Problem of Distribution of Goods by Logistics Centers

Anatoly Panyukov and Khalid Z. Chaloob

South Ural State University,
76 Lenina Ave., 454080, Chelyabinsk, Russia
paniukovav@susu.ru
khalid.e@mail.ru

Abstract. Effective logistics management is recognized as a key factor in improving the performance of companies and their competitiveness. The econometric methods used in practice do not provide the means for promptly solving a multitude of emerging problems, in particular for effective operational management of a network marketing organization. In the paper, algorithms for analyzing and solving the problem of distribution of goods by logistics centers, including decision support system in case of incorrectness of the arising problem are proposed: (1) the method of regularization of the decomposable distribution problem; (2) an effective algorithm for approximating an indecomposable problem by a decomposable problem. The software implementation of the proposed algorithms is easily encapsulated in the MS Office system.

Keywords: Logistics center · Transport problem · Operational management · Distribution task · Regularization · Decomposition · Algorithm

1 Introduction

A enterprise is a complex and dynamic system actively interacting with the external environment. Currently, effective logistics management is recognized as a key factor in improving the performance of companies and their competitiveness [9, 3]. Budashevsky, and Pastukhova [2] propose constructive comparative analysis of methods and models for estimating demand used in economics and marketing and system technology for analyzing and forecasting consumer preferences. Bayev and Drozin [1] consider questions of the dynamics of consumer demand. Levin notes [4] that these methods do not provide the means to quickly solve a lot of emerging problems, in particular for effective operational management of the network marketing organization.

Algorithms for analyzing and solving the problem of the distribution of goods by logistics centers, including a decision support system in case of illposed arising

In: S. Belim et al. (eds.): OPTA-SCL 2018, Omsk, Russia, published at http://ceur-ws.org

problem are proposed in the paper. Software implementation of these algorithms is easily encapsulated in the MS Office system [5, 6].

In the first section we give a formal statement of the problem and introduce the main notation used of this paper. In the second section we consider the decomposable case of a problem reducible to the transport problem with matrix formulation. In the third section we propose a method for regularizing a decomposable problem in the case of its illposed formulation. In the fourth section we propose a method for approximating the problem of the original problem with a decomposable problem.

2 Statement of the Problem

The problem of distributing a set I of goods over a set J of logistics centers is considered. Let x_{ij} be the volume of the commodity $i \in I$ distributed to the center $j \in J$. Let p_{ij} be the marginal profit from the sale of a unit of commodity $i \in I$ at the center $j \in J$. Let λ_{ij} be the cost of distribution of a unit of commodity $i \in I$ by the center $j \in J$. Let d_i be the effective demand for the commodity $i \in I$. Let b_j be the resource for the maintenance of the center $j \in J$. The formal formulation of the problem consists in finding the distribution of goods $i \in I$ at the centers $j \in J$, for which the marginal profit is maximal

$$x^o = \arg\max_{x \in D} \sum_{j \in J} \sum_{i \in I} p_{ij} x_{ij}, \tag{1}$$

all goods are satisfied with the effective demand

$$\sum_{j \in J} x_{ij} = d_i, \qquad i \in I, \tag{2}$$

resource constraints have been met for all centres

$$\sum_{i \in I} \lambda_{ij} x_{ij} \leq b_j, \qquad j \in J, \tag{3}$$

the condition of non-negativity is fulfilled

$$x_{ij} \geq 0, \qquad i \in I, \quad j \in J. \tag{4}$$

The problem (1)-(4) is the known distribution task of linear programming [7, 8]. In general, for this task is not known methods that take into account its specificity, so to solve it apply universal methods of linear programming. For large-scale tasks, this approach requires commercial software. In addition, if problem (1)-(4) has no solution, the principle of making an acceptable decision is not clear in this statement.

3 The Decomposable Case of Task (1)-(4)

For some cases, the parameter λ_{ij} can be represented as the product

$$\lambda_{ij} = \alpha_i \cdot \beta_j, \quad i \in I,\ j \in J \tag{5}$$

where α_i is the resource intensity of the commodity $i \in I$ in conventional units, β_j is the cost of servicing the conventional unit at the center $j \in J$. This case of the problem (1)-(4) is called as *decomposable problem.* It is possible decomposable problem (1)-(4) reducing to the matrix transportation problem. Indeed, for all $j \in J$ we have

$$\left(\sum_{i\in I} \lambda_{ij} x_{ij} \ \le b_j\right) \Leftrightarrow \left(\sum_{i\in I} \alpha_i \beta_j x_{ij} \le b_j\right) \Leftrightarrow \left(\sum_{i\in I} \alpha_i x_{ij} \le \frac{b_j}{\beta_j}\right) \Leftrightarrow \left(\sum_{i\in I} y_{ij} \le \frac{b_j}{\beta_j}\right),$$

here $y_{ij} = \alpha_i \cdot x_{ij}$ for all $i \in I,\ j \in J$. Passing to the variables $y_{ij} = \alpha_i \cdot x_{ij}$ for all $i \in I,\ j \in J$ in problem (1)-(4) get

$$\left(\sum_{j\in J} x_{ij} = d_i\right) \Leftrightarrow \left(\sum_{j\in J} y_{ij} = \frac{d_i}{\alpha_i}\right), \qquad i \in I,$$

$$\sum_{j\in J} \sum_{i\in I} p_{ij} x_{ij} = \sum_{j\in J} \sum_{i\in I} \frac{p_{ij} y_{ij}}{\alpha_i}.$$

Thus, the problem (1)-(4) is equivalent to the following one

$$y^o = \arg\max_{x\in D} \sum_{j\in J} \sum_{i\in I} \frac{p_{ij} y_{ij}}{\alpha_i}\ , \tag{6}$$

$$\sum_{j\in J} y_{ij} = \frac{d_i}{\alpha_i}, \qquad i \in I \tag{7}$$

$$\sum_{i\in I} y_{ij} \le \frac{b_j}{\beta_j}, \qquad j \in J\ , \tag{8}$$

$$y_{ij} \ge 0, \qquad i \in I,\ j \in J. \tag{9}$$

The problem (6)-(9) is open matrix transportation problem [7, 8]. The encapsulated in the MS Office system software to solve such problems of large dimension is known [6].

4 Regularization of Problems (1)-(4) and (6)-(9)

Problem (6)-(9) (hence and (1)-(4)) has a solution when demand does not exceed the supply, i.e.

$$S = \sum_{i \in I} \frac{d_i}{\alpha_i} - \sum_{j \in J} \frac{b_j}{\beta_j} \le 0.$$

Otherwise (that is, if $S > 0$), the problems (1)-(4) and (6)-(9) do not have admissible solutions, and it is required to correct the original problem to find a suitable solution. Possible ways to adjust the conditions of these tasks are:

1. To allow the supply of all goods below the effective demand

$$\sum_{j \in J} y_{ij} = \frac{d_i}{\alpha_i} - y_{i0}, \quad y_{i0} \ge 0, \qquad i \in I;$$

2. To develop the infrastructure of all routes in order to effectively support the effective demand

$$\sum_{i \in I} y_{ij} = \frac{b_j}{\beta_j} + y_{0j}, \quad y_{0j} \ge 0, \qquad j \in J,$$

   where y_{0j} is the amount of investment (in conventional units) in the development of the route $j \in J$.

Let k_i be the allowed fraction of the unsatisfied demand for the commodity $i \in I$ (i.e. $y_{i0} \le k_i d_i$). Let p_{0j} be the amount of investment required to expand the resources of the center $j \in J$ by one conditional unit.

Let us consider the corrected problem taking into account the variables and restrictions introduced in this section.

$$y^o = \arg\max_{y \in D} \sum_{j \in J} \left(\sum_{i \in I} \frac{p_{ij} y_{ij}}{\alpha_i} - p_{0j} y_{0j} \right), \tag{10}$$

$$\sum_{j \in J} y_{ij} + y_{i0} = \frac{d_i}{\alpha_i}, \qquad i \in I, \tag{11}$$

$$\sum_{i \in I} y_{ij} - y_{0j} = \frac{b_j}{\beta_j}, \qquad j \in J, \tag{12}$$

$$y_{i0} \le k_i d_i, \qquad i \in I, \tag{13}$$

$$y_{ij}, y_{i0}, y_{0j} \ge 0, \qquad i \in I, \quad j \in J. \tag{14}$$

The problem (10)-(14) is closed matrix transportation problem [7]. The encapsulated in the MS Office system software to solve such problems of large dimension is known [6].

Obviously, problem (10)-(14) has an optimal solution. The regularization is carried out by introducing additional endogenous variables $y_{i0}, y_{0j} \geq 0,\ i \in I,\ j \in J$, exogenous variables $k_i,\ i \in I$, and constants $p_{0j},\ j \in J$.

It is evident from the constructed model that at fixed prices maintaining effective demand leads to a decrease in marginal profit. Preservation of marginal profit requires an increase in the selling price of goods, which can lead to an irreversible decrease in demand and a decrease in marginal profit. Thus, the task of decision-making in conditions of risk and uncertainty arises. The controlled parameters in this case are the exogenous variables $k_i,\ i \in I$.

Let us use the marginal profit M and the amount of unmet demand S as criteria in the decision-making model. It is obvious that all solutions of problem (10)-(14) are Pareto optimal ones for fixed values of the exogenous variables $k_i,\ i \in I$. The problem of choosing specific values of these variables is difficult to formalize and requires the participation of the decision-maker (DM).

5 Approximation of the Problem (1)-(4) by the Decomposable Problem

As was noted above, problem (1)-(4) is decomposable if the equalities (5) hold. The value of the parameter λ_{ij} is interpreted as the cost of distributing a unit of commodity $i \in I$ by the center $j \in J$, and its value can be determined using statistical measurements. On the contrary, the parameters $\{\alpha_i > 0 :\ i \in I\}$, $\{\beta_j > 0 :\ j \in J\}$ are interpreted using the term "conventional unit", therefore their direct statistical measurement is impossible. Therefore, we consider the equalities (5) as a system of algebraic equations with unknowns $\{\alpha_i > 0 :\ i \in I\}$, $\{\beta_j > 0 :\ j \in J\}$. It is clear that for arbitrary values λ_{ij} the system of equations (5) may be inconsistent.

Let us introduce the function

$$F_\lambda(\alpha,\ \beta) = \sum_{i \in I,\, j \in J} \left| \log \frac{\alpha_i\ \beta_j}{\lambda_{ij}} \right|.$$

Obviously, $\inf F_\lambda = 0$ if and only if the system of equations (5) is consistent. It follows from the nonnegativity of the function $F_\lambda()$ that the value of $\inf F_\lambda$ can be considered as the degree of incompatibility of the system (5). Function $F_\lambda(),\ \lambda > 0$ is continuous in the neighborhood of any minimum, so the infimum is reached, and the optimal approximate solution of the system (5) with a minimal degree of incompatibility

$$(\alpha^o, \beta^o) = \arg \min_{\substack{\{\beta_j > 0:\ \ j \in J\} \\ \{\alpha_i :> 0:\ i \in I\}}} \left[\sum_{i \in I,\, j \in J} \left| \log \frac{\alpha_i\ \beta_j}{\lambda_{ij}} \right| \right]$$

can be considered.

It is easy to see that the optimality of the solution (α^o, β^o) implies the optimality of the solution set $D = \{(\alpha^o \cdot c,\, \beta^o / c):\ c > 0\}$. We shall assume that the solution of the approximating problem is

$$(\alpha^*, \beta^*) = \arg \min_{(\alpha,\beta)\in D} \|(\alpha, \beta)\|_\infty . \tag{15}$$

Note that if $(\alpha, \beta) \in D$ then

$$\alpha^* = \left\{ a_k \cdot \sqrt{\frac{\max_{j\in J} \beta_j}{\max_{i\in I} \alpha_i}} \right\}_{k\in I}, \qquad \beta^* = \left\{ \beta_k \cdot \sqrt{\frac{\max_{i\in I} \alpha_i}{\max_{j\in J} \beta_j}} \right\}_{k\in J}.$$

Thus, the correct formulation of the approximating problem is two-level, but for its solution it is sufficient to find any solution of the problem (1) of the lower level.

6 Algorithm to Solve Problem (15)

The following algorithm allows us to solve the problem (15).
Decomposition algorithm
Input: $I,\ J,\ \mathbf{\Lambda} = \{\lambda_{ij}\}_{i\in I,\, j\in J}$;
Output: $\alpha = \{\alpha_i\}_{i\in I}\,,\quad \beta = \{\beta_j\}_{j\in J}\,,\ F_\Lambda(\alpha,\, \beta)$;

Step 1. (Construct matrix $\hat{\mathbf{\Lambda}}$). For each line $i \in I$ of the matrix $\mathbf{\Lambda}$ execute steps 1.1, 1.2 and 1.3, then go to step 2.

Step 1.1. Build sorted line $i \in I$

$$\mathbf{\Lambda}\,[i] = \Big\{ \lambda^{(k)}_{ij^{(k)}}\ : \\ k = 1, 2, \ldots, |J|\,,\ j^{(k)} \in J,\ \lambda^{(1)}_{ij^{(1)}} \le \lambda^{(2)}_{ij^{(2)}} \le \ldots \le \lambda^{(|J|)}_{ij^{(|j|)}}, \Big\}.$$

Step 1.2. Let $k_- = \left\lfloor \frac{|J|+1}{2} \right\rfloor,\ k_+ = \left\lceil \frac{|J|+1}{2} \right\rceil,\ \alpha_i = \sqrt{\lambda^{(k_+)}_{ij^{(k_+)}} \cdot \lambda^{(k_-)}_{ij^{(k_-)}}}$.

Step 1.3. For $k = 1, 2, \ldots, |J|$ let $\hat{\lambda}^{(k)}_{ij^{(k)}} = \frac{\lambda^{(k)}_{ij^{(k)}}}{\alpha_i}$.

Step 2. (Construct matrix $\tilde{\mathbf{\Lambda}}$). For each column $j \in J$ of the matrix $\hat{\mathbf{\Lambda}}$ execute steps 2.1, 2.2 and 2.3, then go to step 3.

Step 2.1. Build sorted column $j \in J$

$$\hat{\mathbf{\Lambda}}\,[*]\,[j] = \Big\{ \hat{\lambda}^{(k)}_{i^{(k)}j}\ : \\ k = 1, 2, \ldots, |J|\,,\ j^{(k)} \in J,\ \hat{\lambda}^{(1)}_{i^{(1)}j} \le \hat{\lambda}^{(2)}_{i^{(2)}j} \le \ldots \le \hat{\lambda}^{(|I|)}_{i^{(|I|)}j} \Big\}.$$

Step 2.2. Let $k_- = \left\lfloor \frac{|I|+1}{2} \right\rfloor,\ k_+ = \left\lceil \frac{|I|+1}{2} \right\rceil,\ \ \beta_j = \sqrt{\hat{\lambda}^{(k_+)}_{i^{(k_+)}j} \cdot \hat{\lambda}^{(k_-)}_{i^{(k_-)}j}}$.

Step 2.3. For $k = 1, 2, \ldots, |I|$ let $\widetilde{l}^{(k)}_{i^{(k)}j} = \frac{\hat{\lambda}^{(k)}_{i^{(k)}j}}{\beta_j}$.

Step 3. (Normalization). Follow steps 3.1, 3.2, and 3.3, and then go to step 4.

Step 3.1. Let $c = \sqrt{\dfrac{\max_{i\in I} \alpha_i}{\max_{j\in J} \beta_j}}$.

Step 3.2. For all $i \in I$ let $\alpha_i = \alpha_i / c$.

Step 3.3. For all $j \in J$ let $\beta_j = \beta_j \cdot c$.

Step 4. Let $F_\Lambda(\alpha, \beta) = \sum_{i\in I,\, j\in J} \left|\log \frac{\alpha_i\ \beta_j}{\lambda_{ij}}\right|$.

Step 5. Return $\left\{\alpha = \{\alpha_i\}_{i\in I},\ \beta = \{\beta_j\}_{j\in J},\ F_\Lambda(\alpha,\ \beta)\right\}$.

End.

Theorem 1. Decomposition *algorithm correctly solves problem (15). Its algebraic computational complexity does not exceed the value* $O\left(|I| \cdot |J| \cdot \log\left(|I| \cdot |J|\right)\right)$.

Proof. Monotony of the logarithmic function implies

$$\left(\log \frac{\alpha_i\ \beta_j}{\lambda_{ij}} = 0\right) \Leftrightarrow \left(-\log \lambda_{ij} + \log \alpha_i + \ \log \beta_j = 0\right)$$
$$\Leftrightarrow \left(-l_{ij} + x_i - y_j = 0,\ l_{ij} = \log \lambda_{ij},\quad x_i = \log \alpha_i,\quad y_j = \ \log \beta_j\right),\quad i \in I,\ j \in J,$$

Therefore problem (15) is equivalent to problem

$$(x^o, y^o) = \arg \min_{\substack{x\in\mathbf{R}^{|\mathbf{I}|}\\ y\in\mathbf{R}^{|\mathbf{J}|}}} \sum_{i\in I,\, j\in J} |-l_{ij} + x_i - y_j| \tag{16}$$

and there is the one-to-one correspondence between the optimal solutions of these problems.

Problem (16) is equivalent to linear programming problem

$$\sum_{i\in I,\, j\in J} w_{ij} \ \rightarrow \ \min_{x,y,w}, \tag{17}$$

$$-w_{ij} \le -l_{ij} + x_i + y_j \le w_{ij}, \quad w_{ij} \ge 0, \qquad i \in I,\ j \in J. \tag{18}$$

Dual problem (17)-(18) is the problem

$$\sum_{i\in I,\, j\in J} l_{ij} \left(f_{ij} - f_{ji}\right) \ \rightarrow \ \max_f \ , \tag{19}$$

$$\sum_{j\in J} \left(f_{ij} - f_{ji}\right) = 0, \quad i \in I, \tag{20}$$

$$\sum_{i\in I} \left(f_{ij} - f_{ji}\right) = 0, \quad j \in J, \tag{21}$$

$$f_{ij} + f_{ji} \le 1, \quad f_{ij}, f_{ji} \ge 0, \qquad i \in I, \quad j \in J. \tag{22}$$

Making the change of variables $g_{ij} = f_{ij} - f_{ji},\ i \in I,\ j \in J$ in problem (19)-(22) we get the problem of maximum weight circulation

$$\sum_{i\in I,\ j\in J} l_{ij} g_{ij} \ \to \ \max_{g} \ , \tag{23}$$

$$\sum_{j\in J} g_{ij} = 0, \qquad i \in I, \tag{24}$$

$$\sum_{i\in I} g_{ij} = 0, \qquad j \in J, \tag{25}$$

$$-1 \le g_{ij} \le 1, \qquad i \in I, \quad j \in J. \tag{26}$$

Dual problem (23)-(26) is the problem

$$T_\Lambda(r,s,t) = \sum_{i\in I,\ j\in J} (t_{ij} + t_{ji}) \ \to \ \min_{r,s,t} \ , \tag{27}$$

$$r_i + s_j + t_{ij} - t_{ji} = \ l_{ij}, \quad t_{ij},\ t_{ji} \ge 0, \qquad i \in I, \quad j \in J. \tag{28}$$

Let us compare problem (17)-(18) constraints system and problem (27)-(28) constraints system. It is easy to see that the admissibility of the basic solution (r,s,t) of problem (27)-(28) implies the admissibility of solution

$$R = (x = r, \quad y = s, \quad w = \{w_{ij} = t_{ij} + t_{ji} \ : \ i \in I,\ j \in J\})$$

of problem (17)-(18). Moreover, if (r,s,t) is an optimal solution of problem (27)-(28), then R is an optimal solution of problem (17)-(18), since dual problems (19)-(22) and (23)-(26) have corresponding optimal solutions.

The one-to-one compilation of algorithm **Decomposition** in terms of values $(x,\ y,\ l)$) can be represented as following algorithm.

Log_Decomposition algorithm
Input: $I,\ J,\ \mathbf{L} = \{l_{ij} \ \ i \in I,\ j \in J\}$;
Output: $\mathbf{x} = \{x_i \ \ i \in I\}, \quad \mathbf{y} = \{y_j \ \ j \in J\}, \quad F_L(x,\ y)$;

Step 1. (Construct matrix $\hat{L}$). For each line $i \in I$ of the matrix $\mathbf{L}$ execute steps 1.1, 1.2 and 1.3, then go to step 2.

Step 1.1. Build sorted line $i \in I$

$$\mathbf{L}[i] = \Big\{ l_{ij^{(k)}}^{(k)} \ : \ k = 1,2,\ldots,|J|\,,\ j^{(k)} \in J,\ l_{ij^{(1)}}^{(1)} \le l_{ij^{(2)}}^{(2)} \le \ldots \le l_{ij^{(|j|)}}^{(|J|)}, \ \Big\}.$$

Step 1.2. Let $k_- = \left\lfloor \frac{|J|+1}{2} \right\rfloor,\ k_+ = \left\lceil \frac{|J|+1}{2} \right\rceil,\ r_i = \frac{l_{ij^{k_+)}}^{(k_+)} + l_{ij^{(k_-)}}^{(k_-)}}{2}$.

Step 1.3. For $k = 1,2,\ldots,|J|$ let $\hat{l}_{ij^{(k)}}^{(k)} = l_{ij^{(k)}}^{(k)} - r_i$.

Step 2. (Construct matrix $\tilde{\mathbf{L}}$). For each column $j \in J$ of the matrix $\hat{\mathbf{L}}$ execute steps 2.1, 2.2 and 2.3, then go to step AUX.

Step 2.1. Build sorted column $j \in J$

$$\hat{\mathbf{L}}\,[*]\,[j] = \Big\{\hat{l}^{(k)}_{i^{(k)}j}\ :\ k = 1, 2, \ldots, |J|\,,\ j^{(k)} \in J,\ \hat{l}^{(1)}_{i^{(1)}j} \le \hat{l}^{(2)}_{i^{(2)}j} \le \ldots \le \hat{l}^{(|I|)}_{i^{(|I|)}j}\Big\}.$$

Step 2.2. Let $k_- = \left\lfloor \frac{|I|+1}{2} \right\rfloor,\ k_+ = \left\lceil \frac{|I|+1}{2} \right\rceil,\ \ s_j = \frac{\hat{l}^{(k_+)}_{i^{(k_+)}j} + \hat{l}^{(k_-)}_{i^{(k_-)}j}}{2}.$

Step 2.3. For $k = 1, 2, \ldots, |I|$ let $\widetilde{l}^{(k)}_{i^{(k)}j} = \hat{l}^{(k)}_{i^{(k)}j} - s_j$.

Step AUX. (Construct matrix T) For all $i \in I$, $j \in J$ execute steps AUX.1 and AUX.2 then go to step 3.

Step AUX.1 If $\tilde{l}_{ij} > 0$, then put $t_{ij} = \tilde{l}_{ij}$, $t_{ji} = 0$, otherwise put $t_{ji} = -\tilde{l}_{ij}$, $t_{ij} = 0$.

Step AUX.2 Calculate

$$T_\Lambda(r, s, t) = \sum_{i \in I,\ j \in J} (t_{ij} + t_{ji})\,.$$

Step 3. (Normalization). Follow steps 3.1, 3.2, and 3.3, and then go to step 4.

Step 3.1. Let

$$c = \frac{\max_{i \in I} x_i - \max_{j \in J} y_j}{2}.$$

Step 3.2. For all $i \in I$ let $r_i = r_i - c$.

Step 3.3. For all $j \in J$ let $s_j = s_j + c$.

Step 4. Return $\Big\{\mathbf{x} = \{r_i\}_{i \in I}\,,\ \mathbf{y} = \{s_j\}_{j \in J}\,,\ T_\Lambda(r, s, t)\Big\}.$

End.

Here an auxiliary step AUX is introduced to prove the effectiveness of the algorithms. The optimal values of the variables t_{ij}, t_{ji}, $i \in I$, $j \in J$ and the optimal value $T_\Lambda(r, s, t)$ for the problem (27)-(28) are calculated at step AUX.

It is obvious that the solution (r, s, t), constructed by **Log_Decomposition** algorithm, meets all constraints of the problem (27)-(28). On the other hand problem (23)-(26) has an integral-valued optimal solution [10], i.. in the optimal solution g for all $i \in I$, $j \in J$ there is an inclusion $g_{ij} \in \{-1, 0, 1\}$. Let us to construct the solution $\{g_i j\, i \in I,\ j \in J\}$ of problem (23)-(26) using the conditions of complimentary with respect to solutions (r,s,t) of the dual problem (27)-(28), i.e. put

(1) $g_{ij} = -1$ for all $i \in I$, $j \in J$ such that $t_{ij} = 0$, $t_{ji} > 0$;
(2) $g_{ij} = 1$ for all $i \in I$, $j \in J$ such that $t_{ij} > 0$, $t_{ji} = 0$;
(3) $g_{ij} = 0$ for all $i \in I$, $j \in J$ such that $t_{ij} = t_{ji} = 0$.

It is easy to see that the constructed solution $\{g_{ij}\ i \in I,\ j \in J\}$ is an admissible solution of problem (23)-(26). It follows form the second duality theorem in linear programming that (r, s, t) and $\{g_{ij}\ i \in I,\ j \in J\}$ are optimal solutions of the problems (23)-(26) and (27)-(28) correspondingly. The effectiveness of **Log_Decomposition** algorithm, hence **Decomposition** algorithm, is proved.

Let us to turn to the estimation of **Decomposition** algorithm computational complexity. The body of Step 1 contains the sorting of the elements of the row (computational complexity $O(|J|\log|J|)$) and recalculation of its elements (computational complexity $O(|J|)$). Consequently, computational complexity of Step 1 does not exceed values $O(|I||J|\log(|J|)\ )$. Analogous arguments for Step 2 lead to the validity of the assertion that computational complexity of Step 2 does not exceed $O(|I||J|\log(|I|)\ )$. Therefore, the total computational complexity of Steps 1 and 2 will not exceed the values $O(|I||J|\log(|I||J|)\ )$. Step 3 consists of computation of value c and recalculation of values $\alpha = \{\alpha_i\ \ i \in I\}$, $\beta = \{\beta_j\ \ j \in J\}$, its computational complexity does not exceed $O(|I||J|)$. Step 4 consists of summing the $O(|I||J|)$ elements, its computational complexity does not exceed $O(|I||J|)$ as well. Thus, computational complexity of the algorithm does not exceed $O(|I||J|\log(|I||J|)\ )$. The theorem is proved.

7 Conclusion

The proposed algorithms solve the problems of analyzing the distribution of goods by logistics centers, including a decision support system in case of incorrectness of the task. Software implementation of these algorithms is easily encapsulated in the MS Office system.

References

1. Bayev, I.A., Drozin, D.A.: Dinamika pokupatel'skogo sprosa innovatsionnogo tovara (Dynamics of consumer demand for innovative goods). Bulletin of the South Ural State University. Ser. Economics and Management 8(2), 80–85 (2014), (in Russian)
2. Budashevsky, V.G., Pastukhova, O.N.: Tekhnologiya adaptivnogo upravleniya sinergeticheskim vzaimodeystviyem sprosa i predlozheniya na osnove proyektno-issledovatel'skogo kompleksnogo marketingovogo eksperimenta (Technology of adaptive management of the synergistic interaction of supply and demand on the basis of a project-research complex marketing experiment). Bulletin of the South Ural State University. Ser. Economics and Management 10(4), 60–65 (2016), (in Russian)
3. Galyautdinov, R.R.: The mechanisms of interaction of flows and stocks at enterprise from the perspective of logistics. Bulletin of the South Ural State University. Ser. Economics and Management 10(1), 157–163 (2016)
4. Levina, A.L.: Klassifikatsiya predpriyatiy roznichnoy torgovli s uchetom priznakov logisticheskoy integratsii (Classification of retail enterprises taking into account the signs of logistical integration). Bulletin of the South Ural State University. Ser. Economics and Management 10(4), 170–175 (2016), (in Russian)
5. Panyukov, A., Teleghin, V.: Forming of discrete mechanical assembly production program. Journal of Computational and Engineering Mathematics 2(1), 57–64 (2015)

6. Panyukov, A., Telegin, V.: Tekhnika programmnoy realizatsii potokovykh algoritmov (Engineering software implementation flow algorithms). Vestnik Yuzhno-Ural'skogo gosudarstvennogo universiteta. Seriya: Matematicheskoe modelirovanie i programmirovanie (Bulletin of South Ural State University. Series: Mathematical modeling and programming) 27(127), 78–99 (2008), (in Russian)
7. Raskin, L., Kirichenko, I.: Mnogoindeksnyye Zadachi Lineynogo Programmirovaniya (Multi-index Problems of Linear Programming). Radio i svyaz', Moscow (1989), (in Russian)
8. Sira, O.V.: Raspredelitel'naya zadacha lineynogo programmirovaniya (Distribution linear programming problem). Systemy obrobky informatsiyi (Information Processing Systems) 2(109), 168–170 (2013), (in Russian)
9. Stok, J.R., Lambert, D.M.: Strategic Logistics Management. McGraw-Hill/Irwin (2000)
10. Yemelichev, V.A., Kovalev, M.M., Kravtsov, M.K.: Polytopes, Graphs and Optimization. Cambridge University Press, New York (1986)