




# Solving Maximin Location Problems on Networks with Different Metrics and Restrictions

Gennady G. Zabudsky<sup>(✉)</sup> 

Sobolev Institute of Mathematics SB RAS, Novosibirsk 630090, Russia  
zabudsky@ofim.oscsbras.ru

**Abstract.** Several optimal location problems of an obnoxious facility on a network of roads connecting settlements are considered. It is necessary to find such location of the facility so that a minimum distance to a nearest settlement is as large as possible taking into account the resident population. Such facility can be, for example, a nuclear power plant, a waste recycling plant. An overview of various formulations, the properties of the problems and algorithms for solving are given. The main focus is on the problem taking into account a restriction on transportation costs for servicing the settlements by the facility. The cost of servicing the settlements by the facility is determined using the shortest paths in the network. The objective function uses Euclidean metric. Exact algorithm for solving of this problem is proposed.

**Keywords:** Euclidean metric · Maximin criterion · Obnoxious facility · Shortest paths · Voronoi diagram

## 1 Introduction

One of the actively developing areas of operations research is the analysis and solving of facilities location problems. Such problems have great applied importance. They need to be solved in various fields of activity: location of service points, technological equipment in workshops, automated design of electronic devices [1–3].

In general, a facilities location problem is formulated as follows: there is an area with facilities fixed in it and new facilities that need to be located in the area. Specified restrictions on location of new facilities are to be met, and some criterion of a quality of location is to be optimal. The various formulations of such problems are defined by sizes of the facilities, areas in which they should be located (line, plane, network), various restrictions and types of criterion and so on [4–6].

The criterion of optimality in the location problems can be different, and it is depending on the specifics of the facilities and on what functions they

---

The research was funded in accordance with the state task of the IM SB RAS, project FWNF-2022-0020.

perform. A term “facility” can be interpreted quite widely. Minimization of a maximum weighted distance between facilities often is used when considering an optimal location of hospitals, police stations, fire stations. Minimization a weighted sum of distances is used when choosing a location of switches in a telephone network, warehouses, substations in a power grid [7–9]. In addition to these criteria, a maximin and maximum criteria are applied according to which a minimum distance between facilities or a minimum weighted sum of distances is maximized, respectively. These criteria are applied when dealing with obnoxious (undesirable) production facilities that have adverse effects to people or environment. Note that the adverse effects of such facilities decreases with increasing distance to them. Therefore it is easiest to locate them as far away as possible from populated areas.

Most often the maximin criterion in the location problems on networks is applied in two variants. In the first variant, distances are measured by the shortest paths in the networks. This metric is used in significant part of the location problems on networks with the maximin criterion [7, 11–13]. In the second variant, several metrics are used. For example, Euclidean metric is used to measure a negative effect of the obnoxious facility, and the shortest paths metric is used to calculate the cost of servicing customers by the facility [6, 14].

In this paper, the location problems on various networks with a maximin criterion are considered. An overview of the formulations and algorithms for solving of the problems in which the distances in the objective function are measured along the shortest paths in the network is given. The main attention is paid to the problem of optimal location of an obnoxious facility on a transport network connecting some settlements. It is necessary to find such a location of the facility so that a minimum distance to a nearest settlement is as large as possible and at the same time a budget for transportation costs for servicing the settlements by the facility was not violated. The problem uses Euclidean metric to determine the magnitude of the adverse effects of the obnoxious facility to the settlements. The shortest paths in network are used to determine the transportation costs for servicing the settlements by the facility. A polynomial algorithm for exact solving of this problem is proposed.

Section 2 provides an overview of maximin location problems on arbitrary and special networks when distances are measured by shortest paths.

In Sect. 3, a formulation of a maximin problem with different metrics is given. Euclidean metric is used in the objective function of the problem. Shortest paths metric is used in a restriction on transportation costs. Several properties of the problem and a polynomial algorithm for solving this problem are presented.

## 2 Shortest Paths Metric

The Section deals with a location problem of a facility on arbitrary and special networks with the maximin criterion. The distances between the vertices of the network and between the vertices and points on the edges are determined using the shortest paths [7, 11–13]. The maximin location problem is *NP*-hard for an

arbitrary number of facilities, even if the network has one edge [15]. Problem for one facility on general network is polynomially solvable. An overview of some properties of the problem and algorithms for its solving are given.

### 2.1 General Network

To formulate a mathematical model of the problem, we introduce the following notations [11, 12]. Let  $G = (V, E)$  be a connected, undirected network with a set of vertices  $V = \{v_1, v_2, \dots, v_n\}$  and a set of edges  $(v_i, v_j) \in E, v_i, v_j \in V, i < j, i, j \in N = \{1, \dots, n\}$ . Each edge connecting  $v_i$  and  $v_j$  has positive weight (length)  $c(v_i, v_j)$ . Denote by  $d(v_i, v_j)$  the length of a shortest path between vertices  $v_i$  and  $v_j$ . For any  $(v_i, v_j) \in E, c(v_i, v_j) \geq d(v_i, v_j)$ . We also denote by  $\alpha_i$  a parameter (weight) of the vertex  $v_i, i \in N$ , which is always positive.

It is necessary to locate a facility on the network. Therefore, the set of candidate points is the set  $Z$  consisting of the vertices and the infinite set of points on the edges. The location of a point on an edge is determined using the distances from the vertices of the edge. For example,  $z$  is located on the edge  $(v_i, v_j)$  at a distance of  $c(v_i, z) = \lambda c(v_i, v_j)$  from the vertex  $v_i$  and at a distance  $c(v_j, z) = (1 - \lambda)c(v_i, v_j)$  from the vertex  $v_j$ , where  $0 \leq \lambda \leq 1$ . Denote by  $d(v_i, z)$  the length of a shortest path from vertex  $v_i$  to  $z, z \in Z$ . Our objective will be to maximize

$$r(z) = \min_{i \in N} \alpha_i d(v_i, z), \tag{1}$$

where  $z \in Z$ .

If a point  $z^*$  solves the problem (1), then it is said to be a maximin location with the optimal value of the objective function  $r(z^*)$ .

For network distance function  $d(v_i, z), z \in (v_p, v_q), (v_p, v_q) \in E$ , the following properties hold [8, 12]:

$$d(v_i, z) = \min\{d(v_i, v_p) + c(v_i, z), d(v_i, v_q) + c(v_p, v_q) - c(v_i, z)\}$$

where  $i = 1, \dots, n$ , and  $0 \leq c(v_i, z) \leq c(v_p, v_q)$ .

Fuction  $d(v_i, z)$  is continuous and concave on segment  $[0, c(v_p, v_q)]$  and one of the conditions is met

- (a) linearly increases with slope 1 in the edge;
- (b) linearly decreases with a slope -1 on the edge;
- (c) linearly increases with slope 1 on the segment  $[0, z_i(p, q)]$  and linearly decreases with slope -1 on the segment  $[z_i(p, q), c(v_p, v_q)]$ , where

$$z_i(p, q) = (d(v_i, v_q) + c(v_p, v_q) - d(v_i, v_p))/2.$$

In [12], the problem (1) on general network was investigated. The paper presents certain properties of the problem, which allow to find a solution to the problem. Although the problem is non-convex its solution space  $G$  can be divided into edges and resulting subproblems can be solved more easily than the original problem.

In [12], it is shown that the problem (1) on the edge  $(v_p, v_q)$  is equivalent to the following linear programming problem:

$$\begin{aligned} & \max g(z) \\ g(z) = & \min_{1 \leq i \leq 2n, 0 \leq c(v_p, z) \leq c(v_p, v_q)} \{B_i z + C_i\}, \end{aligned}$$

where  $B_i = \alpha_i$ ,  $B_{n+i} = -\alpha_i$ ,  $i = 1, \dots, n$  and  $C_i = \alpha_i d(v_i, v_p)$ ,  $C_{n+i} = \alpha_i(d(v_i, v_q) + c(v_p, v_q))$ ,  $i \in N$ .

Thus, for any edge  $(v_p, v_q) \in E$ , the function  $g(z)$ ,  $z \in (v_p, v_q)$  is continuous, piecewise linear and concave on the segment  $[0, c(v_p, v_q)]$ , consisting of at most  $2n$  strictly monotone segments.

In [12], it is proved that there is any unique maximin location on each edge. As a consequence, there are no more than  $m$  local maximums on the network. A combinatorial algorithm for solving of the problem (1) with a time complexity  $O(mn)$  is proposed.

In [7], a linear programming model to solve the problem (1) was used. The problem is solved for each edge and the maximum value among the set of values of the objective function is selected. Let the obnoxious facility be placed on the edge  $(v_p, v_q) \in E$ . The shortest path from the facility to the vertex  $v_i$  is  $\min\{d(v_p, v_i) + c(v_p, z); d(v_q, v_i) + c(v_p, v_q) - c(v_p, z)\}$ . Then the model has the following form:

$$\begin{aligned} & y \rightarrow \max, \\ & \alpha_i(d(v_p, v_i) + c(v_p, z)) \geq y, \\ & \alpha_i(d(v_q, v_i) + c(v_p, v_q) - c(v_p, z)) \geq y, \\ & c(v_p, z) \leq c(v_p, v_q), \\ & y, c(v_p, z) \geq 0. \end{aligned}$$

### 2.2 Tree Networks

Often in the literature, the maximin location problems on networks of a special type are considered. Network structure allows to find useful properties of the problem and determine new ways to solve it.

If a network is a path, then we can put  $n$  vertices on a real line and identify them with real numbers such that

$$0 = x_1 < x_2 < \dots < x_n$$

and  $d(x_i, x_j) = |x_i - x_j|$ . Then objective function  $r(z)$  is the following expression

$$r(z) = \min\{\alpha_i |z - x_i| : i = 1, \dots, n\} = \min\{r^+(z), r^-(z)\},$$

where

$$\begin{aligned} r^+(z) &= \min\{\alpha_i(z - x_i) : x_i \leq z\}, \\ r^-(z) &= \min\{\alpha_i(x_i - z) : x_i \geq z\}. \end{aligned}$$

In [16], a linear-time algorithm is given that finds a maximum value of the objective function  $r(z)$  along the path.

The star is a tree consisting of a central vertex  $v_0$  which has edges with  $n$  remaining vertices  $\{v_1, \dots, v_n\}$  [16]. By  $S_i$ , denote a problem which consists in determining a local optimal solution on an edge  $(v_0, v_i)$ . It is equivalent to the problem on the path as follows: we place all vertices on the real line in such a way that  $v_0$  in the point  $x_0 = 0$ , vertex  $v_i$  is to the right of  $v_0$  at a distance of  $c(v_0, v_i) = x_i$ , and all other vertices  $v_j$  ( $j \neq i$ ) are to the left of  $v_0$  with a distance of  $c(v_0, v_j) = x_j$ . In problem  $S_i$ , it is necessary to maximize the following function:

$$r_i(z) = \min_{j=0, \dots, n} \{ \min \alpha_j(x_j + z), \alpha_i(x_i - z) \}$$

for  $0 \leq z \leq x_i$ .

Let a maximum be reached in  $z^{(i)}$ . Point  $z^{(i)}$  is a solution following linear programming problem with two variables  $y$  and  $z$ :

$$\min\{z : y \leq \alpha_j(x_j + z), j = 0, \dots, n, y \geq \alpha_i(x_i - z)\}.$$

The constraints that are common to all problems  $S_i$  can be written in the form  $y \leq h(z)$  where

$$h(z) = \min_{j=0, \dots, n} \alpha_j(x_j + z).$$

The function  $h(z)$  is piecewise linear and increasing. Point  $z^{(i)}$  is an intersection of  $h(z)$  with  $\alpha_i(x_i - z)$ . Due to the monotonicity of  $h(z)$  it is required to calculate  $z^* = \max z^{(i)}$ . The value of  $z^*$  can be obtained as an optimal solution to following linear programming problem:

$$\min\{z : y \leq \alpha_j(x_j + z), y \geq \alpha_i(x_i - z), i, j = 0, \dots, n\} \tag{2}$$

Consider an optimal solution  $(z^*, y^*)$  of the problem (2). In [17], it is proved that optimal locations of the obnoxious facility are points on all edges  $(v_0, v_i)$  at the distance  $z^*$  from the central vertex  $v_0$  for which  $y^* = \alpha_i(x_i - z^*)$ .

The problem (2) has  $2n$  constraints and 2 variables. Using an algorithm from [4] for solving linear programming problem with 2 variables, problem (2) can be solved in linear time. Thus, the optimal solution of problem (1) on the star can be found in linear time.

Consider the maximin location problem on weighted trees. To solve the problem on an arbitrary tree with  $n$  vertices, two algorithms are proposed [9, 15] with time complexity  $O(n \log_2 n)$  and  $O(kn \log_2 n)$ , respectively. The parameter  $k$  depends on the structure of the tree. For paths and stars,  $k = O(1)$ . For a balanced tree,  $k = O(\log n)$  but there are trees such that  $k = \Theta(n)$ . For an unweighted tree, a linear algorithm is proposed [17].

### 3 Euclidean and Shortest Paths Metrics

Section deals with the location problem of an obnoxious facility on an arbitrary network with maximin criterion in Euclidean metric and a restriction on

transportation costs. The transportation costs are determined using the shortest paths metric. Model that takes into account the natural geometry of transport networks is proposed. The properties of the problem are found on the basis of which an exact algorithm for finding a solution of the problem is proposed. The analysis of the computational complexity of the algorithm is carried out.

### 3.1 Problem Formulation

There is a network of roads on a plane connecting some settlements. The population size in each settlement and the distances between them are known. It is necessary to place, for example, a waste recycling plant (facility) on the network. There is a budget of transportation costs for the transportation of waste from settlements to the facility. The facility has the adverse effects to the population and therefore it should be located as far away from the settlements. It is needed to place the facility on the network as far as possible from the settlements, taking into account the population size of the settlements and that transportation costs do not exceed the budget.

To formulate a mathematical model of the problem we use some previously introduced notations and introduce new ones. Let  $G = (V, E)$  be an undirected network representing the roads and settlements on a plane. The vertices of the network correspond to the settlements and the edges correspond to the roads. Two positive parameters  $(\alpha_i, w_i)$  are assigned to each vertex  $v_i, i \in N$ . Parameter  $\alpha_i$  reflects a degree of undesirability of placing the facility near the settlement corresponding  $v_i$ . Parameter  $w_i$ , on the contrary, reflects a requirement to place the facility as close as possible to the settlement corresponding  $v_i$ . Parameter  $w_i$ , for example, is the number of peoples in the settlement correspond to vertex  $v_i$ . Parameter  $\alpha_i$  is the inverse value of the number population in the settlement correspond to vertex  $v_i$ . Each vertex  $v_i$  has coordinates  $(a_i, b_i) i \in N$  on the plane. The edges of the network are segments of straight lines on the plane with known lengths  $c(v_i, v_j) > 0, (v_i, v_j) \in E, i, j \in N$ . As in Sect. 2, we denote by  $Z$  the infinite set of points on the network  $G$ . The adverse effects of the facility to the settlements will be measured in Euclidean metric  $\rho(v_i, z), z \in Z, i \in N$ . We will use the shortest paths  $d(v_i, z), z \in Z, i \in N$  in  $G$  for determining transportation costs of servicing the settlements. Let  $T$  be an available budget for the transportation of waste from settlements to the facility. The mathematical model has form:

$$r_1(z) = \min_{i \in N} \alpha_i \rho(v_i, z) \rightarrow \max_z, \quad (3)$$

$$\sum_{i \in N} w_i d(v_i, z) \leq T, \quad (4)$$

$$z \in Z. \quad (5)$$

Expression (3) means maximizing of the minimum weighted distance from the facility to the vertices. Condition (4) guarantees the fulfillment of the restriction on transportation costs. The left part of the inequality (4) is a sum of the weighted shortest paths from some point  $z \in Z$  to all vertices of the network.

Condition (5) means, that the facility is located on infinite set of points on all edges of  $G$ , including all vertices. Problem (3)–(5) is a nonlinear, nonconvex problem with one or more local maximum.

Note that a linear approximation can be used to model real road sections. This will lead to the additional introduction of dummy vertices and edges to the network. This method is used, for example, in [6]. In this paper, we do not consider any method of approximating real roads by straight-line segments.

In [18], an algorithm for finding an approximate solution to the problem is proposed. The algorithm considers a finite set of points on an arbitrary edge of the network. The values of the objective function are calculated at the points, where the budget restriction is met. The point with a maximum value of the objective function determines a local maximum on the edge. An approximate solution to the problem (3)–(5) is the best of the point. An experiment was conducted to solve the problem with the proposed algorithm on the network of the main railway lines of France.

Problems close to the formulated one without taking into account the budget constraint were studied, for example, in [11, 17].

In [19–21], a problem of locating an obnoxious facility in a polygonal area on a plane (polygon) is considered. The problem is finding a point in the polygon, which maximizes a minimum weighted Euclidean distance to given set points in the polygon. It does not matter whether the polygon is convex or not. In fact, this is the problem of maximization function (3) in the polygonal area on the plane. It is proved, that an optimal solution is either in the convex hull of the vertices or on the boundary (segments) of the polygon. In the case of searching the solution on the boundary segment (local optimum) it is necessary to solve a system of two equations: the linear equation of the boundary segment and the nonlinear equation of Euclidean distances. If the local optimum is not located on the boundary of the polygon, then it is necessary to solve a system of three equations of Euclidean distances.

### 3.2 Transportation Costs Function

Let's analyze some properties the sum function of weighted shortest paths from some point  $z \in Z$  on arbitrary edge  $(v_p, v_q)$  to all vertices of the network  $G$ .

$$\sigma(z) = \sum_{i \in N} w_i d(z, v_i). \quad (6)$$

In [11] a concept of edge bottleneck points is introduced. Let a point  $x$  be located on the edge  $(v_p, v_q)$ . The point  $x$  is an edge bottleneck point with respect to vertex  $v_k$  if there is a vertex  $v_k$  such that

$$d(v_k, v_p) + c(v_p, x) = d(v_k, v_q) + c(v_q, x),$$

where  $c(v_p, x), c(v_q, x) > 0$  are the distances from  $v_p$  and  $v_q$  to point  $x$  on the edge  $(v_p, v_q)$ . The equality  $c(v_p, x) + d(v_q, x) = c(v_p, v_q)$  is correct.

Note that a bottleneck point on edge  $(v_p, v_q)$  with respect to vertex  $v_k$  is associated with a cycle formed by the shortest path from vertex  $v_k$  to vertex  $v_p$ , edge  $(v_p, v_q)$ , and the shortest path from vertex  $v_q$  to back to vertex  $v_k$ . This is illustrated in Fig. 1., where the shortest paths from vertex  $v_k$  to  $v_p$  and  $v_q$  are shown in the form of curved lines. The cycle contains the bottleneck point  $x$ .

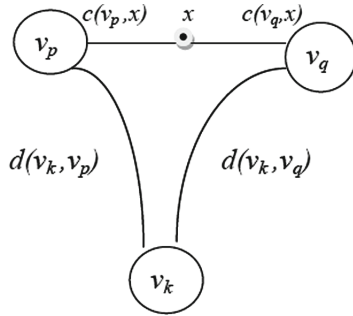


Fig. 1. Cycle contains the bottleneck point  $x$ .

Edge  $(v_p, v_q)$  contains an edge bottleneck point with respect to vertex  $v_k$  if and only if following inequality holds

$$|d(v_k, v_p) - d(v_k, v_q)| < c(v_p, v_q). \tag{7}$$

This means that there is no shortest path from  $v_k$  to  $v_p$  and from  $v_k$  to  $v_q$  containing the edge  $(v_p, v_q)$ .

The papers [1, 11] show that  $\sigma(z)$  on an edge is continuous, piecewise linear, concave function with critical points (intersection of linear functions) only at the bottleneck points of the edge. If  $G$  is a tree, then in this case there are no edge bottleneck points as there are no cycles. Then the following proposition is true.

**Proposition 1.** *If  $G$  is a tree, then function  $\sigma(z)$  is linear on arbitrary edge.*

*Proof.* Let  $(v_p, v_q) \in E$  be an edge of the  $G$ . The set of vertices of the network  $G$  by the edge  $(v_p, v_q)$  is divided into two sets:  $V_L$  and  $V_R$ . The set  $V_L$  is such set of vertices of the network  $G$  that the paths from them to the  $v_q$  pass through the vertex  $v_p$ . By  $N_L$  denote a set of indexes of such vertices. The set  $V_R$  is such set of vertices of the network  $G$  that the paths from them to the  $v_p$  pass through the vertex  $v_q$ . By  $N_R$  denote a set of indexes of such vertices. There are relations  $V_L \cap V_R = \emptyset$  and  $V_L \cup V_R = V$ . The following equalities take place.

$$d(z, v_i) = d(v_i, v_p) + c(v_p, z), i \in N_L.$$

$$d(z, v_i) = d(v_i, v_q) + c(v_p, v_q) - c(v_p, z), i \in N_R.$$



$$\begin{aligned} \sigma(z) &= \sum_{i \in N} w_i d(z, v_i) = \sum_{i \in N_L} w_i d(z, v_i) + \sum_{i \in N_R} w_i d(z, v_i) = \\ &= \sum_{i \in N_L} w_i [d(v_i, v_p) + c(v_p, z)] + \sum_{i \in N_R} w_i [d(v_i, v_q) + c(v_p, v_q) - c(v_p, z)] = \\ &= \sum_{i \in N_L} w_i d(v_i, v_p) + c(v_p, z) \sum_{i \in N_L} w_i + \sum_{i \in N_R} w_i d(v_i, v_q) + \\ &+ c(v_p, v_q) \sum_{i \in N_R} w_i - c(v_p, z) \sum_{i \in N_R} w_i = \\ &= \sum_{i \in N_L} w_i d(v_i, v_p) + \sum_{i \in N_R} w_i d(v_i, v_q) + c(v_p, v_q) \sum_{i \in N_R} w_i + \\ &+ c(v_p, z) \left( \sum_{i \in N_L} w_i - \sum_{i \in N_R} w_i \right). \end{aligned}$$

Let's put

$$\begin{aligned} a &= \sum_{i \in N_L} w_i d(v_i, v_p) + \sum_{i \in N_R} w_i d(v_i, v_q) + c(v_p, v_q) \sum_{i \in N_R} w_i, \\ b &= \sum_{i \in N_L} w_i - \sum_{i \in N_R} w_i. \end{aligned}$$

We get the equation of a straight line  $\sigma(z) = a + bc(v_p, z)$ . Moreover,  $b$  it can be negative or positive. Note that this form will have function  $\sigma(z)$  on any edge of a network, if the edge is a bridge.

### 3.3 Domain of Admissible Solutions

Let's look some properties of the problem (3)–(5) that allow to find all local optimums and exact solution to the problem. A domain of admissible solutions of the problem is only some part of the network  $G$  due to the restriction of transportation costs (budget). All edge segments, on which the value of transportation costs do not exceed the value of the budget  $T$  form the domain of admissible solutions of the problem (3)–(5). By  $D$  denote the domain. If the budget restriction is violated for the vertices of an edge, then the edge does not belong to the domain  $D$ . It follows from the concavity property of the transportation costs function.

Before constructing the domain of admissible solutions of the problem (3)–(5) on an edge  $(v_p, v_q)$ , it is necessary to find all edge bottleneck points. The following algorithm for finding such points on the edge is proposed. Consistently consider all vertices of the network  $G$ . For a current vertex  $v_k$  we check performing of the inequality (7). If the inequality (7) does not hold, then the edge  $(v_p, v_q)$  does not contain an edge bottleneck point relative to  $v_k$ , and we move to another vertex  $G$ . If the inequality holds, then the edge  $(v_p, v_q)$  contains an edge bottleneck point relative to the vertex  $v_k$ . Distance  $s$  from the vertex  $v_p$  to the point is equal to

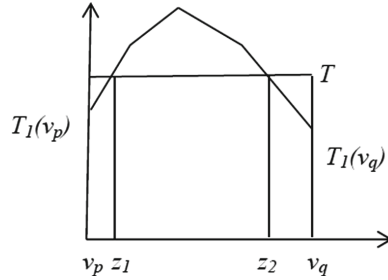
$$s = \frac{|d(v_k, v_p) - d(v_k, v_q)| + c(v_p, v_q)}{2}.$$

After constructing all the bottleneck points on the edge, we determine the domain of admissible solutions on the edge. To do this, we arrange the bottleneck points in ascending order  $s_1 < s_2 < \dots < s_n$  from the vertex  $v_p, p < q$ . Sequentially calculate following values

$$T_1(s_i) = \sum_{k \in N} w_k \min\{d(v_k, v_p) + s_i; d(v_k, v_q) + c(v_p, v_q) - s_i\}.$$

The following variants are possible.

- 1)  $T_1(v_p) \leq T$  and  $T_1(v_q) \leq T$ .
- 1) We look through the bottleneck points sequentially. There is a pair of points  $s_t$  and  $s_{t+1}$  such that  $T_1(s_t) \leq T$  and  $T_1(s_{t+1}) > T$ . If  $T_1(s_t) < T$ , then we construct the equation of the line  $l(s_t, s_{t+1})$  through the points  $(s_t, T_1(s_t))$  and  $(s_{t+1}, T_1(s_{t+1}))$ . Solving the equation  $l(s_t, s_{t+1}) = T$ , and we find a point  $z_1$ , for which  $T_1(z_1) = T$ . If  $T_1(s_t) = T$ , then the value of  $z_1$  is defined as  $z_1 = s_t$ . Next, we calculate a value of the function  $T_1$  at the points  $s_{t+2}, s_{t+3}$  etc. There is such  $k$ , that  $T_1(s_k) > T$  and  $T_1(s_{k+1}) \leq T$ . If  $T_1(s_{k+1}) < T$ , then we construct the equation of the line  $l(s_k, s_{k+1})$  through the specified pair of the points. Solving the equation  $l(s_k, s_{k+1}) = T$ , and we find a point  $z_2$ , for which  $T_1(z_2) = T$ . If  $T_1(s_{k+1}) = T$ , then the value of  $z_2$  is defined as  $z_2 = s_{k+1}$ . Thus two points are found on the edge  $(v_p, v_q)$ , for which  $T_1(z_1) = T_1(z_2) = T$ . The domain of admissible solutions on the edge  $(v_p, v_q)$  represents two segments:  $[v_p, z_1]$  and  $[z_2, v_q]$ , as shown in Fig. 2. If there is no such pair of points  $s_t$  and  $s_{t+1}$ , for which  $T_1(s_t) \leq T$  and  $T_1(s_{t+1}) > T$ , then all points of the edge  $(v_p, v_q)$  belong to the domain of admissible solutions.
- 2)  $T_1(v_p) \leq T$  and  $T_1(v_q) > T$ .  
There is a pair of points  $s_t$  and  $s_{t+1}$  such that  $T_1(s_t) \leq T$  and  $T_1(s_{t+1}) > T$ . If  $T_1(s_t) < T$ , then we construct the equation of the line  $l(s_t, s_{t+1})$  through the points  $(s_t, T_1(s_t))$  and  $(s_{t+1}, T_1(s_{t+1}))$ . Solving the equation  $l(s_t, s_{t+1}) = T$ , and we find the point  $z_1$ , for which  $T_1(z_1) = T$ . If  $T_1(s_t) = T$ , then the value of  $z_1$  is defined as  $z_1 = s_t$ . The domain of admissible solutions on the edge  $(v_p, v_q)$  represents the segment  $[v_p, z_1]$ .
- 3)  $T_1(v_p) > T$  and  $T_1(v_q) \leq T$ .  
There is a pair of points  $s_t$  and  $s_{t+1}$  such that  $T_1(s_t) > T$  and  $T_1(s_{t+1}) \leq T$ . If  $T_1(s_{t+1}) < T$ , then we construct the equation of the line  $l(s_t, s_{t+1})$  through the points  $(s_t, T_1(s_t))$  and  $(s_{t+1}, T_1(s_{t+1}))$ . Solving the equation  $l(s_t, s_{t+1}) = T$ , and we find the point  $z_2$ , for which  $T_1(z_2) = T$ . If  $T_1(s_{t+1}) = T$ , then the value of  $z_2$  is defined as  $z_2 = s_{t+1}$ . The domain of admissible solutions on the edge  $(v_p, v_q)$  represents a segment  $[z_2, v_q]$ .



**Fig. 2.** Function  $\sigma(z)$  and Domain of admissible solutions on edge  $(v_p, v_q)$ .

As a result, the domain of admissible solutions of the problem (3)–(5) on an arbitrary edge  $(v_p, v_q)$  has no more than two segments. One of the ends of these segments will be vertex  $v_p$  or  $v_q$ . Thus, to solve problem (3)–(5) on the domain  $D$ , this is solving a series of the problems on edge segments of the network  $G$ .

If  $G$  is a tree, then there is no need to find bottleneck points. If  $T_1(v_p) \leq T$  and  $T_1(v_q) > T$  ( $T_1(v_p) > T$  and  $T_1(v_q) \leq T$ ), we find an intersection point  $z_1$  ( $z_2$ ) of the straight line  $l(v_p, v_q)$  with the straight line  $y = T$ . Thus, we get the domain of admissible solutions on edge  $(v_p, v_q)$  segment  $[v_p, z_1]$  ( $[z_2, v_q]$ ).

### 3.4 Algorithm

At the first step of the algorithm for solving the problem (3)–(5) it is necessary to check, that the domain  $D$  is not empty. To do this, we solve a problem of finding 1-median on the network in the shortest paths metric. If the solution of the problem has the value of the objective function no more than  $T$ , then the domain of admissible solutions of problem (3)–(5) is not empty.

When searching a local optimum on an edge  $(v_p, v_q)$  belonging to the admissible domain, it is necessary to construct a weighted Voronoi diagram for the vertices of the network  $G$  [3, 5]. Next, we find the intersection points of the edges of the Voronoi diagram with the segments admissible domain  $D$  on the edge  $(v_p, v_q)$ . As a result, the edge segments will be divided into pieces by the intersection points. Each the piece will be located inside some locus of the Voronoi diagram. The function  $r_1$  is convex on each such piece [6]. Consequently, the function  $r_1$  reaches an optimal value at the intersection points of the edges of the Voronoi diagram with the edge segments  $(v_p, v_q)$  belonging to the domain  $D$ . Choosing a point with a maximum value of the function  $r_1$ , we find a local optimum of the problem (3)–(5) on the edge  $(v_p, v_q)$ .

Looking through all the edges of the network  $G$ , the segments of which belong to the domain of admissible solutions  $D$ , we get a set of local optimums. Choosing a maximum value from them, we obtain a global solution to the problem (3)–(5).

The algorithm for finding a local optimum of the problem (3)–(5) on an edge can be briefly presented as follows.

- Step 1.** Solve the 1-median problem on network  $G$ . If optimal value of objective function of the problem is greater  $T$ , then the problem (3)–(5) has no solution.
- Step 2.** Construct the domain of admissible solutions  $D$ .
- Step 3.** Construct the weighted Voronoi diagram of vertices of the network  $G$ .
- Step 4.** Find intersection points of all Voronoi edges with the segments of the edge, belonging to the domain  $D$ .
- Step 5.** Calculate the values of the function (3) at the specified intersection points. The point, at which the maximum value of the objective function is reached, will be a local optimum of the problem on the edge.

There are  $O(n^2)$  vertices and  $O(n^2)$  edges in a weighted Voronoi diagram, which can be generated in  $O(n^2)$  time. There are  $O(mn^2)$  intersection points the edges of Voronoi diagram with the edges of network  $G$ , since each Voronoi edge can be intersect an edge of network  $G$  at most twice [3]. Therefore a complexity of the algorithm is  $O(mn^2)$ .

## 4 Conclusion

The problems of optimal location of an obnoxious facility on a networks of roads connecting settlements are considered. It is necessary to find such location of the facility so that the minimum distance to the nearest settlement is as large as possible. An overview of the properties and algorithms for solving the problems using the shortest paths metric is given.

The main attention is paid to the problem taking into account a restriction on transportation costs for servicing the settlements by the facility. The objective function uses Euclidean metric. The restriction take into account the shortest paths in the network. A polynomial algorithm for finding all local maximums on the edges of the network is proposed. The choice of the optimal solution among the specified optimums can be made by the decision-maker from any additional conditions. Considered model and proposed algorithm can be applied for solving, for example, a problem of location a waste processing plant to reduce the adverse effects to the population.

One of the conditions of the problem that is the assumption, that roads are line segments on a plane. Real roads can be approximated by straight-line segments. Furthermore, an interesting continuation the study of the problem is it solution without the use of Voronoi diagram. For example, using an approach to solving a maximin problem within a bounded region on a plane with Euclidean metric and applying Karuch-Kuhn-Tucker optimality conditions.

## References

1. Eiselt, H.A., Marianov, V.: Foundations of Location Analysis. Springer, New York (2011). <https://doi.org/10.1007/978-1-4419-7572-0>
2. Glukov, V.I., Varepo, L.G., Shalay, V.V., Simonenko, R.V., Belyave, P.S.: Coordinate structure geometrical location specifications. J. Phys.: Conf. Ser. **1901** (2021). <https://doi.org/10.1088/1742-6596/1901/1/012008>

3. Melachrinoudis, E., Smith, J.: An  $O(mn^2)$  algorithm for the maximin problem in  $E^2$ . *Oper. Res. Lett.* **18**, 25–30 (1995)
4. Megiddo, N.: Linear-time algorithms for linear programming in  $R^3$  and related problems. *SIAM J. Comput.* **12**, 759–776 (1983). <https://doi.org/10.1137/0212052>
5. Preparata, P., Shamos, M.: *Computational Geometry: An Introduction*. Springer-Verlag, New York (1985)
6. Heydari, R., Melachrinoudis, E.: Location of a semi-obnoxious facility with elliptic maximin and network minisum objectives. *Eur. J. Oper. Res.* **223**, 452–460 (2012). <https://doi.org/10.1016/j.ejor.2012.06.039>
7. Berman, O., Drezner, Z.: A note on the location of an obnoxious facility on a network. *Eur. J. Oper. Res.* **120**, 215–217 (2000)
8. Hakimi, S.: Optimum locations of switching centers and the absolute centers and medians of a graph. *Oper. Res.* **11**, 450–459 (1964). <https://doi.org/10.1287/opre.12.3.450>
9. Tamir, A.: Improved complexity bounds for center location problems on networks by using dynamic data structures. *SIAM J. Discr. Math.* **1**, 377–396 (1988). <https://doi.org/10.1137/0401038>
10. Moon, I.: Maximin center of pendant vertices in a tree network. *Transp. Sci.* **23**, 213–216 (1989). <https://doi.org/10.1287/trsc.23.3.213>
11. Church, R.L., Garfinkel, R.S.: Location an obnoxious facility on a network. *Transp. Sci.* **12**, 107–118 (1978). <https://doi.org/10.1287/trsc.12.2.107>
12. Melachrinoudis, E., Zhang, G.: An  $O(mn)$  algorithm for the 1-maximin problem on a network. *Comput. Oper. Res.* **26**, 849–869 (1999). [https://doi.org/10.1016/S0305-0548\(98\)00099-9](https://doi.org/10.1016/S0305-0548(98)00099-9)
13. Berman, O., Drezner, Z., Wang, J., Wesolowsky, G.O.: The minimax and maximin location problems on a network with uniform distributed weights. *IIE Trans.* **35**, 1017–1025 (2003). <https://doi.org/10.1080/07408170304397>
14. Melaehrinoudis, E., Yavuz, E., Heydari, R.: An  $O(m^2 + mn^2)$  algorithm for the bi-objective location problem on a network with mixed metrics. *Int. J. Oper. Res.* **23**, 427–450 (2015). <https://doi.org/10.1504/IJOR.2015.070144>
15. Tamir, A.: Obnoxious facility location on graphs. *SIAM J. Discr. Math.* **4**, 550–567 (1991). <https://doi.org/10.1137/0404048>
16. Burkard, R., Dollani, H., Lin, Y., Rote, G.: The obnoxious center problem on a tree. *SIAM J. Discr. Math.* **14**, 498–509 (2001). <https://doi.org/10.1137/S0895480198340967>
17. Burkard, R., Dollani, H.: Center problems with pos/neg weights on trees. *Eur. J. Oper. Res.* **145**, 483–495 (2003). [https://doi.org/10.1016/S0377-2217\(02\)00211-4](https://doi.org/10.1016/S0377-2217(02)00211-4)
18. Zabudsky, G.G., Lisina, M.S.: Approximately algorithm for maximin location problem on network. In: *IEEE Conference 2018 Dynamics of Systems, Mechanisms and Machines*, Omsk (2018). <https://doi.org/10.1109/Dynamics.2018.8601502>
19. Melachrinoudis, E., Cullinane, T.P.: An locating an obnoxious facility within a polygonal region. *Ann. Oper. Res.* **6**, 137–145 (1986). <https://doi.org/10.1007/BF02026821>
20. Dasarathy, B., White, L.J.: A maximin location problem. *Oper. Res.* **28**, 1385–1401 (1980). <https://doi.org/10.1287/opre.28.6.1385>
21. Drezner, Z., Wesolowsky, G.O.: A maximin location problem with maximum distance constraints. *AIIE Trans.* **12**, 249–252 (1980). <https://doi.org/10.1080/05695558008974513>