Storage Location Problem: Properties and Computational Aspects

Petro Stetsyuk, Viktor Stovba and Oleksandr Zhmud

V.M. Glushkov Institute of Cybernetics of the NASU, Academician Glushkov Avenue, 40, Kyiv, 03187, Ukraine

Abstract

Nonlinear programming problem for optimal location of storages is studied so that the total distance, taken into account with coefficients, which are the volumes of products transported from storages to markets (consumers), is minimal. It is shown that the objective function of the problem satisfies a special inequality and, in general case, is a non-smooth nonconvex function. The consistency conditions of linear constraints system of the problem and its variants depending on balance conditions that define degeneracy and non-degeneracy of the constraints system are substantiated. An example of the problem is given when the solver MINOS 5.51 does not obtain a solution to a degenerate problem and obtains a solution to a non-degenerate problem. Work of NEOS server solvers for solving the storage location problem depending on the starting point and degree of degeneracy of the system is studied.

Keywords

Storage location problem, transportation problem, nonlinear programming problem, degeneracy of linear constraints system, software, NEOS server, AMPL, MINOS

1. Introduction

Objects location problem belongs to problems of transport and production type and quite often arises in practice in such areas as health care, waste management system, logistics and transportation of products from producers to consumers (or intermediaries with further transportation to consumers), etc. A large number of publications are devoted to theoretical, computational and applied aspects of this problem. In particular, works [1–3] discuss concepts, models, and algorithms for solving facility location problems, works [4, 5] offer new approaches to their solving. The work [6] examines the optimization problems of production and transport type, as well as methods and algorithms for their solving. The work [7] is devoted to solving the problem of m-travelers and nonlinear programming problem using NEOS server solvers. The work [8] considers so called multi-level facility location problems, which extend some classical facility location problems.

Object location problem is closely related to centroid-based clustering problems so that the optimal solution of the first problem corresponds to a certain partition of a set of points into classes, i.e. solution of a clustering problem. In the general formulation, the object location problem is NP-hard, but it can be reduced to other types of problems, in particular to set cover problem [9].

As a partial case of the object location problem, the storage location problem formulated in the book [10, section 14.2, pp. 370–371] can be considered. In this problem, it is needed to choose the optimal coordinates for the storages locations for markets, where the total distance, weighted by the volumes of products that need to be transported to markets (consumers), is minimized. Here, the coordinates of the storages location are not chosen from the set of potential locations of the storages, but can have arbitrary coordinates on the plane.

The material of the article is presented in the following order. In the second section, the formulation of the nonlinear programming problem for the optimal storage location is given. The third section examines the properties of the objective function of this problem and shows that its solution

International Scientific Symposium «Intelligent Solutions» IntSol-2023, September 27–28, 2023, Kyiv-Uzhhorod, Ukraine
EMAIL: stetsyukp@gmail.com (A. 1); vik.stovba@gmail.com (A. 2); zhmud17@gmail.com (A. 3)
ORCID: 0000-0003-4036-2543 (A. 1); 0000-0003-3023-5815 (A. 2); 0000-0002-4591-1110 (A. 3)

may not be unique. The fourth section substantiates the consistency conditions of the constraints system of the problem and gives an example of a problem that the MINOS solver solves in degenerate case, but does not solve in non-degenerate case, determined by the balance conditions of the constraints system of the problem. The fifth chapter deals with nonlinear programming problem for optimal storage location with equality constraints. It is shown that one arbitrary linear constraint is linearly dependent. An example of solving degenerate and non-degenerate variants of the problem, which are determined by the presence or absence of one linearly dependent constraint, using NEOS server solvers is given. The sixth chapter examines the problem of location 5 storages for transporting products to 19 the most common markets of Kyiv and examines the work of NEOS server solvers depending on the starting point and the degeneracy degree of the constraints system.

2. Storage location problem formulation [9]

Let the locations of n markets (consumers) and the volume of demand on each of them be given. The demand can be satisfied from m storages with given capacities. It is needed to locate these m storages so that the total distance, calculated with weighting coefficients equal to the volumes of products transported from storages to markets, is minimal. It is important to emphasize, that in practice such a criterion for solution evaluation is the ton-kilometer indicator.

Let us build a model of the problem. The notation is the following:

(x_i, y_i) – unknown coordinated of i-th storage ($i = \overline{1,m}$);

c_i – known capacity of i-th storage ($i = \overline{1,m}$);

$\left(a_j, b_j\right)$ – known coordinates of j-th market (consumer) ($j = \overline{1,n}$);

r_j – known volume of j-th market ($j = \overline{1,n}$);

$d_{ij} = \sqrt{\left(x_i - a_j\right)^2 + \left(y_i - b_j\right)^2}$ – distance from i-th storage to j-th market ($i = \overline{1,m}, j = \overline{1,n}$);

z_{ij} – volume of products, transported from i-th storage to j-th market ($i = \overline{1,m}, j = \overline{1,n}$).

Then the m storage location problem and determination of products volumes to be transported from storages to markets is formulated as follows:

$$f(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} z_{ij} = \sum_{i=1}^{m}\sum_{j=1}^{n} z_{ij}\sqrt{\left(x_i - a_j\right)^2 + \left(y_i - b_j\right)^2} \to \min \qquad (1)$$

subject to

$$\sum_{j=1}^{n} z_{ij} \le c_i, \quad i = \overline{1,m}, \qquad (2)$$

$$\sum_{i=1}^{m} z_{ij} = r_j, \quad j = \overline{1,n}, \qquad (3)$$

$$z_{ij} \ge 0, \quad i = \overline{1,m}, \quad j = \overline{1,n}. \qquad (4)$$

The problem (1) – (4) is nonlinear programming problem, for the objective function $f(\mathbf{x}, \mathbf{y}, \mathbf{z})$ is nonlinear and non-smooth. Here variables are coordinates of storages (x_i, y_i) and transportation volumes z_{ij}. If storages locations are known, then distances d_{ij} are known, and only transportation volumes z_{ij} are to be determined. In this case the problem (1) – (4) is opened transportation problem.

3. Properties of objective function of the problem (1) – (4)

For the storage location problem, the objective function (1), depending on $m(n+2)$ continuous variables, is non-smooth nonlinear function and has the following property.

Lemma 1. For the function $f(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{i=1}^{m}\sum_{j=1}^{n} z_{ij}\sqrt{\left(x_i - a_j\right)^2 + \left(y_i - b_j\right)^2}$ and arbitrary $\lambda \in [0,1]$ the following inequality is true:

$$f(\lambda\boldsymbol{x^1}+(1-\lambda)\boldsymbol{x^2},\lambda\boldsymbol{y^1}+(1-\lambda)\boldsymbol{y^2},\lambda\boldsymbol{z^1}+(1-\lambda)\boldsymbol{z^2})\le$$
$$\le\sum_{i=1}^{m}\sum_{j=1}^{n}\lambda f(\boldsymbol{x^1},\boldsymbol{y^1},\boldsymbol{z^1})+(1-\lambda)f(\boldsymbol{x^2},\boldsymbol{y^2},\boldsymbol{z^2})+ \qquad (5)$$
$$+\lambda(\lambda-1)\left(\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}-\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}\right)\left(z_{ij}^1-z_{ij}^2\right).$$

Proof. Using norm convexity and $\lambda\in[0,1]$, let us estimate the left-hand side of the inequality (5):
$$f(\lambda\boldsymbol{x^1}+(1-\lambda)\boldsymbol{x^2},\lambda\boldsymbol{y^1}+(1-\lambda)\boldsymbol{y^2},\lambda\boldsymbol{z^1}+(1-\lambda)\boldsymbol{z^2})=$$
$$=\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\lambda z_{ij}^1+(1-\lambda)z_{ij}^2\right)\sqrt{\left(\lambda x_i^1+(1-\lambda)x_i^2-a_j\right)^2+\left(\lambda y_i^1+(1-\lambda)y_i^2-b_j\right)^2}\le \qquad (6)$$
$$\le\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\lambda z_{ij}^1+(1-\lambda)z_{ij}^2\right)\left(\lambda\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+(1-\lambda)\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}\right).$$

Opening brackets in the right-hand side of the inequality (6) and grouping terms, we get:
$$\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\lambda z_{ij}^1+(1-\lambda)z_{ij}^2\right)\left(\lambda\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+(1-\lambda)\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}\right)=$$
$$=\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\lambda^2z_{ij}^1\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+\lambda(1-\lambda)z_{ij}^1\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}+\right.$$
$$\left.+\lambda(1-\lambda)z_{ij}^2\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+(1-\lambda)^2z_{ij}^2\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}\right)=$$
$$=\sum_{i=1}^{m}\sum_{j=1}^{n}\left((\lambda-\lambda+\lambda^2)z_{ij}^1\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+\lambda(1-\lambda)z_{ij}^1\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}+\right.$$
$$+\lambda(1-\lambda)z_{ij}^2\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+\left((1-\lambda)-(1-\lambda)+\right.$$
$$\left.+(1-\lambda)^2\right)z_{ij}^2\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}\Bigg)=\sum_{i=1}^{m}\sum_{j=1}^{n}\lambda z_{ij}^1\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+$$
$$+(1-\lambda)z_{ij}^2\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}+\lambda(\lambda-1)z_{ij}^1\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+$$
$$+\lambda(1-\lambda)z_{ij}^1\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}+\lambda(1-\lambda)z_{ij}^2\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}-$$
$$+\lambda(\lambda-1)z_{ij}^2\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}=\sum_{i=1}^{m}\sum_{j=1}^{n}\lambda z_{ij}^1\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+$$
$$+(1-\lambda)z_{ij}^2\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}-\lambda(1-\lambda)\times$$
$$\times\left(z_{ij}^1\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}-z_{ij}^1\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}-\right.$$
$$\left.-z_{ij}^2\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}+z_{ij}^2\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}\right)=$$
$$=\sum_{i=1}^{m}\sum_{j=1}^{n}\lambda f(\boldsymbol{x^1},\boldsymbol{y^1},\boldsymbol{z^1})+(1-\lambda)f(\boldsymbol{x^2},\boldsymbol{y^2},\boldsymbol{z^2})+$$
$$+\lambda(\lambda-1)\left(\sqrt{\left(x_i^1-a_j\right)^2+\left(y_i^1-b_j\right)^2}-\sqrt{\left(x_i^2-a_j\right)^2+\left(y_i^2-b_j\right)^2}\right)\left(z_{ij}^1-z_{ij}^2\right),$$

from which we get that the inequality (5) is fulfilled.

In general case, the function (1) is not convex, and the problem (1) – (4) is multiextremal. However, there are partial cases in which the function (1) may be convex if the last term of the right-hand side of the inequality (5) is negative. For this problem, this condition means that, considering two sets of possible storage locations (two supplier firms), for each "storage – market" pair, the first supplier firm either transports a larger volume of products over a shorter distance, or transports a smaller volume of goods, covering a greater distance than the second supplier firm.

It is easy to demonstrate that in general case the solution of the problem (1) – (4) is not unique. Let us consider the problem (1) – (4) with $m = 1$ and $n = 2$, i.e. there are 2 markets with coordinates $(0,0)$ and $(10,0)$. Each market requires the same number of units of the product, for example 10 units. It is needed to choose the location of one storage for transporting products to these two markets (see Fig. 1).

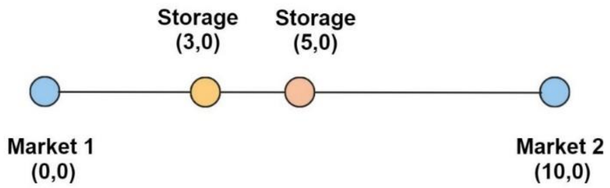

Figure 1: two solutions of the problem (1) – (4) for $m = 1$ and $n = 2$

It is clear that the optimal location of the storage is on the line segment connecting two markets. Let us calculate the value of the objective function (1) for two possible locations of storages with coordinates $(3,0)$ and $(5,0)$:

$$f(3{,}0{,}10) = 10\sqrt{(3-0)^2 + (0-0)^2} + 10\sqrt{(3-10)^2 + (0-0)^2} = 100,$$
$$f(5{,}0{,}10) = 10\sqrt{(5-0)^2 + (0-0)^2} + 10\sqrt{(5-10)^2 + (0-0)^2} = 100.$$

The same value of the objective function (1) for two different points of storage location shows that the solution of the problem can be not unique.

However, if we remove the square root in the second multiplier under the sum sign in the function $f(\mathbf{x}, \mathbf{y}, \mathbf{z})$, i.e., instead of Euclidean distance (norm) we use norm squared, the function will be as follows:

$$f_1(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{i=1}^{m} \sum_{j=1}^{n} z_{ij} \left(\left(x_i - a_j\right)^2 + \left(y_i - b_j\right)^2 \right). \tag{7}$$

The function (7) is smooth and non-convex function. For $m = 1$ and $n = 2$ the solution of the problem (7), (2) – (4) is unique.

4. Properties of the constraints system of the problem (1) – (4)

Linear constraints system (2) – (4) depends on variables z_{ij} only and is typical for opened transportation problems with m suppliers (storages) and n consumers (markets). To clarify the consistency of the system the following criterion ca be used.

Lemma 2. Constraints system (2) – (4) is consistent if and only if $\sum_{i=1}^{m} c_i \geq \sum_{j=1}^{n} r_j$.

Proof. Necessity. Let there exist non-negative $(\bar{x}_i, \bar{y}_i)$ and $\bar{z}_{ij}$, $(i = \overline{1,m},\ j = \overline{1,n})$ that satisfy the constraints system (2) – (4), i.e. the following equalities and inequalities are true:

$$\sum_{j=1}^{n} \bar{z}_{ij} \leq c_i, \quad i = \overline{1,m}, \tag{8}$$

$$\sum_{i=1}^{m} \bar{z}_{ij} = r_j, \quad j = \overline{1,n}, \tag{9}$$

$$\bar{z}_{ij} \geq 0, \quad i = \overline{1,m}, \quad j = \overline{1,n}. \tag{10}$$

Summing the inequality (8) with index $i = \overline{1,m}$ and equality (9) with index $j = \overline{1,n}$, we get:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} \bar{z}_{ij} \leq \sum_{i=1}^{m} c_i, \tag{11}$$

$$\sum_{j=1}^{n} \sum_{i=1}^{m} \bar{z}_{ij} = \sum_{j=1}^{n} r_j. \tag{12}$$

Using the inequality (11) and equality (12), and changing indices order, we get the following inequalities chain:

$$\sum_{i=1}^{m} c_i \geq \sum_{i=1}^{m}\sum_{j=1}^{n} \bar{z}_{ij} = \sum_{j=1}^{n}\sum_{i=1}^{m} \bar{z}_{ij} = \sum_{j=1}^{n} r_j. \tag{13}$$

From (13) we obtain that the inequality $\sum_{i=1}^{m} c_i \geq \sum_{j=1}^{n} r_j$ is true.

Sufficiency. Let the inequality $\sum_{i=1}^{m} c_i \geq \sum_{j=1}^{n} r_j$ be true. Let us consider variables $\tilde{z}_{ij}$, ($i = \overline{1,m}$, $j = \overline{1,n}$), defined as follows:

$$\tilde{z}_{ij} = \frac{c_i r_j}{\sum_{i=1}^{m} c_i}, \quad i = \overline{1,m}, \; j = \overline{1,n}, \tag{14}$$

and variables $(\tilde{x}_i, \tilde{y}_i)$ are arbitrary. Let us show that these variables satisfy constraints (2) – (4).

1. $\sum_{j=1}^{n} \tilde{z}_{ij} = \sum_{j=1}^{n} \frac{c_i r_j}{\sum_{i=1}^{m} c_i} = \frac{c_i}{\sum_{i=1}^{m} c_i}\sum_{j=1}^{n} r_j = c_i \frac{\sum_{j=1}^{n} r_j}{\sum_{i=1}^{m} c_i} = c_i \cdot d \leq c_i, \quad i = \overline{1,m},$

   where $d = \frac{\sum_{j=1}^{n} r_j}{\sum_{i=1}^{m} c_i} \leq 1$, since $\sum_{i=1}^{m} c_i \geq \sum_{j=1}^{n} r_j$.

2. $\sum_{i=1}^{m} \tilde{z}_{ij} = \sum_{i=1}^{m} \frac{c_i r_j}{\sum_{i=1}^{m} c_i} = \frac{r_j}{\sum_{i=1}^{m} c_i}\sum_{i=1}^{m} c_i = r_j, \quad j = \overline{1,n}.$

3. $\tilde{z}_{ij} = \frac{c_i r_j}{\sum_{i=1}^{m} c_i} \geq 0$, since $c_i \geq 0, r_j \geq 0$, $i = \overline{1,m}$, $j = \overline{1,n}$. The magnitude $\sum_{i=1}^{m} c_i > 0$ by assumption, i.e., total capacity of all the storages is non-zero.

Hence, the system (2) – (4) has the feasible point (14), so it is consistent.

If the inequality constraints (2) of the problem (1) – (4) are transformed into equality constraints by introducing additional variables, then the constraints system (2) – (4) is non-degenerate. However, if the equality $\sum_{i=1}^{m} c_i = \sum_{j=1}^{n} r_j$ holds, then the coefficients matrix of the basis variables is not degenerate, but it may contain zero coefficients for additional variables. Such a situation can impair work of methods based on using basis matrix, for example, the simplex method. If the inequality $\sum_{i=1}^{m} c_i > \sum_{j=1}^{n} r_j$ holds, this situation is unlikely since basis variables coefficients are non-zero. Therefore, in order to improve performance of methods, which are working with basis matrix, it is advisable to ensure that the condition $\sum_{i=1}^{m} c_i > \sum_{j=1}^{n} r_j$ is fulfilled. To demonstrate this effect, let us consider the following example.

Example 1. Let us consider the problem (1) – (4) with $m = 4$ and $n = 24$, i.e. there are 4 storages and 24 markets. Each market needs 10 units of products, each storage contains 40 units of products. The location of the markets with the specified coordinates is shown in Figure 2 (marked in blue). For such a problem, there is an analytical solution, according to which the optimal locations of the storages have the following coordinates: $(\mathbf{x}^*, \mathbf{y}^*) = \big((50{,}50), (50{,}250), (50{,}450), (250{,}50), (250{,}250), (250{,}450)\big)$, marked with orange crosses in Figure 2. Optimal value of the function $f^*(\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*)$ for such a configuration of the problem equals $16970{,}6$.

To solve the problem the MINOS 5.51 solver [11] from NEOS server [12] is used. To formulate the problem the AMPL language [13] is used. The initial data are as follows: $c_i = 40$, $i = \overline{1,m}$, $r_j = 10$, $j = \overline{1,n}$. The starting point $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ for the MINOS solver is obtained using pseudorandom number generator. The problem is solved in two variants: when the condition $\sum_{i=1}^{m} c_i = \sum_{j=1}^{n} r_j$ is fulfilled and the condition $\sum_{i=1}^{m} c_i > \sum_{j=1}^{n} r_j$ is true. In the first case $c_i = 40$, $i = \overline{1,m}$, in the second case – $c_i = 40.4$, $i = \overline{1,m}$, i.e., each storage capacities are increased in 1 % to ensure that the inequality $\sum_{i=1}^{m} c_i > \sum_{j=1}^{n} r_j$ is fulfilled. The results of problem solving using the MINOS 5.51 solver are given in Table 1. Here $\hat{f}^*(\mathbf{x}, \mathbf{y}, \mathbf{z})$ is the objective function value, obtained with the solver, $\Delta = \big(\hat{f}^*(\mathbf{x}, \mathbf{y}, \mathbf{z}) - f^*(\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*)\big)/f^*(\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*)$ – relative error of the optimal value of the function. Table 1 shows that MINOS successfully solved the second variant of the problem and does not solve the first variant, since the magnitude Δ is almost 9 % for it. It is important to emphasize that to solve the second variant MINOS used more than twice less iterations comparing to the first variant.

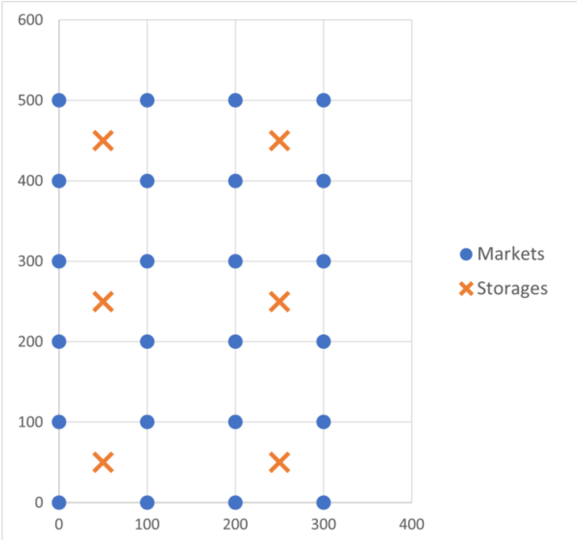

Figure 2: storage location problem for $m = 4$ and $n = 24$

Table 1
The results of solving two variants of the problem using the MINOS 5.51 solver

	First variant	**Second variant**
The number of iterations	368	163
Solving time (sec)	0.016978	0.004625
$\hat{f}^*(\mathbf{x}, \mathbf{y}, \mathbf{z})$	18472.1	16970.6
Δ	0.088481	4.2874e-16

5. Modification of the problem (1) – (4) for equality constraints

Similarly to closed transportation problems the problem (1) – (4) can be reformulated using equality constraints:

$$f(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{i=1}^{m} \sum_{j=1}^{n} z_{ij} \sqrt{(x_i - a_j)^2 + (y_i - b_j)^2} \to \min \qquad (15)$$

subject to

$$\sum_{j=1}^{n} z_{ij} = c_i, \quad i = \overline{1, m}, \qquad (16)$$

$$\sum_{i=1}^{m} z_{ij} = r_j, \quad j = \overline{1, n}, \qquad (17)$$

$$z_{ij} \geq 0, \quad i = \overline{1, m}, \quad j = \overline{1, n}. \qquad (18)$$

Constraints system of the problem (15) – (18) contains $m + n$ equality constraints and is linear dependent. The problem of that kind we will call degenerate problem. For constraints system of the problem (15) – (18) the following lemma is true.

Lemma 3. Constraints system (16) – (18) contains $m + n - 1$ linear independent equations.

Proof. The statement of the lemma is proved in the book [14, p. 198].

Degeneracy of the system (16) – (18) can significantly affect the process of solving the problem (15) – (18) using simplex-type methods, which are based on transition from one basis matrix to another. To overcome the degeneracy of the system (16) – (18), it is advisable to exclude one arbitrary linearly dependent constraint from it. The choice of such a constraint will affect the convergence rate of the method. Let us demonstrate it with the following example.

Example 2. Let us consider the problem (15) – (18) for $m = 3$, $n = 12$. Markets locations with known coordinates are given in Figure 3 (marked in blue).

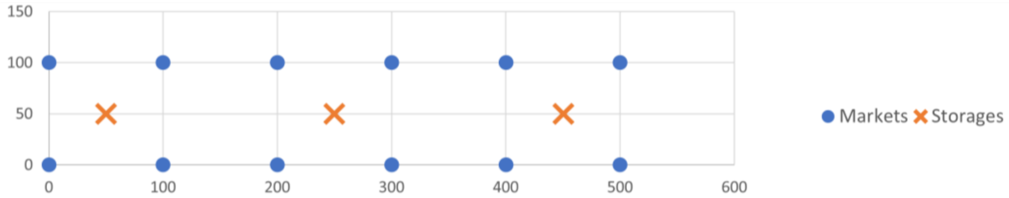

Figure 3: storage location problem for $m = 3$ and $n = 12$

We will solve the problem with different options for extracting one linearly dependent constraint for storages from the constraints system of the problem. For this, we will build four problems. Problem A is the problem (15) – (18). Problems B, C, and D are the problem A, with the first, second, and third constraints removed from the constraints group (16) respectively.

Storage capacities and demand of each market are selected similarly to the previous example, i.e. $c_i = 40,\ i = \overline{1,m},\ r_j = 10,\ j = \overline{1,n}$. For such input data of the problem, the optimal location of the storages is shown in Figure 3 (marked with orange crosses). Herewith, the optimal value of the objective function $f^*(\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*) = 8485{,}28$. The starting point $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ for solvers is obtained using pseudorandom number generator.

Results of the MINOS solver work for the problems A, B, C, D are given in Table 2. Here **iter** is the number of iteration performed be the solver; **obj** is the number of the objective function value calculations; **grad** is the number of gradient calculations; $\mathbf{\Delta} = \left(\hat{\boldsymbol{f}}^*(\mathbf{x}, \mathbf{y}, \mathbf{z}) - \boldsymbol{f}^*(\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*)\right) / \boldsymbol{f}^*(\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*)$ is relative error of the objective function value for the solution obtained by the solver; **time** is the time of solving the problem by the solver in seconds.

Table 2

Results of solving the problems A, B, C, D using the MINOS 5.51 solver

	Problem A	**Problem B**	**Problem C**	**Problem D**
iter	47	46	33	41
obj	45	40	24	42
grad	44	39	23	41
Δ	8.36043e-15	2.1437e-16	1.02898e-14	5.14488e-15
Solving time (sec)	0.00248	0.002263	0.002152	0.002174

Table 2 results demonstrate that the MINOS 5.51 solver successfully solved all four problems: the largest error is achieved in problem C and is 10^{-14}. MINOS spent the most computational resources on solving the degenerate problem, namely 47 iterations, 45 calculations of the objective function values, and 44 calculations of its gradient. For all non-degenerate problems, MINOS consumed less computational resources. The best result is achieved when solving problem C: 33 iterations, 24 calculations of the objective function value, and 23 calculations of the gradient. Let us solve problems A, B, C, D using the following list of solvers from the "Nonlinearly Constrained Optimization" section on NEOS server: Knitro 13.2.0, SNOPT 7.6.1, CONOPT 3.17A, LANCELOT, filterSQP (20020316), Ipopt 3.14.12, LOQO 7.00, OCTERACT Engine 4.4.0. Relative errors of the objective function value for the solution obtained by the solvers are given in Table 3. Table 3 results show that only Knitro, SNOPT, filter, Ipopt, and LOQO solvers successfully solved all problems, and filter shows the smallest relative error Δ among them. The filter solver is the only solver from the list that showed better accuracy than MINOS (see Table 2). When solving the degenerate problem A, Knitro printed messages *suffix feaserror OUT; suffix opterror OUT; suffix numfcevals OUT; suffix numiters OUT*, and Ipopt printed messages *suffix ipopt_zU_out OUT; suffix ipopt_zL_out OUT*. Such messages indicate problems related to degeneracy of the problem constraint system.

CONOPT and LANCELOT solvers were able to solve only problem D and problem C, respectively. This demonstrates expediency of excluding one linearly dependent constraint to overcome degeneracy of the problem. When solving problems A, B and D using LANCELOT, the maximum number of iterations (1000 iterations) was exceeded, so no solution was found. When solving problems A, B, CONOPT displayed message *Evaluation error limit; 2 failed evaluations*, which indicates problems with evaluating the value of the objective function. OCTERACT could not solve the problem in 5 minutes, so it was not included in the table.

Table 3
Accuracy of solving problems A, B, C, D using NEOS server solvers

	Problem A	Problem B	Problem C	Problem D
	Δ	Δ	Δ	Δ
Knitro	1.32e-10	3.13e-8	3.13e-8	3.13e-8
SNOPT	1.25e-12	1.08e-12	7.24e-13	1.08e-12
CONOPT	0.2518	0.1905	0.2761	2.14e-16
LANCELOT	-0.5460	-0.7334	-4.94e-12	-0.45
filter	0.0	2.14e-16	0.0	0.0
Ipopt	-5.65e-9	-5.65e-9	-5.65e-9	-5.65e-9
LOQO	6.77e-9	1.81e-9	1.72e-8	8.49e-9

6. Computational experiment for the problem (1) – (4): $m = 5, n = 19$

Example 3. Let us consider the problem (1) – (4) for location 5 storages for products transportation to 19 the most common markets in Kyiv (see Figure 4). For this, on the map of Kyiv with its surroundings a coordinate grid is superimposed containing 120 squares of size 10×10 with a maximum coordinate of 100 along the abscissa axis and 120 along the ordinate axis. The origin of the coordinates is in the lower left corner. Markets coordinates were determined visually using the grid constructed. Table 4 shows names and coordinates of 19 markets obtained under this scheme.

Table 4
Model coordinates of the most common Kyiv markets

#	Market Name	Coordinates	#	Market Name	Coordinates
1	Illis	(18,81)	11	Zhytnii	(46,82)
2	Shpalernyi	(21,74)	12	Hospodarskyi	(46,88)
3	Borshchahivskyi	(22,71)	13	Volodymyrskyi	(46,67)
4	Rechovyi	(26,79)	14	Bessarabskyi	(48,75)
5	Stolychnyi	(33,56)	15	Ovochevyi	(64,79)
6	Sevastopolskyi	(37,69)	16	Troieshchynskyi	(63,92)
7	Kurenivskyi	(38,93)	17	Pecherskyi	(52,72)
8	Solomianskyi	(42,68)	18	Lisovyi	(69,81)
9	Lukianivskyi	(42,81)	19	Darnytskyi	(71,71)
10	Hurtovyi	(43,75)			

Storages capacities and demand of each market are chosen similarly to the previous examples 1 and 2, i.e., $c_i = 40,\ i = \overline{1,5},\ r_j = 10,\ j = \overline{1,19}$. Starting storages coordinates $(\mathbf{x}_0, \mathbf{y}_0)$ are determined visually. Starting products volumes for transportation $\boldsymbol{z}_0$ are obtained using a pseudorandom number generator on the interval $[0,20]$. For the input data objective function value equals 19218.92 and the inequality $\sum_{i=1}^{m} c_i > \sum_{j=1}^{n} r_j$ is fulfilled, since $\sum_{i=1}^{5} 40 = 200 > \sum_{j=1}^{19} 10 = 190$, so the problem is non-degenerate.

To solve the problem, NEOS server solvers from the "Nonlinearly Constrained Optimization" section with default parameters were used. Only the filter and Knitro solvers successfully solved the problem, while the solutions and the value of $\hat{f}^*(\mathbf{x}, \mathbf{y}, \mathbf{z})$ for both solvers coincide and are equal to 1015.9. All other solvers did not solve the problem and displayed messages indicating problems with solving the problem. In particular, CONOPT displayed the message *Evaluation Error Limit, 2 failed evaluations*, and Ipopt displayed message *Invalid number in NLP function or derivative detected. suffix ipopt_zU_out OUT; suffix ipopt_zL_out OUT*.

Figure 4 shows locations of 19 markets (marked in red), starting locations of storages (marked in yellow), and locations of storages obtained using the filter and Knitro solvers (marked in black).

Figure 4 shows that three clusters can be distinguished in markets location: the western (Illis, Shpalernyi, Borshchahivskyi and Stolychnyi markets), the eastern (Troieshchynskyi, Lisovyi, Ovochevyi and Darnytskyi markets) and the central (the rest of the markets). As a result of solving the problem using filter and Knitro, three storages are located in the central cluster and one each in the western and eastern clusters. Figure 4 allows you to visually assess the connection between storage location problem and centroid-based clustering problem.

To test dependence of solvers work on choice of starting point, the problem was solved using filter and Knitro solvers with 5 different starting points obtained using a pseudorandom number generator. The results of filter and Knitro work for this case are shown in Table 5.

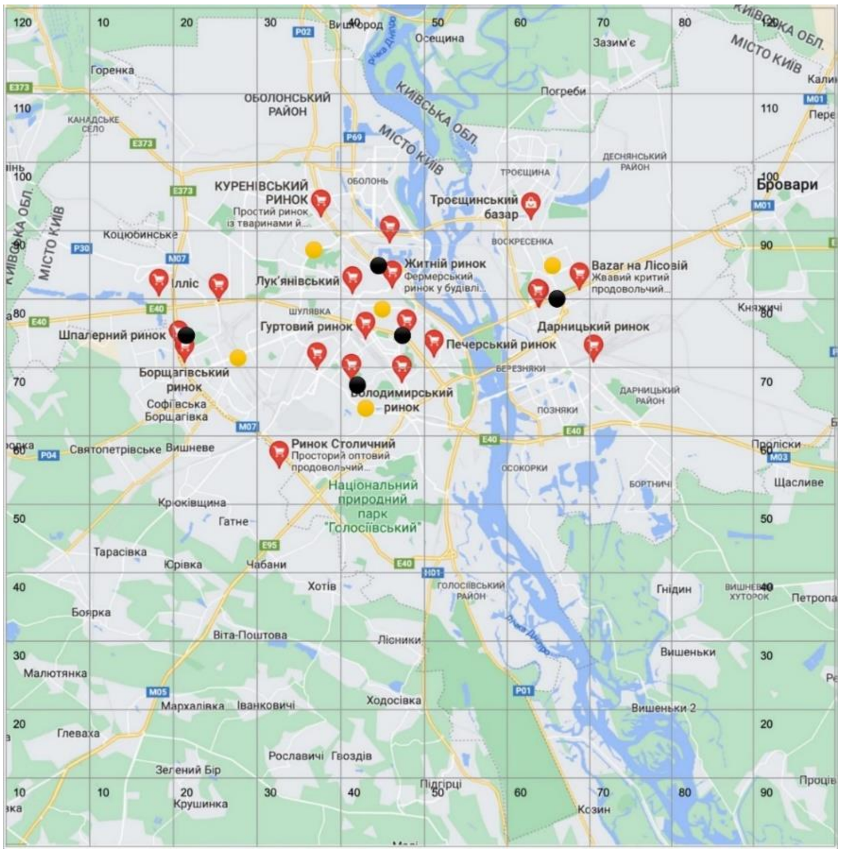

Figure 4: the best location of 5 storages and 19 markets: filter and Knitro

Table 5

Results of solving the problem of location 5 storages and 19 markets using filter та Knitro from different starting points

Starting point number		1	2	3	4	5	**6**
filter	$\hat{f}^*(\mathbf{x},\mathbf{y},\mathbf{z})$	1034.75	1041.33	1021.08	1016.29	1033.34	**1015.9**
	iter	56	84	68	62	78	**58**
Knitro	$\hat{f}^*(\mathbf{x},\mathbf{y},\mathbf{z})$	1015.96	1277.8	1016.33	1016.38	1017.07	**1015.94**
	iter	86	39	78	71	125	**77**

The first line of Table 5 shows the number of the starting point, the second and fourth lines show value of the objective function of the problem obtained by filter and Knitro, respectively, and the third and fifth lines show the number of iterations required by filter and Knitro, respectively. The sixth starting point was used in the previous calculation and is given above.

The results show that in 5 runs from different starting points, the filter and Knitro could not obtain a smaller value of the objective function than the value obtained by the solvers starting from the above point #6.

The solvers obtained the highest value of the objective function when starting from the starting point #2, and filter performed 84 iterations, Knitro – 39 iterations. Starting from the starting point #6, filter and Knitro performed 58 and 77 iterations, respectively, obtaining the same value of the objective function equal to 1015.9. This effect can be explained by the close connection of the storage location problem with centroid-based clustering problems, the objective functions of which are multi-extreme, and the solutions of the problem are its local minima.

7. Conclusions.

The article investigates nonlinear programming problem for the optimal location of storages so that the total distance, calculated with weighting coefficients equal to the volumes of products transported from storages to markets, is minimal. It is shown that the objective function of the problem in general case is a non-smooth non-convex function, and the solution of the problem is non-unique. The consistency conditions of the constraints system of this problem are substantiated and its options are considered depending on the balance conditions that determine degeneracy and non-degeneracy of constraints system. The statement of Lemma 1 can serve as a tool for choosing a starting point. Research in this direction is currently underway.

Three examples of solving the storage location problem using NEOS server solvers from the “Nonlinearly Constrained Optimization” section are considered. The first example shows that the MINOS solver does not solve a problem with a degenerate constraint system of the problem defined by the balance condition and solves a problem with a non-degenerate constraint system. The second example shows that exclusion of one arbitrary linearly dependent constraint from the problem constraint system allows it to be solved faster than a degenerate problem. The third example is related to the optimal location of 5 storages for products transportation to 19 the most common markets in Kyiv. The results of solving this problem by solvers filter and Knitro show that the solution they found depends on the choice of the starting point.

8. Acknowledgements.

The authors are pleased to acknowledge the support by the Volkswagen Foundation under grant number 97 775 and by the project of research works of young scientists №07-02/03-2023/BM 120.34.

9. Reference

[1] R. Zanjirani Farahani, M. Hekmatfar, Facility Location: Concepts, Models, Algorithms and Case Studies, Springe-Verlag Berlin Heidelberg, 2009.
[2] R. Zanjirani Farahani, M. Hekmatfar, B. Fahimnia, N. Kazemzadeh, Hierarchical Facility Location Problem: Models, Classifications, Techniques, and Applications, Computers & Industrial Engineering 68 (2014) 104–117.
[3] Z. Drezner, H.W. Hamacher, Facility Location: Application and Theory, Berlin, Springer, 2001.
[4] L.-Y. Wu, X.-S. Zhang, J.-L. Zhang, Capacitated Facility Location Problem with General Setup Cost, Computers & Operations Research 33.5 (2006) 1226–1241.
[5] K. Jain, M. Mahdian, A. Saberi, A new greedy approach for facility location problems in Proceedings of the thirty-fourth annual ACM symposium on Theory of computing (STOC '02) (2002), Association for Computing Machinery, New York, USA.
[6] V.S. Mikhalevych, V.A. Trubin, N.Z. Shor, Optimization problems of production and transport planning: Models, methods and algorithms, Nauka, 1986.
[7] G.D. Bila, O.O. Korchynsky, P.I. Stetsyuk, O.M. Khomiak, S.B. Shekhovtsov, Using NEOS server for solving two classes of optimization problems, Cybernetics and Computer Technologies 4 (2022) 56–81.
[8] C. Ortiz-Astorquiza, I. Contreras, G. Laporte, Multi-level facility location problems, European Journal of Operational Research 267:3 (2018) 791–805.
[9] E.M. Kiseleva, N.Z. Shor, Continuous problems of optimal set partition: theory, algorithms, applications, Naukova dumka, 2005.
[10] A.F. Gametsky. D.I Solomon, Operational Research, Part II, Academy of Economical Knowledge of Moldova, Academy of Transport, Informatics and Communications, Evrica, 2008.
[11] MINOS. https://neos-server.org/neos/solvers/nco:MINOS/AMPL.html
[12] NEOS Solver. https://neos-server.org/
[13] AMPL – Optimizing the World’s Most Complex Tasks. https://ampl.com/
[14] A.F. Gametsky. D.I Solomon, Operational Research, Part I, Academy of Economical Knowledge of Moldova, Academy of Transport, Informatics and Communications, Evrica, 2004.