Permutation-Matrix Approach to Optimal Linear Assignment Design

Olha Matsyi[a] and Oksana Pichugina[b]

[a] *Kharkiv National Automobile and Highway University, Str.Yaroslava Mudrogo 25, Kharkiv, 61002, Ukraine*
[b] *National Aerospace University "Kharkiv Aviation Institute", Chkalova Street 17, Kharkiv, 61070, Ukraine*

Abstract
The paper presents a new iterative approach to solving Linear Assignment problems (LAP) and finding a perfect matching in a weighted bipartite graph iteratively. For that, a new permutation-matrix model of optimal linear assignment is proposed, which allows recursively finding solutions on a set of augmenting paths built based on the current matching. The results can be combined with other methods for solving a LAP such as the Hungarian Algorithm and minimal cost method in order to find an optimum faster.
Keywords 1
Linear Assignment Problem; optimal assignment; matching; augmenting path; routes; permutation.

1. Introduction

Transport Logistics is a broad application domain for Decision Theory and Optimization Theory dealing with routings and scheduling. It is inevitably connected with optimal placement in space and time of discrete objects. Therefore, Combinatorial Optimization is utilized widely in Transport Logistics problems.

Among transport logistics problems are numerous models of optimizing closed routes (routing models), which contain some conditions and constraints inherent in the actual process of moving objects on a plane or in space. Therefore, routing problems are crucial in rational, from economic prospective to decision-making and accelerating transport operations and management.

Even the most complex routing problems have a lot in common with the classical Vehicle Routing Problem (VRP) formulated by Danzig and Ramser [1, 2] and extended in many sources [3, 4, 5, 6].

This paper is dedicated to a solution of one type of assignment problem (Linear Assignment Problem, LAP), which is, in turn, is closely related to the Salesman Problem (SP) of a formation of a close route of a minimum length in a graph. The SP is a classical NP-complete problem, while a LAP represents a narrow subclass of combinatorial optimization problems solvable for polynomial time. That is why it is highly perspective to find other approaches to a polynomial solution of a LAP and utilize it in effective metaheuristics for the SP.

2. Related work

Conventional methods for solving the Linear Assignment Problem, such as the Hungarian algorithm, Kahn-Munkres method, and potential method are based on different combinatorial

II International Scientific Symposium «Intelligent Solutions» IntSol-2021, September 28–30, 2021, Kyiv-Uzhhorod, Ukraine
EMAIL: olga.matsiy@gmail.com (Olha Matsyi); oksanapichugina1@gmail.com (Oksana Pichugina)
ORCID: 0000-0002-1350-9418 (Olha Matsyi); 0000-0002-7099-8967 (Oksana Pichugina)

optimization approaches. They are all polynomial and characterized by different time complexity, while the best one is $O\left(n^3\right)$, where n is an order of their cost matrices [7, 8, 9, 10].

In [11], an algorithm for solving one version of the LAP is presented, which complexity has been reduced to $O\left(n^2\right)$. It was shown that the algorithm plays a role of procedure functions that can be embedded into the Branch and Bound method for solving LAPs. It resulted in faster than before calculating tighter lower bounds on the cost of closed routes in TSP.

This algorithm is designed to find a perfect matching of the minimum total weight in a weighted bipartite graph on $2n$ vertices. It utilizes an introduced concept of the shortest augmenting path in a graph [11, 12].

One of the common TSP settings is that the distances d_{ij} between each pair of cities i and j $i, j \in \{1,\ 2,\ \ldots\ ,n\}$ are known, and it is required to find such a sequence of the cities $(\pi[1],\ \pi[2],\ \ldots,\ \pi[i],\ \ldots,\ \pi[n])$ minimizing the value

$$\sum_{i=1}^{n-1} d_{\pi[i]\pi[i+1]} + d_{\pi[n]\pi[1]}\ . \tag{1}$$

This value is equal to the length of the shortest route (bypass), starting in a city $\pi[1]$, passing through all cities in turn and ending in $\pi[1]$ after visiting $\pi[n]$. The TSP, in which $d_{ij} = d_{ji}$ for each pair $\{i, j\}$ of cities is called symmetric (STSP) [13, 14, 15].

TSP and STSP are strongly NP-complete problems. They belong to a class of combinatorial optimization problems and, reflecting a continuously growing set of applications and generalizations, remains a topical research topic [15, 16, 17, 18].

Suppose that, to each edge, it is assigned zero weight in a complete graph on $n+1$vertices (thus reflecting that all delivery routes are of the same cost), but there is a fee for using each vehicle unit. This fee is fixed for all vehicles of the same capacity. Here the task is to find the minimum number of cars that will transport n cargo $\{d_{ii}\}_i$.

This problem, known as the container packing problem, is NP-complete in the strong sense. Since VRP includes the TSP and packing problem conditions, there is unlikely to solve the VRP exactly by efficient algorithms [15]. Besides, fulfilling a constraint $\sum_{i=1}^{n} d_{ii} \le K \cdot S$ for a given $K > 2$ and a container capacity S is not a sufficient condition for the existence of a feasible VRP solution.

The VPR is representable as a TSP with constraints. Among the conditions is the one that there is a vehicle initially located in the depot. The vehicle must deliver a homogeneous cargo from production points to consumption ones and then return to the depot. The total number of points of production and consumption is n; they form a set $N = \{1,\ 2,\ \ldots,\ n\}$, while the depo (the base) is assigned an index 0. The cost of transporting cargo from point i to point j ($i, j \in \{0\} \cup N$), the vehicle capacity is equal to S, the weight q_i of the load that must be delivered back from the point of production if $q_i < 0$ or delivered to the point of consumption in case of $q_i > 0$. Also, the balance condition $\sum_{i=1}^{n} q_i = 0$ has to satisfy.

It is required to find a permutation $(\pi[1],\ \pi[2],\ \ldots,\ \pi[i],\ \ldots,\ \pi[n])$ on set N such that

$$0 \le \sum_{i=1}^{u} q_{\pi[i]} \le S,\ u \in N, \tag{2}$$

$$d_{0,\pi[1]} + \sum_{i=1}^{n-1} d_{\pi[i]\pi[i+1]} + d_{\pi[n],0} \to \min \tag{3}$$

It follows from expressions (1) and (3) that the formulated problem is the classical TSP, in which the set of feasible solutions satisfying condition (2) may be empty. For example, it has no solution $(0,\ \pi[1],\ \pi[2],\ \ldots,\ \pi[i],\ \ldots,\ \pi[n],\ 0)$ for S = 1, n = 5, $q_i = 2/3$ for suppliers i = 1, 2, 3 and $q_i = -1$ for consumers $i = 4, 5$ [1].

Step one

$$d_{i^1 j^1} = \min\left\{d_{i_1 j_1},\ d_{i_1 j_2},\ d_{i_2 j_1},\ d_{i_2 j_2}\right\}.$$

After solving the TSP with the matrix $\left[d_{ij}\right]_{ij}$, one must return to the original transport network and add all the arcs to the resulting bypass.

For the VRP, several applied versions are known in the literature. These include, for example, the School Bus Routing Problem (SBRP), which has the following formulation. A school has a fleet of identical vehicles of capacity *S*, designed to deliver each student i to his residence after classes. The school has order number *0*. The travel time t_{ij} from point i to point j is known, $i, j \in \{0\} \cup N$, and the cost of the travel d_{ij} is known as well. Also, there is a requirement that each vehicle must return to point 0 no late than at time T [1].

In SBRP, it is required to find Boolean variables x_{ij}, $i, j \in \{0\} \cup N$, and such a number K of the vehicles satisfying the following constraints: the point 0 is the beginning and end of a route of each vehicle,

$$\sum_{i=1}^{n} x_{i0} = \sum_{i=1}^{n} x_{0i} = k\ , \tag{4}$$

any delivery point i is included in a single route:

$$\sum_{i=1}^{n} x_{ij} = \sum_{i=1}^{n} x_{ji} = 1;\quad j \in N; \tag{5}$$

there are no routes that include only delivery points:

$$\sum_{\substack{i, j \in U \\ U \subset N}} x_{ij} < |U|; \tag{6}$$

the route $(0,\ i[1],\ i[2],\ \ldots,\ i[j],\ \ldots,\ i[r],\ 0)$, $i[j] \in N$ of the vehicle satisfies a capacity condition:

$$\sum_{j=1}^{\tau-1} x_{i[j], i[j+1]} = \tau - 1 \le S - 1\ . \tag{7}$$

Also, there is a time limit T for the route execution:

$$t_{0i[1]} + \sum_{j=1}^{r-1} t_{i[j], i[j+1]} + t_{i[\tau]0} \le T. \tag{8}$$

The SBRP objective function is:

$$\sum_{i, j=0}^{n} d_{ij} x_{ij} \to \min. \tag{9}$$

It is easy to see that, in the SBRP $d_{ii} = 1,\ i = \overline{1, n}$ instead of $d_{ii} \in Z^{+}$ in the VRP, and the number of vehicles is $K = \lceil n/S \rceil$.

If, in the SBRP, the cost d_{ij} and time t_{ij} of moving the vehicle from a point i to a point j are linearly dependent, and $d_{ij} = 0$ if $t_{ij} = 0$. (9) can be replaced by an objective function

$$\sum_{i,j=0}^{n} t_{ij} x_{ij} \to \min \tag{10}$$

utilizing the initial data d_{ij}, $i, j \in \{0\} \cup N$ as auxiliary data for the economic assessment of the constructed solution.

The requirement $d_{ii} = 1$ for $i = \overline{1,n}$ makes a search of (9) much easier. The constraint on the vehicle load takes the form of inequality $k \le \lceil n / S \rceil$. The latter is a necessary and sufficient condition for a feasible solution to the problem.

If we substitute $r = n$ in (7) and (8), the SBRP becomes the TSP on a set of vertices $\{0\} \cup N,\ |N| = n$ of a transport network represented by the full graph.

The k-VRP problem is closely related to the VRP. In contrast to the VRP, the *k*-VRP does not specify the amount d_{ii} of cargo delivered to the *i*-th consumer ($i = \overline{1,n}$) and the capacity *S* of each vehicle, but it is required to serve at most *k* clients. It is necessary to minimize the total cost of routes of all vehicles, the number of which is equal to $m \le \lceil n / k \rceil$. Therefore, the *k*-VRP is solvable for *n*, $k \in Z^{+}$ and $n \ge k$. For $n = k$, it is the TSP defined on a set of permutations induced by $(0,\ i[1],\ i[2],\ \ldots,\ i[j],\ \ldots,\ i[n],\ 0)$. In particular, for *k = 2,* it is polynomially solvable, while for $k \ge 3$, it belongs to the class of NP-complete problems [15, 16, 19, 20, 21].

A peculiarity common for all the above routing problems is that they are formulated as generalizations or variants of the NP-complete problem TSP, where additional constraints are added. These constraints naturally make narrower the TSP feasible domain. The constraints lead to the problems' potential infeasibility, stimulating constant interest in further studying combinatorial optimization problems related to the TSP. In this paper, a new mathematical model of optimal assignment is described, which develops the results of [11, 12, 22, 23, 24, 25].

The paper goal is to build a permutation-matrix model of optimal assignment, which allows recursively finding solutions on a set of augmenting paths built from the current matching.

3. Main part

Let us describe the method for solving the LAP. We will use the following its formulation.For the matrix of costs (weights) $C = \left[c_{ij}\right]_{ij}$ of order n, where $c_{ij} \in R_{+}^{1}$ or $c_{ij} = +\infty$, where R_{0}^{+} is a set of non-negative real numbers, find

$$C(\sigma) = \min_{\pi \in S_n} \sum_{i=1}^{n} c_{i,\pi[i]}. \tag{11}$$

Here $\pi = (\pi[1],\ \pi[2],\ \ldots,\ \pi[n])$ is a permutation of a set $\{1,\ 2,\ \ldots, n\}$ of the columns' indexes of the matrix *C*, S_n is the permutation group, and $\sigma = (\sigma[1],\ \sigma[2],\ \ldots,\ \sigma[n])$ is the optimal permutation (an optimum) corresponding to the objective function value $C(\sigma) = \sum_{i=1}^{n} c_{i\sigma[i]}$.

The LAP is feasible if $C(\sigma) < +\infty$. Respectively, $c_{\sigma[i]} < +\infty,\ i = \overline{1, n}$. Note that a LAP with a cost matrix containing elements $c_{ij} = \infty$ may be unfeasible. In this case, it is necessary to establish that the feasible domain of the problem is empty.

Further, we will assume that we deal with a feasible LAP.

The permutation $\pi = (\pi[1], \pi[2], ..., \pi[n])$, where $C(\pi) < +\infty$, respectively, $c_{\pi[i]} < +\infty \quad i = \overline{1, n}$, is called a feasible solution to the LAP.

The core of the Hungarian algorithm for LAPs is that the cost matrix is equivalently transformed first to make a significant number of its elements zero. The one applies a greedy search to find as many as possible independent zeros. After that, directly the algorithm is applied, increasing by exactly one the set of independent zeros. Like the Hungarian algorithm, our approach to finding the permutation σ sequentially increases by one on each iteration the number elements k, $k = \overline{1, n}$, of the sequence representing a certain part of a feasible solution (a partial solution) of a LAP.

Let us list some properties of this sequence and outline the way of its construction.

Any part of a feasible solution of a LAP consisting of k elements determines uniquely a submatrix $\left[c_{i_s j_t}\right]_{s,t}$ of the matrix C having the order k, such that

$$i_1 < i_2 < ... < i_s < ... < i_k,\ j_1 < j_2 < ... < j_t < ... < j_k.$$

Introduce a sequence

$$\pi_k = (\pi_k[i_1], \pi_k[i_2], ..., \pi_k[i_s], ..., \pi_k[i_k]),\ \pi_k[i_s] \in \{j_1, j_2, ..., j_t, ..., j_k\},$$

$k = \overline{1, n-1}$, such that:

a) it is a solution TO the LAP having the submatrix $\left[c_{i_s j_t}\right]_{s,t}$ as its cost matrix;

b) the cost of the assignment induced by the permutation π_k does not exceed the optimal value of a LAP induced by any submatrix of C having the order k.

Let us try to develop a conversion procedure of a sequence π_k into a sequence π_{k+1}. If such an efficient procedure exists for $k = \overline{0, n-1}$ and the LAP (11) is feasible, then it takes n steps to find $\sigma = \pi_n$. Let us find out how the sequences π_k, $k = \overline{1, n}$ may be constructed.

Step 1. The initial sequence $\pi_1 = (\pi_1[i_1])$ is trivially defined and is identical to the minimal cost greedy method to solve LAPs. We derive the minimum value of the matrix and make the corresponding initial assignment of the order 1: let $c_{lr} = \min\{c_{ij} \mid 1 \le i, j \le n\}$, respectively, $i_1 = l, \pi_1 = (\pi_1[i_1]), \pi_1[i_1] = r$.

Step 2. In order to find $\pi_2 = (\pi_2[i_1], \pi_2[i_2])$ from π_1, let us determine

$c_{ms} = \min\{c_{ij} \mid i \ne l,\ j \ne r\}$, $c_{lp} = \min\{c_{lj} \mid j \ne r\}$, $c_{vr} = \min\{c_{ir} \mid i \ne l\}$ (see Fig. 1).

			p	q		r		s	
m			c_{mp}					$\underline{c_{ms}}$	
l			$\underline{c_{lp}}$			$\underline{c_{lr}}$			
v						$\underline{c_{vr}}$		c_{vs}	
w				c_{wq}					

Figure 1. Matrix C

It is easy to see that if $c_{lr} + c_{ms} \le c_{lp} + c_{vr}$ (further Case 2.1), then conditions a) and b) are satisfied for a sequence π_2 with $i_1 = l,\ \pi_2[i_1] = r,\ i_2 = m,\ \pi_2[i_2] = s.$

It corresponds to a submatrix of the matrix shown in Fig. 1 depicted in Fig. 2.

$$\begin{array}{c|c|c|}
 & r & s \\ \hline
l & c_{lr} & \\ \hline
m & & c_{ms} \\ \hline
\end{array}$$

Figure 2. The C -submatrix with elements c_{lr} and c_{ms}

Otherwise, these conditions are satisfied by a sequence π_2 with elements $i_1 = l,\ \pi_2[i_1] = p,\ i_2 = v,\ \pi_2[i_2] = r$ (further Case 2.2). The corresponding submatrix is depicted in Fig. 3.

$$\begin{array}{c|c|c|}
 & p & r \\ \hline
l & c_{lp} & \\ \hline
v & & c_{vr} \\ \hline
\end{array}$$

Figure 3. The C -submatrix with elements c_{lp}, c_{lr} and c_{vr}

Step 3. Now, we transform the sequence π_2 into a sequence $\pi_3 = \left(\pi_3[i_1],\ \pi_3[i_2],\ \pi_3[i_3]\right)$.

If we deal with Case 2.1, then next we find $c_{wq} = \min\left\{c_{ij} \mid i \neq l, m;\ j \neq s, r\right\}$ (see Fig. 1) and the value

$$MIN1 = c_{lr} + c_{ms} + c_{wq}.$$

Note that $c_{wq} = c_{ms}$, if $\pi_2 = \left(\pi_2[l] = p, \pi_2[v] = r\right)$, i.e. we deal with Case 2.2. The transformation from π_2 into π_3 is the result of solving the following auxiliary problem. For rows l, m and columns r, s of the matrix C induced by the elements first two least elements c_{lr} and c_{ms}, it is required to find a triple of components with the minimum sum of their values. Any two triple elements must be placed in three different rows, particularly including l, m and three various columns including r, s ones.

If such a triple does not contain c_{lr}, but includes c_{ms}, then the sum of its elements is bounded from below by the value

$$S_1 = c_{vr} + c_{lp} + c_{ms},$$

where $c_{vr} = \min\left\{c_{ir} \mid i \neq l, m\right\}$, $c_{lp} = \min\left\{c_{lj} \mid j \neq r, s\right\}$ (further, Case 3.1).

Let the solution of the auxiliary problem be a triple, which includes c_{lr} and does not contain c_{ms}. Then a value

$$S_2 = c_{lr} + c_{mp} + c_{vs},$$

where $c_{mp} = \min\left\{c_{mj} \mid j \neq s, r\right\}$, $c_{vs} = \min\left\{c_{is} \mid i \neq l, m\right\}$ (further, Case 3.2).

Elements of a solution of the auxiliary problem that does not contain c_{lr} and c_{ms} define a value

$$S_3 = \min\left\{c_{lp} + c_{mr} + c_{ws},\ c_{mq} + c_{ls} + c_{vr}\right\}$$

(further, Case 3.3). Here, $c_{ws} = \min\left\{c_{is} \mid i \neq l, m\right\}$, $c_{mr} = \min\left\{c_{mj} \mid j \neq p, s\right\}$ (see Fig. 4), $c_{ls} = \min\left\{c_{is} \mid i \neq m, v\right\}$, $c_{mq} = \min\left\{c_{mj} \mid j \neq r, s\right\}$ (see Fig. 5).

Let

$$MIN2 = \min\left\{S_1, S_2, S_3\right\}.$$

It is clear that it corresponds to the sequence π_3 we are looking for if $MIN2 \le MIN1$, otherwise $\pi_3 = (\pi_3[l] = r,\ \pi_3[m] = s,\ \pi_3[w] = q)$ (further, Case 3.4).

		p	q		r		s	
m			c_{mq}				c_{ms}	
l					c_{lr}		c_{ls}	
v					c_{vr}			
w								

Figure 4. Elements of an auxiliary problem solution that does not contain c_{lr} and c_{ms} that defines S_3 given $c_{ws} = \min\{c_{is} \mid i \ne l, m\}$, $c_{mr} = \min\{c_{mj} \mid j \ne p, s\}$

		p	q		r			s	
m			c_{mq}					c_{ms}	
l					c_{lr}			c_{ls}	
v					c_{vr}				
w									

Figure 5. Elements of an auxiliary problem solution that does not contain c_{lr} and c_{ms} that defines S_3 given $c_{ls} = \min\{c_{is} \mid i \ne m, v\}$, $c_{mq} = \min\{c_{mj} \mid j \ne r, s\}$

Depending on which one of the Cases 3.1-3.4 corresponds to $\min\{MIN2, MIN2\}$, the sequence π_3 is set.

Step k. A sequence $\pi_k = (\pi_k[i_1], \pi_k[i_2], \ldots, \pi_k[i_s], \ldots, \pi_k[i_k])$ with the properties a) and b) generally is transformed into a sequence $\pi_{k+1} = (\pi_{k+1}[i_1],\ \pi_{k+1}[i_2], \ldots, \pi_{k+1}[i_r], \ldots, \pi_{k+1}[i_{k+1}])$ with the same properties as the following.

In the matrix C, it is found

$$c_{i_{k+1}, \pi_k[i_{k+1}]} = \min\{c_{ij} \mid i \ne i_1, i_2, \ldots, i_s, \ldots, i_k,\ j \ne \pi_k[i_1], \pi_k[i_2], \ldots, \pi_k[i_s], \ldots, \pi_k[i_k]\},$$

then a sequence $\pi_{k+1}^1 = \left(\pi_k,\ \pi_k[i_{k+1}]\right)$ (further, Case k.1) is formed and

$$MIN1 = \sum_{s=1}^{k} c_{i_s,\pi_k[i_s]} + c_{i_{k+1},\pi_k[i_{k+1}]}$$

is evaluated.

Next, the problem of finding $k+1$ elements is solved, for which in the matrix C the minimum sum of their values is attained. At the same time, they are located in different rows and columns, including all rows and columns with numbers specified by values $c_{\pi_k[i_1]}$, $c_{\pi_k[i_2]}$, ..., $c_{\pi_k[i_s]}$, ..., $c_{\pi_k[i_k]}$. This induces several cases Cases k.2-k.K and the values $S_2,...,S_K$. Then $MIN2 = \min\{S_2,...,S_K\}$ is evaluated, and the partial permutation π_{k+1}^2 of the order k, where the value is achieved, is derived. Finally, π_{k+1} is found by assigning $\pi_{k+1} = \pi_{k+1}^1$ in case of $MIN1 \le MIN2$. Otherwise, $\pi_{k+1} = \pi_{k+1}^2$.

In the same way, the iterative process continues until π_n will be found. Process terminates and $\sigma = \pi_n$ is set.

The complexity of our method for solving LAPs described above has complexity $O\left(n^3\right)$, the best known so far and coinciding with the improved version of the Hungarian algorithm. This means that these two can be combined in order to get a hybrid method working, n average, better than the methods itself.

4. Conclusion

A new permutation-matrix model of optimal assignment is proposed. It allows recursively finding solutions on a set of augmenting paths built with respect to the current matching. The proposed scheme for finding an optimal assignment underlies a method of solving a LAP where a solution to the problem is found exclusively employing matching theory for bipartite graphs.

The developed model for finding the optimal destination develops transport logistics' theory. It is focused on improving the organization of transportation in real-time and in real situations of vehicle traffic. Its implementation allows reducing the time and fuel consumption for carrying out transport works.

The method uses the well-known algorithm for finding maximum matchings in bipartite unweighted graphs, built according to a scheme that expands approaches to solving hard optimization problems.

5. References

[1] D. P. Bertsekas, Network Optimization: Continuous and Discrete Models, Athena Scientific, Belmont, Mass, 1998.

[2] E.M. Bronshtein, T.A. Zaiko, Deterministic optimization problems of transport logistics, Automation and telemechanics, 10 (20100 113-147.

[3] G. Cornuejols, F. Harche, Polyhedral study of the capacitated vehicle routing problem, Mathematical Programming, V. 60, 1–3 (1993) 21–52. doi: 10.1007/BF01580599.

[4] A. Langevin, D. Riopel (Ed.), Logistics Systems: Design and Optimization, Springer US, 2005. doi: 10.1007/b106452.

[5] J.-Y. Potvin, Vehicle Routing, in: C. A. Floudas and P. M. Pardalos (Ed.), Encyclopedia of Optimization, Springer US, 2008, pp. 4019–4022. doi: 10.1007/978-0-387-74759-0_702.

[6] A. V. Vural, B. Eksioglu, Vehicle routing problem with simultaneous pickups and deliveries Vehicle Routing Problem with Simultaneous Pickups and Deliveries, in: C. A. Floudas and P. M. Pardalos (Ed.), Encyclopedia of Optimization, Springer US, 2008, pp. 4022–4027. doi: 10.1007/978-0-387-74759-0_703.

[7] D.J. Little, K. Murty, D. Sweeney, K. Carell, Algorithm for solving the travelling salesman problem, Economics and Mathematical Methods, V. 1, 1 (1965) 90-107.

[8] C. Helmberg, The m-Cost ATSP, in: G. Cornuéjols, R. E. Burkard and G. J. Woeginger (Ed.), Integer Programming and Combinatorial Optimization, Springer, Berlin, Heidelberg, 1999, pp. 242–258. doi: 10.1007/3-540-48777-8_19.

[9] E. Mainika, Optimization algorithms on networks and graphs, Mir, Moscow, 1981.

[10] R. E. Burkard, E. Çela, Linear Assignment Problems and Extensions, in: D.-Z. Du and P. M. Pardalos (Ed.), Handbook of Combinatorial Optimization, Springer US, 1999, pp. 75–149. doi: 10.1007/978-1-4757-3023-4_2.

[11] A.V. Panishev, D.D. Plechisty, V.A. Shevchenko, Local search procedures in combinatorial algorithms for solving the general traveling salesman problem, Artificial Intelligence, 1 (2005) 465–470.

[12] O.B. Matsiy, A.V. Morozov, A.V. Panishev, The Recurrent Method to Solve the Assignment Problem. Cybern Syst Anal, 51, 2015, pp. 939–946. doi:10.1007/s10559-015-9786-x.

[13] S.M. Rezer, S.E. Lovetsky, I.I. Melamed, Mathematical methods of optimal planning in transport systems, Results of Science and Technology, series ”Organization of Transport Management”, 1990.

[14] E. X. Gimadi, A.V. Shahshneider, Approximate estimation algorithms for routing tasks at random inputs with a limited number of clients in each route, Automation and telemechanics, 2 (2012) 126–139.

[15] C. H. Papadimitriou, K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity. Dover Publications, Mineola, N.Y., 1998.

[16] M. R. Garey, D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-completeness, W.H. Freeman & Co Ltd, New York, 1979.

[17] L. Lovasz, M. D. Plummer, Matching Theory, Chelsea Pub Co., Providence, R.I, 2009.

[18] T.L. Saaty, Optimization in integers and related extremal problems: from a course given at the University of California, Los Angeles, and George Washington University, Washington DC, RWS Publications, 2014.

[19] J. Evans, Optimization Algorithms for Networks and Graphs, 2nd ed., CRC Press, 2017.

[20] K. Thulasiraman, S. Arumugam, A. Brandstädt, T. Nishizeki, Handbook of Graph Theory, Combinatorial Optimization, and Algorithms, CRC Press, 2016.

[21] A. Langevin and D. Riopel (Eds.), Logistics Systems: Design and Optimization, Springer US, 2005. doi: 10.1007/b106452.

[22] O. Pichugina and O. Matsyi, Boolean Satisfiability Problem: Discrete and Continuous Reformulations with Applications, in Proceedings of the 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), 2020, pp. 623–627. doi: 10.1109/TCSET49122.2020.235507.

[23] Y.G. Stoyan, S.V. Yakovlev, Theory and Methods of Euclidean Combinatorial Optimization: Current Status and Prospects. Cybern Syst Anal, 56 (2020) 366–379. https://doi.org/10.1007/s10559-020-00253-6.

[24] S. Yakovlev, O. Pichugina, On constrained optimization of polynomials on permutation set CEUR Workshop Proceedings, v. 2353, 2019, pp. 570–580.

[25] I. Averbakh, W. Yu, Multi-depot traveling salesmen location problems on networks with special structure, Ann Oper Res, V. 286, 1 (2020) 635–648. doi: 10.1007/s10479-018-2812-4.