ISSN 0005-1179, Automation and Remote Control, 2013, Vol. 74, No. 1, pp. 90–104. © Pleiades Publishing, Ltd., 2013.
Original Russian Text © L.G. Afraimovich, 2013, published in Avtomatika i Telemekhanika, 2013, No. 1, pp. 116–134.

Multiindex Transportation Problems with 2-Embedded Structure

L. G. Afraimovich

Nizhni Novgorod State University, Nizhni Novgorod, Russia

Received September 21, 2011

Abstract—We consider multiindex transportation problems of linear and integer linear programming. As a method of solving them, we propose an approach based on reductions of multiindex transportation problems to min-flow problems. We show that under the reduction scheme we consider, the 2-embeddability condition for multiindex problems is a necessary and sufficient condition for the problem to be reducible to a min-cost flow problem.

DOI: 10.1134/S0005117913010086

1. INTRODUCTION

There exists a wide class of applied resource distribution problems formalized as multiindex (integer) linear programming problems of transportation type. Examples of such problems include (see [1–5]) problems of volume–calendar planning, data transmission channel power distribution, natural gas mining and transportation, refining condensed natural gas, etc. Multiindex assignment problems (a subclass of multiindex transportation integer linear programming problems) arise, for instance, in scheduling theory [6, 7] and in computer vision [8, 9].

To solve multiindex transportation problems of linear programming, one can use general methods like the simplex method or the Karmarkar algorithm [10, 11]. A number of works have been devoted specifically to solving multiindex linear programming problems of transportation type. Two-index problems have been studied the most [12]. Special subclasses of three- and four-index problems have been considered, e.g., in [13–15]. In the general setting, the class of multiindex problems has been studied in [13]. Conditions under which one can reduce dimension and/or reduce the number of indices in multiindex transportation problems have been discussed in [16]. The geometric properties of the set of admissible solutions for multiindex transportation systems of linear inequalities are discussed in [14, 17, 18].

It is especially interesting to solve integer multiindex linear programming problems of transportation type. It is known that the matrix of constraints for a two-index transportation problem is absolutely unimodular, and the class of two-index integer linear programming problems is thus solvable in polynomial time [12]. However, in the general setting the class of integer multiindex transportation problems is NP-hard already in the three-index case [19]. Moreover, for problems of this class there are no polynomial ε-approximating algorithms unless $P = NP$, and this result also holds already for the three-index case [20]. If there are no additional restrictions on the parameters, only general integer linear programming algorithms, exponential in their computational complexity, are applicable (e.g., the branch-and-bound method or the Gomory method [10, 21]). Among integer multiindex transportation problems, the best studied class is the class of multiindex assignment problems. A comprehensive survey of results related to complexity analysis and approximate algorithms in special subclasses of multiindex assignment problems is given in [22]; in addition to that, we note the works [23–25].

One promising direction in developing efficient algorithms for studying multiindex linear programming problems is finding subclasses of problems to which flow methods can be applied. This direction is influenced, importantly, by active studies in network optimization [26]. Existing efficient flow algorithms (see [27, 28]) allow, when linear programming problems reduce to flow problems, for algorithms that have lower computational complexity than general linear programming algorithms. In a number of cases, reduction to flow problems also allows for an algorithm for the original problem that is guaranteed to find an integer-valued solution and thus we are able to distinguish polynomially solvable subclasses of integer linear programming problems. The reducibility of linear programming problems to flow problems has been studied in [29–32]; these works differ, importantly, in the concepts of reduction used in each of them. The problem of reducing multiindex transportation linear programming problems has been less extensively studied. It is known that two-index problems reduce to flow problems [12]. The question of reducibility for multiindex problems with an arbitrary number of indices has been considered in [1, 33, 34].

In the study of multiindex systems of linear inequalities, the concept of reducing a system of linear inequalities to the problem of finding an admissible circulation was formulated in [34]. An important characteristic feature of this concept is the existence of a correspondence between variables in the original systems of inequalities and simple cycles in the auxiliary network. The proposed reduction scheme guarantees that an arbitrary admissible circulation in the auxiliary network will define such an admissible solution in the original system of inequalities that variables are assigned flow values along the corresponding simple cycles. The value of the flow along simple cycles is defined via cyclic decomposition of the admissible circulation.

It has been shown in [1, 33] that special 2-embeddedness conditions on the set of subsets of indices over which one sums in the problem's system of constraints are sufficient (in case of three-index problems, necessary and sufficient unless P = NP) for a reduction to the problem of finding a min-cost flow. One characteristic feature of the reductions used in [1, 33] is that there exists a correspondence between variables in the original problem and arcs in the auxiliary network. The proposed reduction scheme guarantees that any optimal flow in the auxiliary network will correspond to such an optimal solution for the original problem that the variables are assigned flow values along the corresponding arcs in the network. If the 2-embeddedness conditions hold, solving a multiindex linear programming problem of transportation type reduces to finding the minimal cost flow in a network with $O(n)$ vertices and $O(n)$ arcs, where n is the number of variables in the original problem.

In this work, we continue our study of reducing multiindex transportation problems to flow problems in the general reduction paradigm proposed in [1, 33]. In Section 2, we give a formalization of multiindex linear programming problems of transportation type and introduce the necessary notation. Section 3 introduces the concept of reduction we use in this work. In Section 4 we show that the 2-embedding condition is a necessary and sufficient condition for the reducibility (under this reduction scheme) of multiindex transportation problems to the class of min-cost flow problems. This result generalizes reducibility results shown in [1, 33] and gives (in the proposed reduction scheme) an exhausting answer to the question of reducibility of a class of multiindex problems to a class of min-cost flow problems. The resulting constructive reduction for multiindex problems with 2-embedded structure is optimal in the sense that a reduction with asymptotically lower computational costs is impossible, and an arbitrarily large increase in the computational cost of the reduction does not let us extend the class of reducible multiindex problems.

2. MULTIINDEX TRANSPORTATION PROBLEMS

To pose multiindex linear programming problems of transportation type, we use a formalization proposed in [13]. Let $s \in N$ and $N(s) = \{1, \ldots, s\}$. To each number l we assign a corresponding parameter j_l, called an *index*, that takes values from the set $J_l = \{1, \ldots, n_l\}$, where $n_l \geq 2$, $l \in N(s)$. Let $f = \{k_1, \ldots, k_t\} \subseteq N(s)$; $k_i < k_{i+1}$, $i = \overline{1, t-1}$. We call a set of values for indices $F_f = (j_{k_1}, \ldots, j_{k_t})_f$ a t-index. The set of all t-indices corresponding to f is defined as $E_f = \{F_f | F \in J_{k_1} \times \ldots \times J_{k_t}\}$. We will omit the subscript f in the t-indices if it does not lead to ambiguity. We denote the component j_i of the set F_f as $F_f(i) = j_i$, $i \in f$. Let $f' \subseteq f'' \subseteq N(s)$; then we denote $F_{f'} = (F_{f''})_{f'}$ if $F_{f'} \in E_{f'}$, $F_{f''} \in E_{f''}$ and $F_{f'}(i) = F_{f''}(i)$, $i \in f'$. If $F_{f'} \in E_{f'}$, $F_{f''} \in E_{f''}$, where $f', f'' \subseteq N(s)$ and $f' \cap f'' = \varnothing$, then we denote by $F_{f'}F_{f''}$ such a set that $F_{f'}F_{f''} \in E_{f' \cup f''}$ and $(F_{f'}F_{f''})_{f'} = F_{f'}$, $(F_{f'}F_{f''})_{f''} = F_{f''}$. We further define $\overline{f} = N(s) \backslash f$; then, according to our notation, $F_{N(s)} = F_f F_{\overline{f}}$ if $F_f = (F_{N(s)})_f$ and $F_{\overline{f}} = (F_{N(s)})_{\overline{f}}$.

To each set F_f we assign a real number z_{F_f}, $F_f \in E_f$. This mapping of the set of t-indices E_f to the set of real numbers we call (similar to [13]) the t-index matrix, denoted by $\{z_{F_f}\}$. For an s-index matrix $\{z_{N(s)}\}$, we introduce the following notation:

$$\sum_{F_f \in E_f} z_{F_f F_{\overline{f}}} = \sum_{j_{k_1} \in J_{k_1}} \sum_{j_{k_2} \in J_{k_2}} \cdots \sum_{j_{k_t} \in J_{k_t}} z_{F_f\, F_{\overline{f}}}, \quad F_{\overline{f}} \in E_{\overline{f}}.$$

This notation for the subsums of an s-index matrix will be used to formalize multiindex transportation problems.

Let M be a given set, $M \subseteq 2^{N(s)}$; $\{a_{F_{\overline{f}}}\}$, $\{b_{F_{\overline{f}}}\}$, given $|\overline{f}|$-index matrices of free coefficients, $0 \leq a_{F_{\overline{f}}} \leq b_{F_{\overline{f}}}$, $F_{\overline{f}} \in E_{\overline{f}}$, $f \in M$; $\{c_{F_{N(s)}}\}$, a given s-index matrix of objective function coefficients; $\{x_{F_{N(s)}}\}$, an s-index matrix of unknowns. Then a multiindex linear programming problems of transportation type is formalized as follows:

$$a_{F_{\overline{f}}} \leq \sum_{F_f \in E_f} x_{F_f F_{\overline{f}}} \leq b_{F_{\overline{f}}}, \quad F_{\overline{f}} \in E_{\overline{f}}, \quad f \in M; \tag{1}$$

$$x_{F_{N(s)}} \geq 0, \quad F_{N(s)} \in E_{N(s)}; \tag{2}$$

$$\sum_{F_{N(s)} \in E_{N(s)}} c_{F_{N(s)}} x_{F_{N(s)}} \to \min. \tag{3}$$

We will denote problem (1)–(3) by $w(s; M; n_1, n_2, \ldots, n_s; \{a_{F_{\overline{f}}}\}, \{b_{F_{\overline{j}}}\}, f \in M; \{c_{F_{N(s)}}\})$; the class of all multiindex problems of the form (1)–(3) for a given set M, by $W(M)$.

If $w \in W(M)$ then we denote a constraint of the form (1) for problem w corresponding to a fixed set $f \in M$ and a fixed tuple $F_{\overline{f}} \in E_{\overline{f}}$ by $d(w, f, F_{\overline{f}})$. We denote the matrix of the system of constraints for problem w by $Matr(w)$; the row of matrix $Matr(w)$ defined by the two-sided inequality $d(w, f, F_{\overline{f}})$, by $row(w, f, F_{\overline{f}})$, $F_{\overline{f}} \in E_{\overline{f}}$, $f \in M$; the column of matrix $Matr(w)$ corresponding to the variable $x_{F_{N(s)}}$, by $col(w, F_{N(s)})$, $F_{N(s)} \in E_{N(s)}$. Suppose that we are given sequences $f^{(1)}, \ldots, f^{(k_1)} \in M$, $F^{(1)}_{\overline{f^{(1)}}} \in E_{\overline{f^{(1)}}}, \ldots, F^{(k_1)}_{\overline{f^{(k_1)}}} \in E_{\overline{f^{(k_1)}}}$, $F^{(1)}_{N(s)}, \ldots, F^{(k_2)}_{N(s)} \in E_{N(s)}$. Then we denote the submatrix formed by elements of matrix $Matr(w)$ on the intersection of rows $row(w, f^{(i)}, F^{(i)}_{\overline{f^{(i)}}})$, $i = \overline{1, k_1}$ and columns $col(w, F^{(j)}_{N(s)})$, $j = \overline{1, k_2}$, by $Matr(w;\ (f^{(i)}, F^{(i)}_{\overline{f^{(i)}}}),\ i = \overline{1, k_1};\ F^{(j)}_{N(s)},\ j = \overline{1, k_2})$.

3. REDUCTIONS

Let us formalize the concept of reduction that we will further use to study reducibility between multiindex problems and flow algorithms.

Let $A \in R^{n \times m}$, $b, b^-, b^+ \in R^n$, $c \in R^m$ be fixed parameters; $x \in R^m$, the vector of unknowns. By $w(A, b, c)$ we denote the linear programming problem $\min\{(c, x) | Ax \leq b, x \geq 0\}$; by $w(A, b^-, b^+, c)$, the linear programming problem $\min\{(c, x) | b^- \leq Ax \leq b^+, x \geq 0\}$. For convenience, we denote by $nrow(A)$ and $ncol(A)$ the number of rows and columns in matrix A respectively. We note that problem $w(A, b^-, b^+, c)$ can be described with a notation of the form $w(A, b, c)$. Nevertheless, we will use the notation $w(A, b^-, b^+, c)$ in case when we want to emphasize that the problem's system of constraints is a system of two-sided inequalities. We will also consider integer linear programming problems. If $w = w(A, b, c)$ is a linear programming problem, by w_Z we denote the integer linear programming problem $w_Z = \min\{(c, x)|Ax \leq b, x \in Z_+^{ncol(A)}\}$. Let W be an arbitrary class of linear programming problems; we define the corresponding class of integer linear programming problems $W_Z = \{w_Z|w \in W\}$.

We further consider two classes of linear programming problems W' and W''. In essence, a class W' is reducible to class W'' if for any problem $w' \in W'$ it is possible to construct a corresponding problem $w'' \in W''$ in such a way that the solution of problem w'' determines the solution of problem w'. When formalizing a specific reduction scheme, we will determine computational costs and/or specific computational procedures related to:

—constructing the matrix of the system of constraints for problem w'' by the original parameters of problem w';

—constructing free coefficients and objective function coefficients for problem w'' by the original parameters of problem w';

—constructing a solution of problem w' by a solution of problem w''.

The notation for the reduction scheme proposed below is introduced similar to R. Graham's notation used to classify scheduling theory problems [35].

Definition 1. We say that a class W' is $t_1 - s_1|t_2 - s_2|t_3 - s_3$ reducible to class W'' if for every problem $w' = w(A', b', c') \in W'$ one can in time $O(t_1)$ construct the matrix A'', in time $O(t_2)$ construct the vectors b'', c'' such that $w'' = w(A'', b'', c'') \in W''$ and, moreover,

—problem w' is feasible (bounded) if and only if problem w'' is feasible (bounded);

—if an optimal (admissible) solution x'' of problem w'' is known then an optimal (admissible) solution x' of problem w' can be constructed in time $O(t_3)$.

Here $(-s_1), (-s_2), (-s_3)$ are optional string notations for computational procedures related to constructing the matrix of the system of constraints, free coefficients and objective function coefficients, and constructing the problem's solution in general.

We call problem w'' (see Definition 1) the problem corresponding to problem w'. Sometimes, for convenience, we will replace computational complexity estimates t_1, t_2, t_3 with L or P, meaning respectively linear or polynomial functions in the size of an individual problem w'.

In this work, we study the possibility to reduce the class of multiindex linear programming problems of transportation type to the class of problems of finding minimal cost flows defined as follows. Consider a directed graph $G = (V_G, A_G)$, $A_G \subseteq V_G \times V_G$, where V_G and A_G are the sets of vertices and arcs of the graph G respectively. Let l_{ij}, u_{ij} denote the throughputs of the arc (i, j); e_{ij}, the cost of an arc (i, j); x_{ij}, an unknown value of the flow along the arc (i, j), $(i, j) \in A_G$. Then by $v(G; l_{ij}, u_{ij}, e_{ij}, (i, j) \in A_G)$ we denote the following min-cost flow problem:

$$\sum_{j:(i,j)\in A_G} x_{ij} - \sum_{j:(j,i)\in A_G} x_{ji} = 0, \quad i \in V_G,$$

$$l_{ij} \leq x_{ij} \leq u_{ij}, \quad (i, j) \in A_G,$$

$$x_{ij} \geq 0, \quad (i, j) \in A_G,$$

$$\sum_{(i,j)\in A_G} e_{ij} x_{ij} \to \min.$$

We denote by $Graph$ the set of all directed graphs. We define the class of problems of finding min-cost flows as $W_{Graph} = \{v(G, l_{ij}, u_{ij}, e_{ij}, (i,j) \in A_G) | l_{ij}, u_{ij} \in Z_+, e_{ij} \in Z, (i,j) \in A_G, G \in Graph\}$.

Definition 2. Let W be a class of linear programming problems with a two-sided system of linear inequalities. We say that a class W is $t_1|t_2 - equal|t_3 - edge$ reducible to a class W_{Graph} if class W is $t_1|t_2|t_3$ reducible to class W_{Graph} and an arbitrary problem $w = w(A, b^-, b^+, c) \in W$ and its corresponding problem $v = v(G; l_{ij}, u_{ij}, e_{ij}, (i,j) \in A_G) \in W_{Graph}$ satisfy the following conditions: there exist such injective functions $\alpha : \{1, \ldots, nrow(A)\} \to A_G, \beta : \{1, \ldots, ncol(A)\} \to A_G$, that

—$l_{\alpha(i)} = b_i^-$, $u_{\alpha(i)} = b_i^+$, $i \in \{1, \ldots, nrow(A)\}$; $l_{(u,v)} = 0$, $u_{uv} = b^*$, $(u,v) \in A_G \backslash \{\alpha(i) | i \in \{1, \ldots, nrow(A)\}\}$, where $b^* = \sum_{k=1}^{nrow(A)} b_k^+$;

—$e_{\beta(i)} = c_i$, $i \in \{1, \ldots, ncol(A)\}$; $e_{uv} = 0$, $(u,v) \in A_G \backslash \{\beta(i) | i \in \{1, \ldots, ncol(A)\}\}$;

—if x_{ij}, $(i,j) \in A_G$, is an optimal (admissible) solution of problem v then $(x_{\beta(1)}, x_{\beta(2)}, \ldots, x_{\beta(ncol(A))})$ will be an optimal (admissible) solution of problem w.

Remark. For b^* one can use any sufficiently large value that would be equivalent to no upper bound on the throughput in an arc.

Thus, according to Definition 2, in case class W is $t_1|t_2 - equal|t_3 - edge$ reducible to class W_{Graph} we guarantee that if $w \in W$, $v = v(G; l_{ij}, u_{ij}, c_{ij}, (i,j) \in A_G) \in W_{Graph}$ and v is the problem corresponding to problem w, then in constructing the min-cost problem v the throughputs and costs of arcs in the problem are defined via the coefficients of problem w, and the solution of problem w is found via a subset of components of a solution of problem v. Then we can propose an algorithm for solving problem w which is based on the solution of the corresponding problem v with computational complexity $O(t_1 + t_2 + t_3 + \mu(|V_G|, |A_G|))$, where $\mu(n, m)$ is the computational complexity of the algorithm for solving a min-cost flow problem in a network with n vertices and m arcs. A survey of computational complexity estimates for known flow algorithms can be found, e.g., in [27, 28]. Further, in this work we consider the conditions under which the class $W(M)$ is $t_1|t_2 - equal|t_3 - edge$ reducible to class W_{Graph}.

4. REDUCIBILITY CONDITIONS FOR MULTIINDEX PROBLEMS

The form of linear programming problems in class $W(M)$ is defined by the given set M. Therefore, the problem is to find the conditions that a set M has to satisfy in order for a solution of a problem from the class $W(M)$ to allow finding with flow algorithms.

Theorem 1. *Let $M \subseteq 2^{N(s)}$. If a class $W(M)$ is $t_1|t_2 - equal|t_3 - edge$ reducible to class W_{Graph} then for an arbitrary problem $w \in W(M)$ the matrix $Matr(w)$ is absolutely unimodular.*

Proof by contradiction. Suppose that class $W(M)$ is $t_1|t_2 - equal|t_3 - edge$ reducible to class W_{Graph}, and there exists a problem $w \in W(M)$ such that the matrix $Matr(w)$ is not absolutely unimodular. According to [36], the absolute unimodularity condition of the matrix is necessary and sufficient for all vertices in the polyhedron of the corresponding feasible system of linear inequalities to be integer-valued. Therefore, there exists a problem $w' \in W(M)$ satisfying the following conditions:

—$Matr(w') = Matr(w)$;

—free coefficients in problem w' are integers;

—the system of constraints in problem w' is feasible;

—problem w' has a single optimal solution, and it is not integer.

The class $W(M)$ is $t_1|t_2 - equal|t_3 - edge$ reducible to class W_{Graph}. Then consider the problem $v \in W_{Graph}$ corresponding to problem w'. By Definition 2, network throughputs in the min-cost flow problem v are integer-valued. The matrix of system of constraints in problem v is absolutely unimodular, so problem v has an integer optimal solution. Then, by Definition 2, problem w' also has an integer optimal solution. We have arrived at a contradiction, which completes the proof of the theorem.

Definition 3. A set M, $M \subseteq 2^{N(s)}$, is called k-embedded if there exists a partition of the set M into k subsets $M_i = \{f_1^{(i)}, \dots, f_{m_i}^{(i)}\}$, $i = \overline{1,k}$, such that $f_j^{(i)} \subseteq f_{j+1}^{(i)}$, $j = \overline{1, m_i - 1}$, $i = \overline{1,k}$.

Previously we have found the following sufficient reducibility conditions.

Theorem 2 [1]. *Let $M \subseteq 2^{N(s)}$. In order for the class $W(M)$ to be $L|L-equal|L-edge$ reducible to class W_{Graph} it is sufficient for the set M to be 2-embedded.*

A constructive way of proving Theorem 1, proposed in [1], for the case of 2-embeddedness of the set M lets on, for each problem $w \in W(M)$, to construct the corresponding problem $v = v(G; l_{ij}, u_{ij}, e_{ij}, (i,j) \in A_G) \in W_{Graph}$. Here $|V_G| = O(|E_{N(s)}|)$, $|A_G| = O(|E_{N(s)}|)$, where the value $|E_{N(s)}|$ coincides with the number of variables in problem w. By Definition 2, in case of $L|L-equal|L-edge$ reducibility the construction of the corresponding flow problem and finding a solution of the original problem by the solution of the corresponding problem both take linear time. Let $\mu(n,m)$ be the computational complexity of solving the min-cost flow problem in a network with n vertices and m arcs. Then we can formulate the following corollary.

Corollary 1. *Let $M \subseteq 2^{N(s)}$. If the set M is 2-embedded then the class of problems $W(M)$ is solvable in time $O(\mu(|E_{N(s)}|, |E_{N(s)}|))$.*

It is known [26] that the min-cost flow problem (admissible flow problem) in a network with two-sided throughputs with n vertices and m arcs reduces to finding a min-cost flow with a given value (max-flow problem) in a network with single-sided throughputs with $O(n)$ vertices and $O(n+m)$ arcs. We use the Orlin algorithm proposed in [27] to find the min-cost flow of a given value. To find the maximal flow, we use the Goldberd–Rao algorithm [28]. Then, by Corollary 2, if the set M is 2-embedded then there exist algorithms for finding an optimal and admissible solution of problem $w \in W(M)$ that require $O(|E_{N(s)}|^2 \log^2 |E_{N(s)}|)$ and $O(|E_{N(s)}|^2 \log |E_{N(s)}|)$ computational operations respectively. This approach is also suitable for solving a class of integer-valued multiindex problems $W_Z(M)$.

We further show that the 2-embedding condition is necessary and sufficient for a multiindex transportation problem to be reducible to a min-cost flow problem.

Definition 4. Let $M \subseteq 2^{N(s)}$ and $g \subseteq N(s)$; then we denote $M(g) = \{f \cap g | f \in M\}$.

Definition 5. Let $s_1 \le s_2$ and $M_1 \subseteq 2^{N(s_1)}$, $M_2 \subseteq 2^{N(s_2)}$. Then we denote $M_1 \prec M_2$ if there exists a subset $g \subseteq N(s_2)$, $|g| = s_1$ and a bijection π: $g \to N(s_1)$ such that $M_1 \subseteq \cup_{f \in M_2(g)} \{\{\pi(i) | i \in f\}\}$.

In essence, Definition 5 means that if we "ignore" the indices from the set $N(s_2) \backslash g$, for every problem from class $W(M_2)$ there exists an equivalent (up to renumerating the indices π) problem from class $W(M_1)$.

Lemma 1. *Let $s_1 \le s_2$, $M_1 \subseteq 2^{N(s_1)}$, $M_2 \subseteq 2^{N(s_2)}$, and $M_1 \prec M_2$. Then, if there exists a problem $w_1 \in W(M_1)$ such that the matrix $Matr(w_1)$ is not absolutely unimodular then there exists a problem $w_2 \in W(M_2)$ such that matrix $Matr(w_2)$ also is not absolutely unimodular.*

Proof. Suppose that the lemma's conditions hold. Then, by Definition 5, there exists a subset $g \subseteq N(s_2)$, $|g| = s_1$ and a bijection $\pi : g \to N(s_1)$ such that $M_1 \subseteq \cup_{f \in M_2(g)} \{\{\pi(i) | i \in f\}\}$.

Let us show that for every problem $w_1 \in W(M_1)$ there exists a problem $w_2 \in W(M_2)$ such that $Matr(w_1)$ is a submatrix of $Matr(w_2)$. Consider an arbitrary problem $w_1 \in W(M_1)$, $w_1 = w(s_1; n'_1, \dots, n'_{s_1}; \{a'_{F_{\overline{f}}}\}, \{b'_{F_{\overline{f}}}\}, f \in M_1; \{c'_{F_{N(s_1)}}\})$ and choose $w_2 \in W(M_2)$, $w_2 = w(s_2; n''_1, \dots, n''_{s_2}; \{a''_{F_{\overline{f}}}\}, \{b''_{F_{\overline{f}}}\}, f \in M_2; \{c''_{F_{N(s_2)}}\})$ such that $n'_{\pi(i)} = n''_i$, $i \in g$. Then consider an arbitrary row $row(w_1, f_1, F_{\overline{f_1}})$ of matrix $Matr(w_1)$. The row $row(w_1, f_1, F_{\overline{f_1}})$ can also be specified as a submatrix $Matr(w_1; f_1, F_{\overline{f_1}}; F_{N(s_1)} \in E_{N(s_1)})$. Choose $f_2 \in M_2$ in such a way that $f_1 = \{\pi(i) | i \in f_2 \cap g\}$

and choose $F_{\overline{f_2}} \in E_{\overline{f_2}}$ such that $F_{\overline{f_2}}(i) = F_{\overline{f_1}}(\pi(i))$, $i \in \overline{f_2} \cap g$ and $F_{\overline{f_2}}(i) = 1$, $i \in \overline{f_2} \backslash g$. In the matrix row $row(w_2, f_2, F_{\overline{f_2}})$, consider elements located on the intersection with the columns $\{col(w_2, F_{N(s_2)}) | F_{N(s_2)} = F_g(1, \ldots, 1)_{\bar{g}}$, where $F_g \in E_g\}$. We see that the matrices $Matr(w_1; (f_1, F_{\overline{f_1}}); F_{N(s_1)} \in E_{N(s_1)})$ and $Matr(w_2; (f_2, F_{\overline{f_2}}); F_g(1, \ldots, 1)_{\overline{g}}, F_g \in E_g)$ of size $1 \times |E_{N(s_1)}|$ coincide up to a permutation of columns given by the function π. Therefore, matrix $Matr(w_1)$ is a submatrix of $Matr(w_2)$.

Consequently, if there exists a problem $w_1 \in W(M_1)$ such that the matrix $Matr(w_1)$ is not absolutely unimodular (i.e., contains a minor other than $0, 1, -1$), then there exists a problem $w_2 \in W(M_2)$ such that $Matr(w_2)$ also contains a minor other than $0, 1, -1$. This completes the proof of the lemma.

Lemma 2. *Let $M \subseteq 2^{N(s)}$. If one of the following three conditions holds*:

(1) $s = 3$, $M = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$,

(2) $s = 3$, $M = \{\{1\}, \{2\}, \{3\}\}$,

(3) $s = 4$, $M = \{\{1, 2\}, \{2, 3\}, \{1, 4\}\}$,

then there exists $w \in W(M)$ such that matrix $Matr(w)$ is not absolutely unimodular:

Proof. (1) Let $s = 3$, $M = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$. By definition, $n_1, n_2, n_3 \geq 2$, so we choose an arbitrary problem $w \in W(M)$ and consider its submatrix

$$H = Matr(w;$$
$$(\{1,2\},(1)_{\{3\}}),(\{1,3\},(1)_{\{2\}}),(\{2,3\},(1)_{\{1\}});$$
$$(1,1,2)_{N(3)}, (1,2,1)_{N(3)}, (2,1,1)_{N(3)}).$$

We see that $H = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$ and $\det H = 2$.

(2) Let $s = 3$, $M = \{\{1\}, \{2\}, \{3\}\}$. Consider an arbitrary problem $w \in W(M)$ such that $n_1, n_2, n_3 \geq 3$ and consider its submatrix

$$H = Matr(w;$$
$$(\{1\},(1,3)_{\{2,3\}}),(\{1\},(2,2)_{\{2,3\}}),(\{1\},(2,3)_{\{2,3\}}),$$
$$(\{3\},(1,1)_{\{1,2\}}),(\{3\},(2,2)_{\{1,2\}})),$$
$$(\{2\},(1,2)_{\{1,3\}}),(\{2\},(3,3)_{\{1,3\}});$$
$$(1,1,2)_{N(3)}, (1,1,3)_{N(3)}, (1,2,2)_{N(3)}, (2,2,2)_{N(3)},$$
$$(2,2,3)_{N(3)}, (3,1,3)_{N(3)}, (3,2,3)_{N(3)}).$$

We see that $H = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$ and $\det H = 2$.

(3) Let $s = 4$, $M = \{\{1,2\},\{2,3\},\{1,4\}\}$. By definition, $n_1, n_2, n_3 \geq 2$, so choose an arbitrary problem $w \in W(M)$ and consider its submatrix

$$
\begin{aligned}
H = Matr(w;&\\
&(\{1,2\},(1,1)_{\{3,4\}}),(\{2,3\},(1,1)_{\{1,4\}}),(\{2,3\},(1,2)_{\{1,4\}}),\\
&(\{1,4\},(1,1)_{\{2,3\}}),(\{1,4\},(1,2)_{\{2,3\}});\\
&(1,1,1,2)_{N(4)},(1,1,2,1)_{N(4)},(1,1,2,2)_{N(4)},(1,2,1,1)_{N(4)},(2,1,1,1)_{N(4)}).
\end{aligned}
$$

It is easy to see that $H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}$ and $\det H = -2$. This completes the proof of the lemma.

Remark. Matrices used to prove Lemma 2 have been obtained with a parallelized program developed by a student A.S. Katerov [37]. The software was run on the supercomputer of the collective use computational center at FGUP "RFYATS–VNIIEF."

Lemma 3. *Let $M \subseteq 2^{N(s)}$. In order for the set M to be 1-embedded it is necessary and sufficient that for every $f_1, f_2 \in M$ there exist $k, l \in \{1,2\}, k \neq l$, such that $f_k \subseteq f_l$.*

Proof. Necessity proof. Let the set M be 1-embedded. Then, according to Definition 3, the set M can be represented as follows: $M = \{f_1^{(1)}, \ldots, f_{m_1}^{(1)}\}$, where $f_j^{(1)} \subseteq f_{j+1}^{(1)}$, $j = \overline{1, m_1 - 1}$. Consider arbitrary $j_1, j_2 \in \{1, \ldots, m_1\}$. If $j_1 \leq j_2$ then $f_{j_1}^{(1)} \subseteq f_{j_2}^{(1)}$, otherwise $f_{j_2}^{(1)} \subseteq f_{j_1}^{(1)}$.

Sufficiency proof. Suppose that for every $f_1, f_2 \in M$ there exist $k, l \in \{1,2\}$, $k \neq l$, such that $f_k \subseteq f_l$. We reorder the elements of M in the nondescending order of their sizes, $M = \{f_{t_1}, \ldots, f_{t_{|M|}}\}$, $|f_{t_j}| \leq |f_{t_{j+1}}|$, $j = \overline{1, |M| - 1}$. Since $f_{t_j} \neq f_{t_{j+1}}$ and $|f_{t_j}| \leq |f_{t_{j+1}}|$, we get $f_{t_{j+1}} \not\subseteq f_{t_j}$, and then, by assumption, $f_{t_j} \subseteq f_{t_{j+1}}$, $j = \overline{1, |M| - 1}$. By Definition 3, the set M is 1-embedded. This completes the proof of the lemma.

Theorem 3. *Let $M \subseteq 2^{N(s)}$. In order for the set M to be 2-embedded it is necessary and sufficient that for every $f_1, f_2, f_3 \in M$ there exist $k, l \in \{1,2,3\}, k \neq l$, such that $f_k \subseteq f_l$.*

Proof. Necessity proof. Suppose that the set M is 2-embedded; then, by Definition 3, there exists a partition of the set M into 2 subsets $M_1 = \{f_1^{(1)}, \ldots, f_{m_1}^{(1)}\}$, $M_2 = \{f_1^{(2)}, \ldots, f_{m_2}^{(2)}\}$ such that $f_j^{(i)} \subseteq f_{j+1}^{(i)}$, $j = \overline{1, m_i - 1}$, $i = \overline{1,2}$. Consider arbitrary $f_1, f_2, f_3 \in M$. There exist $k, l \in \{1,2,3\}, k \neq l$ and $t \in \{1,2\}$ such that $f_k, f_l \in M_t$. If $|f_k| \leq |f_l|$ then $f_k \subseteq f_l$, otherwise $f_l \subseteq f_k$.

Sufficiency proof. Suppose that for every $f_1, f_2, f_3 \in M$ there exist $k, l \in \{1,2,3\}$, $k \neq l$, such that $f_k \subseteq f_l$. We give a constructive proof, showing an algorithm for finding the partition $M_1 = \{f_1^{(1)}, \ldots, f_{m_1}^{(1)}\}$, $M_2 = \{f_1^{(2)}, \ldots, f_{m_2}^{(2)}\}$ of the set M such that $f_j^{(i)} \subseteq f_{j+1}^{(i)}$, $j = \overline{1, m_i - 1}$, $i = \overline{1,2}$.

We reorder elements of the set M in nondescending order of their sizes, $M = \{g_1, \ldots, g_{|M|}\}$, $|f_{t_j}| \leq |f_{t_{j+1}}|$, $j = \overline{1, |M| - 1}$. The algorithm is as follows.

Step 1. Let $M_1, M_2 := \varnothing$, $j := 1$. Goto 2.

Step 2. Reorder elements of the sets M_1, M_2 in nondescending order of their sizes $M_1 = \{f_1^{(1)}, \ldots, f_{m_1}^{(1)}\}$, $M_2 = \{f_1^{(2)}, \ldots, f_{m_2}^{(2)}\}$, where $f_k^{(i)} \leq f_{k+1}^{(i)}$, $k = \overline{1, m_i - 1}$, $i = \overline{1,2}$. If $j > |M|$ the algorithm is over; otherwise goto 3.

Step 3. If there exists $l \in \{1,2\}$ such that $f_{m_l}^{(l)} \subseteq g_j$ then $M_l := M_l \cup \{g_j\}$, $j := j + 1$, goto 2; otherwise, goto 4.

Step 4. Denote $I_l = \{f_i^{(l)} | f_s^l \not\subset g_j, s = \overline{i, m_l}, i \in \{1, \ldots, m_l\}\}$, $i_l^* = \min_{i | f_i^{(l)} \in I_l} i$, $l = \overline{1, 2}$. Consider the set $I = I_1 \cup I_2$ and reorder its elements in nondescending order of their sizes $I = \{q_1, \ldots, q_{|I|}\}$, $|q_s| \leq |q_{s+1}|$, $s = \overline{1, |I| - 1}$. Let

$$t = \begin{cases} 1, & \text{if } \ q_1 \in I_1 \\ 2, & \text{if } \ q_1 \in I_2, \end{cases} \qquad r = \begin{cases} 1, & \text{if } \ q_1 \notin I_1 \\ 2, & \text{if } \ q_1 \notin I_2. \end{cases}$$

Then $M_t := I_r \cup M_t$, $M_r := \{g_j\} \cup M_r \backslash I_r$, $j := j + 1$, goto 2.

Let us show that the resulting algorithm is correct. By construction, by the beginning of each of Steps 2 it holds that $M_1 \cap M_2 = \varnothing$, and the next element g_j is added either to M_1 or to M_2. Thus, after the algorithm is over M_1, M_2 represent a partition of the set M. Let us further show that before each of Steps 2 the sets M_1, M_2 are 1-embedded.

Initially $M_1, M_2 = \varnothing$, and by Definition 3 they are 1-embedded. Let $M_i = \{f_1^{(i)}, \ldots, f_{m_i}^{(i)}\}$, $f_j^{(i)} \subseteq f_{j+1}^{(i)}$, $j = \overline{1, m_i - 1}$, $i = \overline{1, 2}$. If on Step 3 there exists $l \in \{1, 2\}$ such that $f_{m_l}^{(l)} \subseteq g_j$ then $M_l := M_l \cup \{g_j\}$, and since $f_j^{(l)} \subseteq f_{j+1}^{(l)}$, $j = \overline{1, m_l - 1}$, the set M_l is 1-embedded. In case of going on to Step 4 it holds that $f_{m_1}^{(1)}, f_{m_2}^{(2)} \not\subset g_j$, so $f_{m_1}^{(1)} \in I_1$, $f_{m_2}^{(2)} \in I_2$, and, therefore, $I_1, I_2 \neq \varnothing$ and there exist values i_1^*, i_2^*. Schematically, the sets M_1 and M_2 can be represented as shown on Fig. 1 (here elements of the 1-embedded set $M_i \backslash I_i$ are subsets of elements of the 1-embedded set I_i, $i = \overline{1, 2}$):

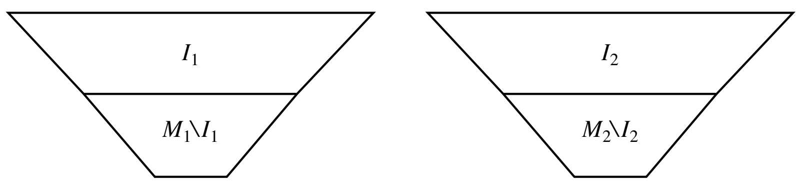

Fig. 1.

By construction, $q \not\subset g_j$, $q \in I$. On the other hand, according to the algorithm above $|g_j| \geq q$, $q \in I$, so $g_j \not\subset q$, $q \in I$. Consider arbitrary $q' \in I_1$, $q'' \in I_2$. As we have already shown, $g_j \not\subset q', q''$ and $q', q'' \not\subset g_j$, then by the assumptions of the lemma one of the following relations hold: $q' \subseteq q''$ or $q'' \subseteq q'$. Therefore, by Lemma 3 the set I is 1-embedded, and $q_s \subseteq q_{s+1}$, $s = \overline{1, |I| - 1}$. By construction, $q_1 \in I_t$. Further, either $M_t \backslash I_t = \varnothing$ or $M_t \backslash I_t = \{f_1^{(t)}, \ldots, f_{i_t^* - 1}^{(t)}\}$, and $f_{i_t^* - 1}^{(t)} \subseteq f_{i_t^*}^{(t)} = q_1$. Therefore, the constructed set $M_t := I \cup M_t \backslash I_t$ is 1-embedded. Further, either $M_r \backslash I_r = \varnothing$ or $M_r \backslash I_r = \{f_1^{(r)}, \ldots, f_{i_r^* - 1}^{(r)}\}$, and by construction $f_{i_r^* - 1}^{(r)} \subseteq g_j$. Therefore, the constructed set $M_r := \{g_j\} \cup M_r \backslash I_r$ is 1-embedded. Sets M_1 and M_2 constructed on Step 4 are schematically shown on Fig. 2 (here elements of the 1-embedded set $M_t \backslash I_t$ are subsets of elements of the 1-embedded set $I_r \cup I_t$, and elements of the 1-embedded set $M_r \backslash I_r$ are subsets of the set g_j):

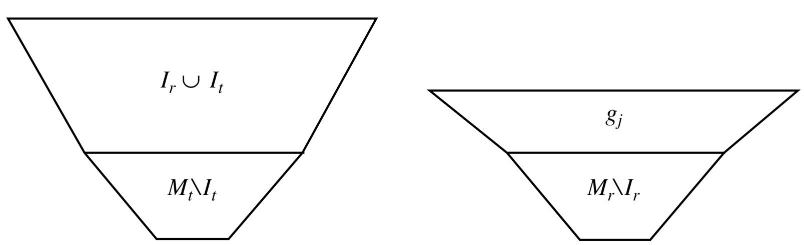

Fig. 2.

Consequently, after the algorithm has finished the constructed sets M_1, M_2 represent a partition of the set M and are 1-embedded. This completes the proof of the theorem.

Consider arbitrary sets $f_1, f_2, f_3 \subseteq N(s)$ for which the following condition holds:

$$f_1 \not\subset f_2, f_3, \quad f_2 \not\subset f_1, f_3, \quad f_3 \not\subset f_1, f_2. \tag{4}$$

Condition (4) holds if and only if there exist elements $a_{ij} \in N(s)$, $j \in \{1,2,3\}\backslash\{i\}$, $i \in \{1,2,3\}$, such that

$$\begin{aligned} &a_{12} \in f_1,\ a_{12} \notin f_2, \quad a_{21} \in f_2,\ a_{21} \notin f_1, \quad a_{31} \in f_3,\ a_{31} \notin f_1,\\ &a_{13} \in f_1,\ a_{13} \notin f_3, \quad a_{23} \in f_2,\ a_{23} \notin f_3, \quad a_{32} \in f_3,\ a_{32} \notin f_2. \end{aligned}$$

We will call the set $A = \{a_{ij} | a_{ij} \in f_i, a_{ij} \notin f_j, j \in \{1,2,3\}\backslash\{i\}, i \in \{1,2,3\}\}$ a dividing set for f_1,f_2,f_3. We denote by $A(f_1, f_2, f_3)$ the set of all sets dividing f_1, f_2, f_3. Theorem 3 together with the notion of a dividing set imply the following corollary.

Corollary 2. *Let $M \subseteq 2^{N(s)}$. In order for the set M to be 2-embedded it is necessary and sufficient that $A(f_1, f_2, f_3) = \varnothing$ for any $f_1, f_2, f_3 \in M$.*

If $A \in A(f_1, f_2, f_3)$, $A = \{a_{ij} | a_{ij} \in f_i, a_{ij} \notin f_j, j \in \{1,2,3\}\backslash\{i\}, i \in \{1,2,3\}\}$, then by $p(A)$ we will denote the value

$$p(A) = |\{a_{12}\} \cup \{a_{13}\}| + |\{a_{21}\} \cup \{a_{23}\}| + |\{a_{31}\} \cup \{a_{32}\}|.$$

Let $d^*(f_1, f_2, f_3) = \min_{A \in A(f_1,f_2,f_3)} |A|$. Then consider the problem of choosing, among the sets dividing f_1, f_2, f_3, a set of size $d^*(f_1, f_2, f_3)$ with maximal value of $p(A)$:

$$A^*(f_1, f_2, f_3) = \underset{A \in A(f_1,f_2,f_3) | d^*(f_1,f_2,f_3) = |A|}{\arg\max} p(A). \tag{5}$$

A solution $A^*(f_1, f_2, f_3)$ of problem (5) possesses the following important property.

Lemma 4. *Let $f_1, f_2, f_3 \subseteq N(s)$, $A(f_1, f_2, f_3) \neq \varnothing$ and $A^*(f_1, f_2, f_3) = \{a^*_{ij} | a^*_{ij} \in f_i, a^*_{ij} \notin f_j$, $j \in \{1,2,3\}\backslash\{i\}, i \in \{1,2,3\}\}$. If $a^*_{ij} \in f_k$ then $a^*_{ij} = a^*_{kj}$, where $i,j,k \in \{1,2,3\}$, $i \neq j$, $i \neq k$, $j \neq k$.*

Proof by contradiction. Suppose that conditions of the lemma hold, and there exist $i,j,k \in \{1,2,3\}$, $i \neq j$, $i \neq k$, $j \neq k$, such that $a^*_{ij} \in f_k$, but $a^*_{ij} \neq a^*_{kj}$. Consider the set $A' = \{a'_{st} |\ s \in \{1,2,3\}\backslash\{t\},\ t \in \{1,2,3\}\}$, constructed as follows:

$$\begin{aligned} &a'_{ij} = a^*_{ij}, \quad a'_{ji} = a^*_{ji}, \quad a'_{ki} = a^*_{ki},\\ &a'_{ik} = a^*_{ik}, \quad a'_{jk} = a^*_{jk}, \quad a'_{kj} = a^*_{ij}. \end{aligned}$$

By assumption $a'_{kj} = a^*_{ij} \in f_k$, here by construction we have $a'_{kj} = a^*_{ij} \notin f_j$. Therefore, $a'_{st} \in f_s$, $a'_{st} \notin f_t$, $t \in \{1,2,3\}\backslash\{s\}$, $s \in \{1,2,3\}$ and $A' \in A(f_1, f_2, f_3)$. Let us show the proof by considering the following two possible cases.

(1) Let $a^*_{kj} = a^*_{ki}$. Since $a'_{ki} = a^*_{ki} = a^*_{kj}$, $a'_{kj} = a'_{ij} = a^*_{ij}$, we get that $|A'| =$

$$\begin{aligned} &|\{a'_{ij}\} \cup \{a'_{ik}\} \cup \{a'_{ji}\} \cup \{a'_{jk}\} \cup \{a'_{ki}\} \cup \{a'_{kj}\}|\\ =&|\{a'_{ij}\} \cup \{a'_{ik}\} \cup \{a'_{ji}\} \cup \{a'_{jk}\} \cup \{a'_{ki}\}|\\ =&|\{a^*_{ij}\} \cup \{a^*_{ik}\} \cup \{a^*_{ji}\} \cup \{a^*_{jk}\} \cup \{a^*_{ki}\}|\\ =&|\{a^*_{ij}\} \cup \{a^*_{ik}\} \cup \{a^*_{ji}\} \cup \{a^*_{jk}\} \cup \{a^*_{ki}\} \cup \{a^*_{kj}\}| \ = |A^*(f_1, f_2, f_3)|. \end{aligned}$$

By construction, $a'_{kj} = a^*_{ij} \in f_i$ and $a'_{ki} = a^*_{ki} \notin f_i$, and thus, $a'_{kj} \neq a'_{ki}$. By assumption, $a^*_{kj} = a^*_{ki}$. Therefore, $p(A') =$

$$\begin{aligned}&|\{a'_{ij}\} \cup \{a'_{ik}\}| + |\{a'_{ji}\} \cup \{a'_{jk}\}| + |\{a'_{ki}\} \cup \{a'_{kj}\}|\\ =&|\{a^*_{ij}\} \cup \{a^*_{ik}\}| + |\{a^*_{ji}\} \cup \{a^*_{jk}\}| + 2\end{aligned}$$

and $p(A^*(f_1, f_2, f_3)) =$

$$\begin{aligned}&|\{a^*_{ij}\} \cup \{a^*_{ik}\}| + |\{a^*_{ji}\} \cup \{a^*_{jk}\}| + |\{a^*_{ki}\} \cup \{a^*_{kj}\}|\\ =&|\{a^*_{ij}\} \cup \{a^*_{ik}\}| + |\{a^*_{ji}\} \cup \{a^*_{jk}\}| + 1.\end{aligned}$$

Consequently, $p(A') = p(A^*(f_1, f_2, f_3)) + 1$, and we have arrived at a contradiction.

(2) Let $a^*_{kj} \neq a^*_{ki}$. By assumption, $a^*_{kj} \neq a^*_{ij}$. By construction, $a^*_{kj} \in f_k$ and $a^*_{ik} \notin f_k$, then $a^*_{kj} \neq a^*_{ik}$. Further, by construction $a^*_{kj} \notin f_j$, $a^*_{ji}, a^*_{jk} \in f_j$, so $a^*_{kj} \neq a^*_{ji}, a^*_{jk}$. Therefore, $|A'| =$

$$\begin{aligned}&|\{a'_{ij}\} \cup \{a'_{ik}\} \cup \{a'_{ji}\} \cup \{a'_{jk}\} \cup \{a'_{ki}\} \cup \{a'_{kj}\}|\\ =&|\{a'_{ij}\} \cup \{a'_{ik}\} \cup \{a'_{ji}\} \cup \{a'_{jk}\} \cup \{a'_{ki}\}|\\ =&|\{a^*_{ij}\} \cup \{a^*_{ik}\} \cup \{a^*_{ji}\} \cup \{a^*_{jk}\} \cup \{a^*_{ki}\}|\\ =&|A^*(f_1, f_2, f_3) \backslash \{a^*_{kj}\}| = |A^*(f_1, f_2, f_3)| - 1.\end{aligned}$$

We have arrived at a contradiction, so our assumption was false. This completes the proof of the lemma.

Lemma 5. *Let $M \subseteq 2^{N(s)}$. If the set M is not 2-embedded then there exists a problem $w \in W(M)$ such that matrix $Matr(w)$ is not absolutely unimodular.*

Proof. Suppose that the set $M \subseteq 2^{N(s)}$ is not 2-embedded. Then, by Corollary 2, there exist $f_1, f_2, f_3 \in M$ such that $A(f_1, f_2, f_3) \neq \varnothing$. Consider the set $A^*(f_1, f_2, f_3) = \{a^*_{ij} | a^*_{ij} \in f_i, a^*_{ij} \notin f_j,$ $j \in \{1, 2, 3\} \backslash \{i\},\ i \in \{1, 2, 3\}\}$, that represents a solution of problem (5). We will schematically represent possible combinations of the set $A^*(f_1, f_2, f_3)$ as a graph G_f with the set of vertices $V = \{ij|\ j \in \{1, 2, 3\} \backslash \{i\},\ i \in \{1, 2, 3\}\}$ and the set of arcs $\{(ij, kl) | a^*_{ij} = a^*_{kl},\ ij \neq kl,\ ij, kl \in V\}$. We show the proof by considering the following two possible cases.

(1) Suppose that for every $i \in \{1, 2, 3\}$ there exists such $t_i \in \{1, 2, 3\} \backslash \{i\}$ that $a^*_{it_i} \notin f_{k_i}$, where $k_i \in \{1, 2, 3\} \backslash \{i, t_i\}$. Consider the elements $a^*_{1t_1}, a^*_{2t_2}, a^*_{3t_3}$. By construction,

$$a^*_{1t_1} \notin f_2, f_3 \quad a^*_{2t_2} \notin f_1, f_3 \quad a^*_{3t_3} \notin f_1, f_2.$$

Then the subgraph of graph G_f induced by the subset of vertices $\{1t_1, 2t_2, 3t_3\}$ will look like the following (Fig. 3):

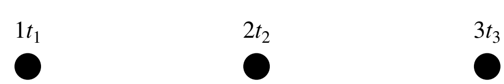

Fig. 3.

Therefore, $\{\{a^*_{1t_1}\}, \{a^*_{2t_2}\}, \{a^*_{3t_3}\}\} \subseteq M(\{a^*_{1t_1}, a^*_{2t_2}, a^*_{3t_3}\})$, and by Definition 5 $\{\{1\}, \{2\}, \{3\}\} \prec M$. Then according to Lemmas 1 and 2 there exists $w \in W(M)$ such that $Matr(w)$ is not absolutely unimodular.

(2) Suppose that there exists $i \in \{1, 2, 3\}$ such that for every $j \in \{1, 2, 3\} \backslash \{i\}$ it holds that $a^*_{ij} \in f_k$, where $k \in \{1, 2, 3\} \backslash \{i, j\}$. Without loss of generality we assume that $i = 1$, or else simply

renumber the elements. Then $a^*_{12} \in f_3$, $a^*_{13} \in f_2$. By Lemma 4, $a^*_{12} = a^*_{32}$, $a^*_{13} = a^*_{23}$. By construction, $a^*_{kl} \in f_k$, $a^*_{kl} \notin f_l$, $l \in \{1, 2, 3\}\backslash\{k\}$, $k \in \{1, 2, 3\}$. Consequently, $a^*_{kl} \neq a^*_{lm}$, $m, k \in \{1, 2, 3\}\backslash\{l\}$, $l \in \{1, 2, 3\}$. By construction, $a^*_{12} \in f_3$, $a^*_{13} \notin f_3$, so $a^*_{12} \neq a^*_{13}$. By construction, $a^*_{23} = a^*_{13} \in f_1$, $a^*_{21} \notin f_1$, so $a^*_{21} \neq a^*_{23}$; $a^*_{32} = a^*_{12} \in f_1$, $a^*_{31} \notin f_1$, so we get $a^*_{31} \neq a^*_{32}$. Further one of two subcases is possible: either $a^*_{21} = a^*_{31}$ or $a^*_{21} \neq a^*_{31}$.

(2.1) Let $a^*_{21} = a^*_{31}$. Then the graph G_f will look like the following (Fig. 4):

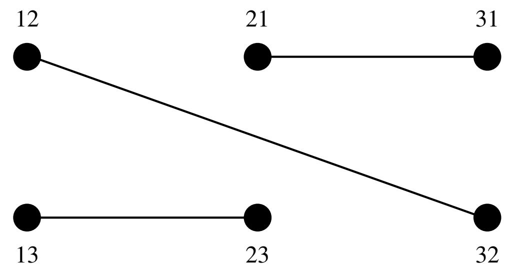

Fig. 4.

This means that $\{\{a^*_{12}, a^*_{13}\}, \{a^*_{13}, a^*_{21}\}, \{a^*_{12}, a^*_{21}\}\} \subseteq M(\{a^*_{12}, a^*_{13}, a^*_{21}\})$, and by Definition 5 $\{\{1, 2\}, \{1, 3\}, \{2, 3\}\} \prec M$. Then according to Lemmas 1, 2 there exists $w \in W(M)$ such that $Matr(w)$ is not absolutely unimodular.

(2.2) Let $a^*_{21} \neq a^*_{31}$. Then the graph G_f will look like the following (Fig. 5):

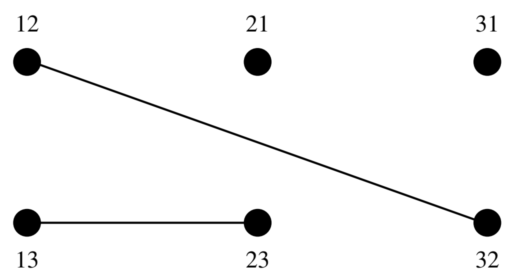

Fig. 5.

Therefore, $\{\{a^*_{12},\ a^*_{13}\},\ \{a^*_{13},\ a^*_{21}\},\ \{a^*_{12},\ a^*_{31}\}\} \subseteq M(\{a^*_{12},\ a^*_{13},\ a^*_{21},\ a^*_{31}\})$ and by Definition 5 $\{\{1, 2\}, \{2, 3\}, \{1, 4\}\} \prec M$. Then, according Lemmas 1 and 2, there exists $w \in W(M)$ such that the matrix of system of constraints for problem w is not absolutely unimodular. This completes the proof of the lemma.

Theorem 4. *Let $M \subseteq 2^{N(s)}$. In order for the class $W(M)$ to be $t_1|t_2 - equal|t_3 - edge$ reducible to class W_{Graph}, where $t_1(n), t_2(n), t_3(n) \geq n$, $n \in N$, it is necessary and sufficient that the set M is 2-embedded.*

Proof. Sufficiency automatically follows from Theorem 2.

Necessity proof. We argue by contradiction. Suppose that the class $W(M)$ is $t_1|t_2 - equal|t_3 - edge$ reducible to class W_{Graph}, but the set M is not 2-embedded. Then, by Lemma 5, there exists a problem $w \in W(M)$ such that matrix $Matr(w)$ is not absolutely unimodular. On the other hand, Theorem 1 implies that matrix $Matr(w)$ is absolutely unimodular. We get a contradiction, so our original assumption was wrong. This completes the proof of the theorem.

By Theorem 4, the resulting 2-embeddedness condition is a necessary and sufficient condition of the reducibility (according to our concept of reduction introduced above) of multiindex problems to min-cost flow problems. Moreover, it turns out that the reduction scheme for a class of multiindex problems $W(M)$ with a 2-embedded set M, proposed in the constructive proof of Theorem 2 and having linear computational costs, is optimal in the sense that

—reductions with asymptotically sublinear computational costs are impossible since we need to look through all input data;

—an arbitrarily large increase in the computational power available for the reduction does not lead to extending the class of multiindex problems reducible to the class W_{Graph}.

Thus, Theorems 2, 4 together represent an exhaustive answer to the study of $t_1|t_2-equal|t_3-edge$ reducibility of classes of multiindex problems $W(M)$ to the class of min-cost flow problems W_{Graph}.

5. CONCLUSION

In this work, we have studied $t_1|t_2-equal|t_3-edge$ reductions of multiindex transportation problems to the class of min-cost flow problems. This definition of a reduction has let us introduce a relation between variables in the original problems and arcs in the auxiliary network. Here the minimal cost flow in the auxiliary network also determines an optimal solution of the original problem such that variables are assigned the value of the flow on the corresponding arcs of the network.

We have distinguished a class of multiindex transportation problems with 2-embedded structure which we have shown to be $L|L-equal|L-edge$ reducible to the class of min-cost flow problems. Based on this reducibility, we have constructed an algorithm for solving multiindex transportation problems with 2-embedded structure. The algorithm in case of applying it to find min-cost flows with Orlin's method [27] has computational complexity $O(|E_{N(s)}|^2\log^2|E_{N(s)}|)$, where the value $|E_{N(s)}|$ equals the number of variables in the multiindex problem. This algorithm is also applicable for finding an integer-valued solution.

The main result of this work is the proof that the 2-embeddedness condition for a set $M\subseteq 2^{N(s)}$ represents a necessary and sufficient condition for the $t_1|t_2-equal|t_3-edge$ reducibility of class $W(M)$ to class W_{Graph}. This result generalizes reduction results shown previously in [1, 33] and gives (for the considered reduction scheme) an exhaustive answer to the problem of reducibility for a class of multiindex transportation problems to the class of min-cost flow problems. The presented constructive algorithm for reducing a class of multiindex problems with 2-embedded structure is optimal in the sense that reductions with asymptotically smaller computational costs are impossible, and an arbitrarily large increase in the computational costs of the reduction will not lead to extending the class of reducible multiindex problems.

Further studies may be related to specifying subclasses of multiindex problems reducible to specific subclasses of min-cost flow problems for which more efficient algorithms are known (e.g., finding a flow in a tree network [38] or in a network with planar structure [39]). It is interesting to develop new concepts of reduction that would allow to extend the applicability of flow algorithms to solving multiindex problems.

REFERENCES

1. Afraimovich, L.G. and Prilutskii, M.Kh., Multiindex Resource Distributions for Hierarchical Systems, *Autom. Remote Control*, 2006, vol. 67, no. 6, pp. 1007–1016.
2. Afraimovich, L.G. and Prilutskii, M.Kh., Multiproduct Flows in Tree-Like Networks, *Izv. Ross. Akad. Nauk, Teor. Sist. Upravlen.*, 2008, no. 2, pp. 57–63.
3. Prilutskii, M.Kh., Multicriterial Multiindex Problems of Volume Calendar Planning, *Izv. Ross. Akad. Nauk, Teor. Sist. Upravlen.*, 2007, no. 1, pp. 78–82.

4. Prilutskii, M.Kh., Multicriterial Distribution of a Homogeneous Resource in Hierarchical Systems, *Autom. Remote Control*, 1996, vol. 57, no. 2, part 2, pp. 266–271.

5. Kostyukov, V.E. and Prilutskii, M.Kh., *Resource Distribution in Hierarchical Systems. Optimization Problems for Mining and Transportation of Gas and Refining Gas Condensate*, Nizhni Novgorod: Nizhegorod. Gos. Univ., 2010.

6. Lim, A., Rodrigues, B., and Zhang, X., Scheduling Sports Competitions at Multiple Venues—Revisited, *Eur. J. Oper. Res.*, 2006, vol. 175, pp. 171–186.

7. Gunawan, A., Ng, K.M., and Poh, K.L., Solving the Teacher Assignment-Course Scheduling Problem by a Hybrid Algorithm, *Int. J. Comput. Inform. Eng.*, 2007, vol. 1, no. 2, pp. 137–142.

8. Storms, P.P.A. and Spieksma, F.C.R., An LP-Based Algorithm for the Data Association Problem in Multitarget Tracking, *Comput. Oper. Res.*, 2003, vol. 30, no. 7, pp. 1067–1085.

9. Poore, A.B., Multidimensional Assignment Formulation of Data Association Problems Arising from Multitarget and Multisensor Tracking, *Comput. Optim. Appl.*, 1994, vol. 3, no. 1, pp. 27–57.

10. Schrijver, A., *Theory of Linear and Integer Programming*, New York: Wiley, 1986. Translated under the title *Teoriya lineinogo i tselochislennogo programmirovaniya*, Moscow: Mir, 1991.

11. Papadimitrou, Ch. and Steiglitz, K., *Combinatorial Optimization: Algorithms and Complexity*, Englewood Cliffs: Prentice Hall, 1982. Translated under the title *Kombinatornaya optimizatsiya. Algoritmy i slozhnost'*, Moscow: Mir, 1985.

12. Gale, D., *The Theory of Linear Economic Models*, New York: McGraw-Hill, 1960. Translated under the title *Teoriya lineinykh ekonomicheskikh modelei*, Moscow: Mir, 1969.

13. Raskin, L.G. and Kirichenko, I.O., *Mnogoindeksnye zadachi lineinogo programmirovaniya* (Multiindex Linear Programming Problems), Moscow: Radio i Svyaz', 1982.

14. Emelichev, V.A., Kovalev, M.M., and Kravtsov, M.K., *Mnogogranniki, grafy, optimizatsiya* (Polyhedra, Graphs, and Optimization), Moscow: Nauka, 1981.

15. Bandopadhyaya, L. and Puri, M.C., Impaired Flow Multi-Index Transportation Problem with Axial Constraints, *J. Austral. Math. Soc., Ser. B*, 1988, vol. 29(3), pp. 296–309.

16. Junginger, W., On Representatives of Multi-Index Transportation Problems, *Eur. J. Oper. Res.*, 1993, vol. 66(3), pp. 353–371.

17. De Loera, J.A., Kim, E.D., Onn, S., and Santos, F., Graphs of Transportation Polytopes, *J. Combinat. Theory, Ser. A*, 2009, vol. 116, no. 8, pp. 1306–1325.

18. Kravtsov, M.A. and Krachkovskii, A.P., On Certain Properties of Three-Index Transportation Polytopes, *Diskret. Mat.*, 1999, vol. 11, no. 3, pp. 109–125.

19. Garey, M.R. and Johnson, D.S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Francisco: Freeman, 1979. Translation under the title *Vychislitel'nye mashiny i trudnoreshaemye zadachi*, Moscow: Mir, 1982.

20. Crama, Y. and Spieksma, F.C.R., Approximation Algorithms for Three-Dimensional Assignment Problems with Triangle Inequalities, *Eur J. Oper. Res.*, 1992, vol. 60, pp. 273–279.

21. Finkel'shtein, Yu.Yu., *Priblizhennye metody i prikladnye zadachi diskretnogo programmirovaniya* (Approximate Methods and Applied Problems of Discrete Programming), Moscow: Nauka, 1976.

22. Spieksma, F.C.R., Multi Index Assignment Problems: Complexity, Approximation, Applications, in *Nonlinear Assignment Problems: Algorithms and Applications*, P.M. Pardalos, L.S. Pitsoulis, Eds., Dordrecht: Kluwer, 2000, pp. 1–11.

23. Gimadi, E.Kh. and Korkishko, N.M., On One Algorithm of Solving a Three-Index Axial Assignment Problem on Single-Cycle Permutations, *Diskret. Anal. Issled. Oper., Ser. 1*, 2003, vol. 10, no. 2, pp. 56–65.

24. Gimadi, E.Kh. and Glazkov, Yu.V., On an Asymptotically Exact Algorithm for Solving One Modification of the Three-Index Planar Assignment Problem, *Diskret. Anal. Issled. Oper., Ser. 2*, 2006, vol. 13, no. 1, pp. 10–26.

25. Sergeev, S.I., New Lower Bounds for the Triplanar Assignment Problem. Use of the Classical Model, *Autom. Remote Control*, 2008, vol. 69, no. 12, pp. 2039–2060.

26. Ahuja, R.K., Magnati, T.L., and Orlin, J.B., *Network Flows: Theory, Algorithms, and Applications*, New Jersey: Prentice Hall, 1993.

27. Orlin, J.B., A Faster Strongly Polynomial Minimum Cost Flow Algorithm, *Oper. Res.*, 1993, vol. 41, no. 2, pp. 338–350.

28. Goldberg, A.V. and Rao, S., Beyond the Flow Decomposition Barrier, *J. ACM*, 1998, vol. 45, no. 5, pp. 783–797.

29. Litvak, B.G. and Rappoport, A.M., Linear Programming Problems That Allow a Network Formulation, *Ekonom. Mat. Metody*, 1970, vol. 6, no. 4, pp. 594–604.

30. Lin, Y., A Recognition Problem in Converting Linear Programming to Network Flow Models, *Appl. Math. J. Chinese Univ.*, 1993, vol. 8, no. 1, pp. 76–85.

31. Kovalev, M.M., *Matroidy v diskretnoi optimizatsii* (Matroid in Discrete Optimization), Moscow: Editorial URSS, 2003.

32. Gülpinar, N., Gutin, G., Mitra, G., and Zverovitch, A., Extracting Pure Network Submatrices in Linear Programs using Signed Graphs, *Discret. Appl. Math.*, 2004, vol. 137, no. 3, pp. 359–372.

33. Afraimovich, L.G., Three-Index Linear Programs with Nested Structure, *Autom. Remote Control*, 2011, vol. 72, no. 8, pp. 1679–1689.

34. Afraimovich, L.G., Cyclic Reducibility of Multiindex Systems of Linear Inequalities of Transportation Type, *Izv. Ross. Akad. Nauk, Teor. Sist. Upravlen.*, 2010, no. 4, pp. 83–90.

35. Chen, B., Potts, C.N., and Woeginger, G.J., A Review of Machine Scheduling. Complexity, Algorithms and Approximability, in *Handbook of Combinatorial Optimization*, New York: Kluwer, 1998, vol. 3, pp. 21–169.

36. Gofman, A.D. and Kraskal, D.B., Integer-Valued Boundary Points of Convex Polyhedra, in *Lineinye neravenstva i smezhnye voprosy* (Linear Inequalities and Adjacent Problems), Moscow: Inostrannaya Literatura, 1959, pp. 325–347.

37. Katerov, A.S., A Study of the Reducibility of Three-Index Transportation Problems to Finding a Flow in a Network with Parallel Computations, in *Prikladnaya informatika i matematicheskoe modelirovanie* (Aplied Informatics and Mathematical Modeling), Moscow: MGUP, 2011, pp. 47–57.

38. Prilutskii, M.Kh., Decomposition of a Homogeneous Resource in Hierarchical Systems with Tree-Like Structure, in *Proc. Int. Conf. "Systems Identification and Control problems" (SICPRO'2000)*, Moscow, September 26–28, 2000, Moscow: Inst. Probl. Upravlen., 2000, pp. 2038–2049.

39. Borradaile, G., Klein, P.N., Mozes, S., Nussbaum, Y., and Wulff-Nilsen, S., Multiple-Source Multiple-Sink Maximum Flow in Directed Planar Graphs in Near-Linear Time, *CoRR*, abs/1105.2228, 2011.

This paper was recommended for publication by A.A. Lazarev, a member of the Editorial Board