - f_{yi}(y, t_{xi}),\ f_{xi-1}(x, t_{xi}) - f_{yi-1}(y, t_{xi}),\ f_{xi-1}(x, t_{yi-1}) \\
&\qquad - f_{yi}(y, t_{yi-1}),\ f_{xi}(x, t_{yi-1}) - f_{yi-1}(y, t_{yi-1}),\ f_{xi}(x, t_{yi-1})
\end{aligned}$$

$$
\begin{aligned}
&- f_{yi}(y, t_{yi-1}), f_{xi-1}(x, t_{yi-1}) - f_{yi-1}(y, t_{yi-1}), f_{xi-1}(x, t_{yi})\\
&- f_{yi}(y, t_{yi}), f_{xi}(x, t_{yi}) - f_{yi-1}(y, t_{yi}), f_{xi}(x, t_{yi}) - f_{yi}(y, t_{yi}), f_{xi-1}(x, t_{yi})\\
&- f_{yi-1}(y, t_{yi}), f_{xi-1}\left(x, \frac{-B_{xi-1} - B_{yi}}{2A_{xi-1} + 2A_{yi}}\right)\\
&- f_{yi}\left(y, \frac{-B_{xi-1} - B_{yi}}{2A_{xi-1} + 2A_{yi}}\right), f_{xi}\left(x, \frac{-B_{xi} - B_{yi-1}}{2A_{xi} + 2A_{yi-1}}\right)\\
&- f_{yi-1}\left(y, \frac{-B_{xi} - B_{yi-1}}{2A_{xi} + 2A_{yi-1}}\right), f_{xi}\left(x, \frac{-B_{xi} - B_{yi}}{2A_{xi} + 2A_{yi}}\right)\\
&- f_{yi}\left(y, \frac{-B_{xi} - B_{yi}}{2A_{xi} + 2A_{yi}}\right), f_{xi-1}\left(x, \frac{-B_{xi-1} - B_{yi-1}}{2A_{xi-1} + 2A_{yi-1}}\right)\\
&-f_{yi-1}\left(y, \frac{-B_{xi-1} - B_{yi-1}}{2A_{xi-1} + 2A_{yi-1}}\right)\Bigg\}.
\end{aligned}
$$

□

Thus, we get the following Lemma.

Theorem 4.5 *An optimal solution of the dynamic robust vertex 1-center location problem under minmax regret criterion is obtained in $O(n^3)$ time.*

Proof Step 1 is performed in $O(n^2 \log n)$ time. Step 2 is done in a linear time for every vertices x and y on tree T. Then the total time complexity of Step 2 and consequently, the total time complexity of Algorithm 1 is $o(n^3)$. □

Example 4.6 Consider the given dynamic tree network of Example 3.6. We want to find a vertex that minimizes the robust 1-center location problem with minmax regret criterion. We first compute $f(x,t) - f(y,t)$ for every pair of vertices x and y. The computed values of $f(x,t) - f(y,t)$ is written in Table 3. The procedure of finding $f(x,t) - f(y,t)$ with $x = v_1$ and $y = v_2$ is as follows. Let $x = v_1$ and $y = v_2$. Therefore, $m_x = 2$ and $m_y = 3$. Also, $l = m_x = 2$, $\bar{t}_0 = -1$ and $\bar{t}_1 = \frac{-15+\sqrt{477}}{18}$. For every $t \in \left[\bar{t}_0, \bar{t}_1\right]$, $f(x,t) - f(y,t) = 12t + 20 - (3t + 12) = 9t + 8$. For every $t \in \left[\bar{t}_1, \max\{2, \frac{-15+\sqrt{477}}{18}\}\right]$, $f(x,t) - f(y,t) = 12t + 20 - (9t^2 + 18t + 5) = 15 - 9t^2 - 6t$ since $\max\left\{2, \frac{-15+\sqrt{477}}{18}\right\} = 2 = t_{x1}$. For other pair of vertices, the maps are determined in an analogous way. Now we use Lemma 4.4 and obtain an optimal solution for the robust vertex 1-center location problem. Thus, we have

$$\min_{x\in\{v_1,\ldots,v_6\}} \max_{x\in\{v_1,\ldots,v_6\}} (f(x,t) - f(y,t)) = \min\{16, 84, 64, 104, 50, 184\} = 16$$

Therefore, v_1 is a robust solution of the robust 1-center location problem with minmax regret criterion. The optimal value of objective function is equal to 16.

Corollary 4.7 *If $A_{xi} = 0$ for all $i = 1, \ldots, n$ and $x \in V$, then the time complexity of the dynamic robust vertex 1-center location problem under minmax regret criterion is reduced to $O(n^2\alpha(n)\log n)$.*

5 Dynamic robust absolute 1-center location problem under minmax criterion

If we use the worst case criterion for the absolute 1-center location problem, then we must find a point on T such that this point minimizes the maximum value of the objective function

Table 3 The values of $f(x,t) - f(y,t)$

x, y	$f(x,t) - f(y,t)$
v_1, v_2	$9t + 8, \quad t \in \left[-1, \frac{-15+\sqrt{477}}{18}\right]$
	$15 - 9t^2 - 6t, \quad t \in \left[\frac{-15+\sqrt{477}}{18}, 3\right]$
v_1, v_3	$12t + 12\,, \quad t \in \left[1, \frac{1}{3}\right]$
	$17 - 9t^2, \quad t \in \left[\frac{1}{3}, 3\right]$
v_1, v_4	$8t + 8, \quad t \in \left[-1, \frac{-3+\sqrt{33}}{6}\right]$
	$16 - 12t^2 - 4t, \quad t \in \left[\frac{-3+\sqrt{33}}{6}, 3\right]$
v_1, v_5	$8t + 4, \quad t \in \left[-1, \frac{-19+\sqrt{753}}{24}\right]$
	$15 - 7t - 18t^2, \quad t \in \left[\frac{-19+\sqrt{753}}{24}, 3\right]$
v_1, v_6	$0\,, \quad t \in \left[-1, \frac{-12+\sqrt{1152}}{32}\right]$
	$14 - 18t^2 - 12t, \quad t \in \left[\frac{-12+\sqrt{1152}}{32}, 3\right]$
v_2, v_3	$3t + 4, \quad t \in \left[-1, \frac{1}{3}\right]$
	$9 - 9t - 9t^2, \quad t \in \left[\frac{1}{3}, \frac{-15+\sqrt{477}}{18}\right]$
	$6t + 2, \quad t \in \left[\frac{-15+\sqrt{477}}{18}, 3\right]$
v_2, v_4	$-t, \quad t \in \left[-1, \frac{-15+\sqrt{477}}{18}\right]$
	$9t^2 + 14t - 7, \quad t \in \left[\frac{-15+\sqrt{477}}{18}, \frac{-3+\sqrt{33}}{6}\right]$
	$-3t^2 + 2t + 1, \quad t \in \left[\frac{-3+\sqrt{33}}{6}, 3\right]$
v_2, v_5	$-t - 4, \quad t \in \left[-1, \frac{-19+\sqrt{753}}{24}\right]$
	$7 - 16t - 12t^2, \quad t \in \left[\frac{-19+\sqrt{753}}{24}, \frac{-15+\sqrt{477}}{18}\right]$
	$-3t^2 - t, \quad t \in \left[\frac{-15+\sqrt{477}}{18}, 3\right]$
v_2, v_6	$-9t - 8, \quad t \in \left[-1, \frac{-12+\sqrt{1152}}{32}\right]$
	$6 - 12t - 18t^2, \quad t \in \left[\frac{-12+\sqrt{1152}}{32}, \frac{-15+\sqrt{477}}{18}\right]$
	$-9t^2 - 6t - 1, \quad t \in \left[\frac{-15+\sqrt{477}}{18}, 3\right]$
v_3, v_4	$-4t - 4, \quad t \in \left[-1, \frac{1}{3}\right]$
	$9t^2 + 8t - 9, \quad t \in \left[\frac{1}{3}, \frac{-3+\sqrt{33}}{6}\right]$
	$-3t^2 - 4t - 1, \quad t \in \left[\frac{-3+\sqrt{33}}{6}, 3\right]$
v_3, v_5	$-4t - 8, \quad t \in \left[-1, \frac{1}{3}\right]$
	$9t^2 + 8t - 13, \quad t \in \left[\frac{1}{3}, \frac{-12+\sqrt{753}}{24}\right]$
	$-3t^2 - 7t - 2, \quad t \in \left[\frac{-12+\sqrt{753}}{24}, 3\right]$

Table 3 continued

x, y	$f(x,t) - f(y,t)$
v_3, v_6	$-12t - 12, \quad t \in \left[-1, \frac{-12+\sqrt{1152}}{32}\right]$
	$2 - 24t - 18t^2, \quad t \in \left[\frac{-12+\sqrt{1152}}{32}, \frac{1}{3}\right]$
	$-9t^2 - 12t - 3, \quad t \in \left[\frac{1}{3}, 3\right]$
v_4, v_5	$-4\,, \quad t \in \left[-1, \frac{-19+\sqrt{753}}{24}\right]$
	$7 - 15t - 12t^2, \quad t \in \left[\frac{-19+\sqrt{753}}{24}, \frac{-3+\sqrt{33}}{6}\right]$
	$-3t - 1, \quad t \in \left[\frac{-3+\sqrt{33}}{6}, 3\right]$
v_4, v_6	$-8t - 8, \quad t \in \left[-1, \frac{-12+\sqrt{1152}}{32}\right]$
	$6 - 20t - 18t^2, \quad t \in \left[\frac{-12+\sqrt{1152}}{32}, \frac{-3+\sqrt{33}}{6}\right]$
	$-6t^2 - 8t - 2, \quad t \in \left[\frac{-3+\sqrt{33}}{6}, 3\right]$
v_5, v_6	$-8t - 4, \quad t \in \left[-1, \frac{-12+\sqrt{1152}}{32}\right]$
	$10 - 20t - 18t^2, \quad t \in \left[\frac{-12+\sqrt{1152}}{32}, \frac{-19+\sqrt{753}}{24}\right]$
	$-6t^2 - 5t - 1, \quad t \in \left[\frac{-19+\sqrt{753}}{24}, 3\right]$

on interval $[l, u]$. To do this, first we can find a point on each edge of the tree T that minimizes the worst case criterion. These points are called the local minimums and their number are equal to number of edges of T (i.e., $n - 1$). Then among these points we select a point as an optimal solution for the robust absolute 1-center location problem under minmax criterion. This solution is called the global minimum. This criterion on edge $e \in E$ is considered as follows:

$$\min_{x \in e} \max_{t \in [l,u]} f(x,t), \tag{8}$$

where

$$f(x,t) = \max_{i=1,\ldots,n} (A_i t^2 + B_i t + C_i).$$

It is clear that there is an one-to-one correspondence between the length of edge $e = (u, v) \in E$ and the interval $[0, 1]$, where 0 is corresponding to $d^t(u, u)$ and 1 is corresponding to $d^t(u, v)$, for all $t \in [l, u]$. Therefore, if we assume that $x \in [0, 1]$, then A_i, B_i and C_i for $i = 1, \ldots, n$ are as linear functions of x.
Then $f(x, t)$ is an upper envelope of n continuous, totally defined and bivariate functions. Consequently, $f(x, t)$ is a piecewise bivariate function where each piece of this function is a plane or a parabola. Let the number of pieces of this function be m and define

$$f_i(x,t) = A_i t^2 + B_i t + C_i, \quad \forall i = 1, \ldots, n.$$

We want to compute $f(x, t)$ on intervals $[l, u]$ and $[0, 1]$ in which $t \in [l, u]$ and $x \in [0, 1]$. To calculate this function, we first examine its properties.

Lemma 5.1 *1. At one point of the domain, the function with maximum value is equal to upper envelope in this point.*

2. *If (x_k, t_k) is the starting point of domain for function $f_k(x,t)$, then the domain of this function is $\left[x_k, x_{k+1}\right] \times \left[t_k, t_{k+1}\right]$, where (x_{k+1}, t_{k+1}) is the smallest intersection point of this function with other functions.*

Proof 1. It is obvious.

2. Let (x_{k+1}, t_{k+1}) and (x_{k+2}, t_{k+2}) be the intersection points of function $f_k(x,t)$ with functions $f_i(x,t)$ and $f_j(x,t)$. Also, $x_{k+1} < x_{k+2}$ and $t_{k+1} < t_{k+2}$.

$$f_k(x,t) > f_i(x,t), \quad f_k(x,t) > f_j(x,t) \quad \forall (x,t) \in \left[x_k, x_{k+1}\right] \times \left[t_k, t_{k+1}\right]$$

and

$$f_k(x_{k+1}, t_{k+1}) = f_i(x_{k+1}, t_{k+1}).$$

Given that $f_k(x,t)$ is decreasing and $f_i(x,t)$ is increasing in this interval, thus

$$f_k(x,t) < f_i(x,t), \quad \forall x > x_{k+1} andt > t_{k+1}.$$

□

For obtaining function $f(x,t)$, we perform the following algorithm:
Algorithm 2: Find $f(x,t)$ on intervals $[l, u]$ and $[0, 1]$

1. Set $k = 0$, $k' = k + 1$, $t_0 = l$ and $x_0 = 0$.
2. Calculate the value of functions $f_i(x,t) = A_i t^2 + B_i t + C_i$ for $i = 1, \ldots, n$ at point t_k, x_k.
3. Select the function with maximum value in specified point and denote the selected function by $f_{i_k}(x,t)$.
4. Let $\left(x_j, t_j\right)$ be the intersection of $f_{i_k}(x,t)$ with $f_j(x,t)$ for $2 \leq j \leq n$ such that $x_{k+1} \leq x_j$ and $t_{k+1} \leq t_j$ for $1 \leq j \leq n$. Show the domain of $f_{i_k}(x,t)$ with $P_k = I_k \times J_k$, where $I_k = \left[t_k, t_{k+1}\right]$ and $J_k = \left[x_k, x_{k+1}\right]$.
5. If (x_{k+1}, t_{k+1}), $x_{k+1} < 1$ and $t_{k+1} < u$, is the intersection of $f_{i_k}(x,t)$ with the one of functions $f_j(x,t)$, then call it $f_{i_{k+1}}(x,t)$. Set $k = k + 1$ and go to step 2.
6. Otherwise, call the first one $f_{i_{k+1}}(x,t)$ and the second function $f_{i_{k'+1}}(x,t)$. Set $k = k+1$ and repeat the steps 2-6 for $f_{i_{k+1}}(x,t)$ and $f_{i_{k'+1}}(x,t)$ separately. In this case, $J_{k'+1} = J_k$, $I_{k'+1} = \left[t_k, t_{k+1}\right]$ and $J_{k+1} = \left[x_k, x_{k+1}\right]$, $I_{k+1} = I_k$, respectively.

This procedure performs at most n iteration and any time contains $O(n)$ operations. Therefore, the total time for performing the characterized procedure is $O(n^2)$.

Lemma 5.2 *An optimal solution of the absolute robust absolute 1-center location problem on T is obtained in $O(n^3)$ time.*

Proof For each edge of tree T run Algorithm 2. Thus the time complexity of problem is equal to $O(n^3)$. □

6 Dynamic robust absolute 1-center location problem with minmax regret criterion

Consider now the robust deviation absolute 1-center location problem in which the length of edges and the weight of vertices are linear functions of parameter t. This problem is written as follows:

$$\min_{x \in T} \max_{t \in [l,u]} (f(x,t) - f(x^*,t)). \tag{9}$$

According to Lemma 2.2, we have

$$f(x^*, t) = \max_{i,j=1,\ldots,n} \frac{w_i(t)w_j(t)d^t(v_i, v_j)}{w_i(t) + w_j(t)}.$$

Similar to Sect. 5, we consider this problem on each edge $e \in E$ and solve this problem on described edge. Let

$$f(x^*, t) = \max_{i,j=1,\ldots,n} \frac{\alpha_{ij}t^3 + \beta_{ij}t^2 + \gamma_{ij}t + \delta_{ij}}{(a_i + a_j)t + (b_i + b_j)},$$

where

$$\alpha_{ij} = a_i a_j \left(\sum_{e\in P_{ij}} a_e\right),$$

$$\beta_{ij} = \left(a_i a_j \left(\sum_{e\in P_{ij}} b_e\right) + a_i b_j \left(\sum_{e\in P_{ij}} a_e\right) + b_i a_j \left(\sum_{e\in P_{ij}} a_e\right)\right),$$

$$\gamma_{ij} = \left(a_i b_j \left(\sum_{e\in P_{ij}} b_e\right) + b_i a_j \left(\sum_{e\in P_{ij}} b_e\right) + b_i b_j \left(\sum_{e\in P_{ij}} a_e\right)\right)$$

and

$$\delta_{ij} = b_i b_j \left(\sum_{e\in P_{ij}} b_e\right).$$

Notice that P_{ij} represents the unique path between the vertices v_i and v_j on tree T. Therefore, $f(x^*, t)$ is the maximum of $O(n^2)$ univariate fractional functions. for solving problem (9), first we must compute the upper envelope of $O(n^2)$ univariate fractional functions on interval $[l, u]$. To compute the upper envelope we distinguish the following two cases:

Case 1. Let the denominator of functions

$$f_{ij}(t) = \frac{\alpha_{ij}t^3 + \beta_{ij}t^2 + \gamma_{ij}t + \delta_{ij}}{(a_i + a_j)t + b_i + b_j}$$

be a scaler, for all $1 \le i, j \le n$, i.e,

$$a_i + a_j = 0.$$

In this case, $f(x^*, t)$ is the upper envelope of $O(n^2)$ continuous, totally defined, univariate functions such that the maximum degree of each of them is three. Furthermore, each disjoint pair $f_{ij}(t)$ and $f_{kh}(t)$ intersect in at most three points on interval $[l, u]$, $1 \le i, j, k, h \le n$. (We know that each polynomial with degree n has at most n real roots). Therefore, we can use Theorem 3.3 and calculate $f(x^*, t)$ on interval $[l, u]$. Thus, we conclude the following Lemma.

Lemma 6.1 *The upper envelope of* $\{f_{ij} : 1 \le i, j \le n\}$ *is obtained in* $O(n^2\alpha(n^2)\log n)$ *time. It has at most* $m = O(n^2\alpha(n^2))$ *subinterval.*

Proof In Lemma 3.1 and Theorem 3.3, if we set $s = 3$ and n^2 instead n, we can conclude this Lemma. □

Now we have to solve the dynamic robust absolute 1-center location problem on each obtained subinterval.
It can be solved separately on every edge of tree in $O(n^2)$ time using Algorithm 2. Therefore, the number of local minimums on tree T and interval $[l, u]$ is at most $O(n^3\alpha(n^2))$. Consequently, we can select an optimal solution for the dynamic robust absolute 1-center location problem on interval $[l, u]$ between the obtained optimal solutions.

Lemma 6.2 *Time complexity of computing an optimal solution for the dynamic robust absolute 1-center location problem under minmax regret criterion in the case of $a_i + a_j = 0$ is $O(n^5\alpha(n^2))$.*

Case 2. Let $a_i + a_j \neq 0$, for all i, j. In this case we use Algorithm 2 two times. In the first time we use this algorithm for computing $\max_{i,j=1,\dots,n} f_{ij}(t)$ on interval $[l, u]$. In the following we present a brief description of how to calculate $\max_{i,j=1,\dots,n} f_{ij}(t)$ using Algorithm 2 on interval $[l, u]$. It should be noted that Algorithm 2 is used for bivariate functions, but we want to use it for univariate functions.

1. Set $t_0 = l$ and compute the value of functions $f_{ij}(t)$ at point t_0.
2. Select the function that has maximum value at this point. Denote this function with $f_{ij_1}(t)$.
3. Compute the intersection of $f_{ij_1}(t)$ with the remaining functions.
4. Choose the minimum intersection point and consider this point with t_1. Denote the function corresponding to this point by f_{ij_2}.
5. Repeat steps 1 to 4 until $t_k \geq u$.

If this approach is repeated, then $f(x^*, t)$ is computed. In this case, at most $O(n^2)$ pieces and their corresponding intervals are obtained in $O(n^4)$ time.
Again, the dynamic robust absolute 1-center location problem on each edge $e \in E$ and each interval is computed, separately using the Algorithm 2. On each interval and each edge the dynamic robust absolute 1-center location problem under minmax regret criterion can be calculated in $O(n^2)$ time. Then we can conclude the following Lemma.

Lemma 6.3 *Dynamic robust absolute 1-center location problem under minmax regret criterion on T in case $a_i + a_j \neq 0$ can be solved in $O(n^5)$ time.*

If we compare case 1 and case 2 together, we see that the used method used in the second case has less time complexity. Therefore, the second case method is more efficient.

Remark 6.4 By the same process, we can obtain an optimal solution for relative robust absolute(vertex) 1-center location problem on tree T.

7 Robust deviation 1-center location problem on trees

In this section, we assume that T is a weighted tree and the edge lengths and vertex weights of T correspond to a finite set of positive real and real numbers, respectively. Denote the set of all possible events for the edge lengths and vertex weights of tree T with a finite number of scenarios, which is denoted by S. We now consider the robust deviation 1-center location problem. For obtaining the optimal solution of this problem, first we have to compute the 1-centers on the tree T for all scenarios. Using Lemma 2.1, we get the following Lemma:

Lemma 7.1 *Under all scenarios in S, the time complexity of computing $f(x^*, s)$ is equal to $O(n \log n|S|)$.*

For solving the minmax regret 1-center location problem with the candidate locations E, it is sufficient to solve this problem on each edge of tree T, separately. Therefore, we first compute the optimal solution of this problem for every edge $e \in E$.

$$\min_{x\in e}(\min_{x\in V})\max_{s\in S} R(x,s). \tag{10}$$

We can obtain the optimal solution of this problem at $O(|S|)$ time using the Kouvelis's algorithm (Kouvelis et al. 1993). Then we have

Lemma 7.2 *The robust deviation 1-center location problem on tree networks can be solved in $O(|S|n\log n)$ time.*

Suppose now that the weights of all vertices of tree T are equal to one. Also, for each $s \in S$, let $p[a_s, b_s]$ be a longest path on T under scenario s which is computed in linear time by Handler's algorithm (Handler 1973). On the other hand, we know that the middle point of this path indicates the absolute 1-center under scenario $s \in S$. Then, for all $x \in T$ we have

$$R(x,s) = \max_{i=1,\ldots,n} d^s(v_i,x) - \frac{d^s(a_s,b_s)}{2}.$$

Furthermore, we know that the furthest vertex from each point x on T is the endpoint of longest path on T. Therefore, $\max_{i=1,\ldots,n} d^s(v_i,x) = \max\{d^s(a_s,x), d^s(b_s,x)\}$. It follows that

$$R(x,s) = \max\left\{d^s(a_s,x) - \frac{d^s(a_s,b_s)}{2}, d^s(b_s,x) - \frac{d^s(a_s,b_s)}{2}\right\}.$$

Now we solve the minmax regret 1-center location problem on each edge $e = (u,v) \in E$. If we define a mapping on edge $e = (u,v)$ such that in this map the points u and v correspond to 0 and 1, respectively, and the distance between u and each point x on edge e except u and v is equal to $xd^s(u,v)$, where $x = \frac{d^s(u,x)}{d^s(u,v)}$, then we conclude that

$$d^s(a_s,x) - \frac{d^s(a_s,b_s)}{2} = \frac{2d^s(a_s,u) - d^s(a_s,b_s)}{2} + xd^s(u,v).$$

It is clear that $d^s(a_s,x) - \frac{d^s(a_s,b_s)}{2}$ and $d^s(b_s,x) - \frac{d^s(a_s,b_s)}{2}$ are the linear functions which are defined on interval $[0,1]$. Therefore, an optimal solution for the robust deviation 1-center location problem on edge e can be computed in $O(|S|)$ time using the presented algorithm in Kouvelis et al. (1993).

Lemma 7.3 *An optimal solution for the robust deviation 1-center location problem on the un-weighted tree T is obtained in $O(n|S|)$ time.*

8 Conclusion

In this paper, we considered the robust vertex (absolute) 1-center location problem on tree networks in which vertex weights and edge lengths have been considered as dynamic data or as a discrete set of scenarios. We considered the problems with minmax and minmax regret criteria and developed combinatorial algorithms which solve the problems under investigation in polynomial times.

We showed that in all problems, the optimal solution was located in one of the breakpoints. The dynamic robust vertex 1-center problem under minmax and minmax regret criteria were solved in times $O(n^2 \log n)$ and $O(n^3)$. Also, an optimal solution for the dynamic robust absolute 1-center problem under minmax and minmax regret criteria were computed in times $O(n^3)$ and $O(n^5)$. In the two first problems, the objective functions are univariable, but in the next two problems, the objective functions are bivariables. Finally, we considered the robust deviation 1-center problem and obtained an algorithm for this problem in $O(|S|n \log n)$ time complexity. In this problem, the vertex weights and edge lengths were considered as discrete set of scenarios.

For further research, we can consider the robust inverse (reverse) location problems. We can investigate the robust inverse 1-center problem. Furthermore, we can also study the minmax reverse 1-center problem.

Declarations

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

Human Participants This article does not contain any studies with human participants or animals performed by any of the authors.

References

Ackermann W (1928) Zum Hilbertschen Aufbau der reellen zahlen. Math Ann 99(1):118–133

Agarwal PK, Scharzkopf O, Sharir M (1996) The overlay of lower envelopes and its applications. Discrete Comput Geom 15(1):1–13

Aloulou MA, Kalai R, Vanderpooten D (2005) Minmax regret 1-center problem on a network with a discrete set of scenarios. Cahiers de Recherche en Ligne du LAMSADE-Document (132)

Averbakh I (2003) Complexity of robust single facility location problems on networks with uncertain edge lengths. Discrete Appl Math 127(3):505–522

Averbakh I, Berman O (1997) Minmax regret p-center location on a network with demand uncertainty. Location Sci 5(4):247–254

Averbakh I, Berman O (2000) Algorithms for the robust 1-center problem on a tree. Eur J Oper Res 123(2):92–302

Bhattacharya B, Kameda T, Song Z (2012) Minmax regret 1-center on a path/cycle/tree. In 6th Int'l Conf. on Advanced Engineering Computing and Applications in Sciences (ADVCOMP), 108–113

Bhattacharya B, Kameda T, Song Z (2014) Improved minmax regret 1-center algorithms for cactus networks with c cycles. Latin American Symposium on Theoretical Informatics. Springer, Berlin, pp 330–341

Bhattacharya B, Kameda T, Song Z (2015) Minmax regret 1-center algorithms for path/tree/unicycle/cactus networks. Discrete Appl Math 195:18–30

Burkard RE, Dollani H (2002) A note on the robust 1-center problem on trees. Ann Oper Res 110(1–4):69–82

Dearing PM, Francis RL (1974) A minmax location problems on networks. Transport Sci 8(4):333–343

Handler GY (1973) Minmax location of a facility in an undirected tree graph. Transport Sci 7(3):287–293

Hakimi SL (1965) Optimal distribution of switching centers in a communications and some related graph-theoric problems. Oper Res 13(3):462–475

Kariv O, Hakimi SL (1979) An algorithmic approach to network location problems. I. The p-centers. SIAM J Appl Math 37(3):513–538

Kouvelis P, Vairaktarakis GL, Yu G (1993) Robust 1-median location on a tree in the presence of demand and transportation cost uncertainty. Department of Industrial and Systems Engineering, University of Florida, Gainesville

Lin TC, Yu HI, Wang BF (2006) Improved algorithms for the minmax regret 1-center and $1-$median problem. International Symposium on Algorithms and Computation, Springer, Berlin, pp 537–546

Megiddo N (1983) Linear-time algorithms for linear programming in $\mathcal{R}^3$ and related problems. SIAM J Comput 12(4):759–776

Megiddo N, Tamir A (1983) New results on the complexity of p-center problems. SIAM J Comput 12(4):751–758