ISSN 0005-1179, Automation and Remote Control, 2011, Vol. 72, No. 8, pp. 1679–1689. © Pleiades Publishing, Ltd., 2011.
Original Russian Text © L.G. Afraimovich, 2011, published in Avtomatika i Telemekhanika, 2011, No. 8, pp. 109–120.

SYSTEM ANALYSIS AND OPERATIONS RESEARCH

Three-Index Linear Programs with Nested Structure

L. G. Afraimovich

Nizhni Novgorod State University, Nizhni Novgorod, Russia

Received May 11, 2010

Abstract—This paper deals with solutions of multi-index linear programs of the transportation type. The approach based on the analysis of reducibility of multi-index transportation problems to flow algorithms is taken as the main technical tool. Sufficient conditions of reducibility are proposed, which are based on the notion of nesting for the set of problem constraints. It is shown that these conditions are necessary and sufficient for reducibility of three-index problems; otherwise, the well-know hypothesis on the non-equivalence of the classes P and NP is wrong.

DOI: 10.1134/S0005117911080066

1. INTRODUCTION

A wide class of resource allocation problems can be formulated as multi-index linear programs (LPs) of the transportation type. Among such problems are the transportation problem with intermediate nodes, spatial scheduling for enterprise departments, distribution of data transmission channel capacities between the Internet providers, optimization of stock of orders, etc., [1–4]. Multi-index LPs also appear in the area of computer vision [5, 6]. In other words, the development of efficient solution methods for a particular subclass of LP problems (namely, multi-index LPs of the transportation type) enables a substantial reduction in computational burden associated with the solution of important classes of applied and theoretical problems.

To solve multi-index transportation problems, general methods of linear programming can be applied such as simplex method, Karmarkar's algorithm, etc., [7, 8]. There are also a number of works attempted at solving specifically multi-index LPs. The most well-studied is the class of two-index problems, [9]. Important classes of three- and four-index problems are considered in [10–12]. In the general formulation, multi-index problems were studied in [10]. Various conditions that facilitate the reduction of dimension and/or the number of indices in multi-index transportation problems were discussed in [13, 14]. Various geometric properties of feasible solution sets of multi-index systems of linear inequalities were obtained in [11, 15, 16].

Integer multi-index LPs of the transportation type are also of a special interest. As it is known, [9], two-index integer LPs are solvable in polynomial time. However this problem becomes NP-hard in the three-index case [17, 18]. There also exist a number of papers aimed at the development of approximate polynomial-time algorithms for integer multi-index transportation problems, [19, 20].

One of the promising lines of research in the development of efficient algorithms for multi-index LPs is the determination of subclasses of problems that admit flow methods as solution tools. Advanced studies in network optimization [21] provide a strong impact to the development of this subject area. Provided that an LP is reducible to a flow problem, the existing efficient flow algorithms [22, 23] can be applied which outperform general linear programming methods in the estimated computational complexity. Reducibility of linear programs to flow algorithms was studied in the works [24–27] which differ in the reducibility concepts adopted. Reducibility of multi-index LP transportation problems is a less elaborated issue; the reducibility of two-index problems to flow algorithms was shown in [9].

The issue of reducibility of multi-index problems with arbitrary number of indices to flow algorithms was first considered in [1] based on the literature available. The reducibility concept used in [1] is an extension of the approach proposed in [24]. A salient feature of this concept is the existence of a correspondence between the variables of the original problem and the arcs of the auxiliary network. The proposed scheme of reducibility guarantees that an arbitrary optimal flow in the auxiliary network defines an optimal solution of the original problem such that the variables are assigned the values of the flow along the corresponding arcs of the network. Using this concept, sufficient conditions of reducibility were found in [1]. Under the satisfaction of these conditions, solution of multi-index LPs of the transportation type reduces to finding a minimum cost (mincost) flow in the network with $O(m+n)$ nodes and $O(m+n)$ arcs, where n is the number of variables and m is the number of inequality constraints in the original problem.

In [28], reducibility of multi-index systems of linear inequalities was analyzed, and the concept of cyclic reducibility was proposed. This reduction scheme guarantees that the feasible flow in the auxiliary network defines a feasible solution of the original problem such that the variables are assigned the values of the flow along the corresponding cycles in the network. Under the satisfaction of the sufficient conditions of reducibility obtained in [28], solution of a multi-index system of linear inequalities can be cast into the problem of finding a feasible flow in a network with $O(sn)$ nodes and $O(sn^2)$ arcs, where n is the number of variables and s is the number of indices in the multi-index system of linear inequalities. The proposed approach is equally efficient in the analysis of systems of integer linear inequalities. Importantly, there exist special classes of systems of integer linear inequalities which satisfy the sufficient conditions of reducibility obtained in [28] and having the property that the matrix of constraints is not absolutely unimodular. As an example, one might consider the system of constraints in the NP-hard three-index axial assignment problem [29].

This work is a continuation of the research initiated in [1]; an extension of the reducibility concept for LPs is proposed. In the framework of this concept, it is shown that, if a class of LPs is reducible to the class of mincost flow problems, it possesses a number of important properties. It is also demonstrated that, for the class of three-index LPs, the sufficient conditions of reducibility proposed in [1] turn out to be necessary and sufficient ones; otherwise, the well-know hypothesis on the non-equivalence of the classes P and NP, see [17], is wrong.

2. MULTI-INDEX TRANSPORTATION PROBLEMS

In the formulation of multi-index problems of linear programming of the transportation type, we use the formalism introduced in [10]. Let $s \in N$ and $N(s) = \{1, \dots, s\}$. With every number l, we associate a parameter j_l referred to as index, which takes values in the set $J_l = \{1, \dots, n_l\}$, where $n_l \geq 2$, $l \in N(s)$. Let $f = \{k_1, \dots, k_t\} \subseteq N(s)$. The set of values of the indices $F_f = (j_{k_1}, \dots, j_{k_t})$ will be referred to as t-index, and the set of all t-indices is defined as the direct product of the sets of feasible values of the associated indices and will be denoted by $E_f = J_{k_1} \times \ldots \times J_{k_t}$. With every set F_f we associate a real number z_{F_f}, $F_f \in E_f$. Similarly to [10], this mapping of the set of t-indices E_f to the set of real numbers will be referred to as t-index matrix and defined as $\{z_{j_{k_1},\dots,j_{k_t}}\} = \{z_{F_f}\}$. Let $\overline{f} = N(s) \backslash f$. Then define by $F_{N(s)} = F_f F_{\overline{f}}$ the s-index set $(j_{k_1}, \dots, j_{k_t}, j_{k_{t+1}}, \dots, j_{k_s})$. Introduce the following notation:

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

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

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

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

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

In the sequel, given the set M and the parameters n_l, $l \in N(s)$, the matrix of constraints in the multi-index linear program (LP) (1)–(3) will be referred to as $D(M, n_1, \ldots, n_s)$. Let $D(M) = \{D(M, n_1, \ldots, n_s) | n_l \in N, n_l \geq 2, l \in N(s)\}$, be the class of matrices of constraints in the multi-index LPs defined by the given set $M \subseteq 2^{N(s)}$.

3. THE REDUCIBILITY CONCEPT

Let us formalize the reducibility concept, which will be used below in the analysis of reducibility of multi-index problems to flow algorithms.

Let $A \in R^{n \times m}$, $b, b^-, b^+ \in R^n$, $c \in R^m$ be given parameters and $x \in R^m$ be the vector of unknowns. Denote by $w(A, b, c)$ and $w(A, b^-, b^+, c)$ the linear programs $\min\{(c, x) | Ax \leq b, x \geq 0\}$ and $\min\{(c, x) | b^- \leq Ax \leq b^+, x \geq 0\}$, respectively, and let $row(A)$ and $col(A)$ denote the number of rows and columns of the matrix A. Note that the problem $w(A, b^-, b^+, c)$ can be described with the use of notation $w(A, b, c)$. However the notation $w(A, b^-, b^+, c)$ will be used whenever it is stressed that the system of constraints in the problem is represented by a set of two-sided inequalities. We also consider integer linear programs; i.e., if $w = w(A, b, c)$ is a linear program, then the corresponding integer linear program $\min \left\{(c, x) | Ax \leq b, x \in Z_+^{col(A)}\right\}$ will be denoted by w_Z. Let W be an arbitrary class of linear programs. The corresponding class of integer linear programs will be denoted by $W_Z = \{w_Z | w \in W\}$.

We next consider two classes of linear programs, W' and W''. Informally, by reducibility of the class W' to the class W'' it is meant that, for an arbitrary problem $w' \in W'$, a corresponding problem $w'' \in W''$ can be constructed such that the solution of w'' defines the solution of w'. When formalizing one or another reducibility scheme, we will determine the time required for the solution and/or specific computational procedures associated with

—the construction of the matrix of constraints in the problem w'' from the parameters of the problem w';

—the construction of the free coefficients and the coefficients of the objective function of the problem w'' from the parameters of the problem w';

—finding a solution of the problem w' from that of the problem w''.

Definition 1. The class W' is said to be $t_1|t_2|t_3$ reducible to the class W'' if for every problem $w' = w(A', b', c') \in W'$, a matrix A'' and two vectors b'', c'' can be composed in time $O(t_1)$ and $O(t_2)$, respectively, such that $w'' = w(A'', b'', c'') \in W''$ and the following conditions hold:

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

—the optimal (feasible) solution x' of the problem w' can be found in time $O(t_3)$, provided that the optimal (feasible) solution of the problem w'' is known.

The problem w'' in Definition 1 is said to correspond to the problem w' (a corresponding problem). If the functions t_1, t_2, t_3 are linear in the size of the instance w', this reducibility will be denoted by $L|L|L$; if these functions are polynomials (the degrees of these polynomials are of no interest here), the reducibility will be denoted as $P|P|P$.

To formalize specific reducibility schemes, on top of determining the time needed for the solution, it is sometimes required to define the computational procedures associated with the related constructions. In that case, the notation $t_1 - s_1|t_2 - s_2|t_3 - s_3$ will be used in the formalization of the reducibility concept, where s_1, s_2, s_3 are optional row notations for the computational procedures associated with the construction of the matrix of constraints, the free coefficients and the coefficients of the objective function, and the solution, respectively. This notation is introduced by the analogy with the notation proposed by R. Graham, which is used in the classification of problems of the scheduling theory [30].

It is interesting to understand if the class of multi-index LPs of the transportation type is reducible to the class of mincost flow problems. The corresponding classes are defined as follows.

Let $M \subseteq 2^{N(s)}$, then define by

$$W(M) = \left\{ w(A, b^-, b^+, c) | \quad b^- b^+ \in Z_+^{row(A)},\ c \in Z^{col(A)},\ A \in D(M) \right\}$$

the class of multi-index LPs of the transportation type with constraints that correspond to the given set M. In accordance with the notation introduced above, the corresponding class of multi-index problems of integer linear programming will be denoted as $W_Z(M)$.

Consider the directed graph $G = (V_G, A_G)$, $A_G \subseteq V_G^2$, where V_G and A_G are the sets of nodes and arcs of the graph G. Let l_{ij}, u_{ij} denote the capacities of the arc (i, j), e_{ij} be the cost of the arc (i, j), and let x_{ij} be an unknown value of the flow along the arc (i, j), $(i, j) \in A_G$. Then, by $v(G; l_{ij}, u_{ij}, e_{ij}, (i, j) \in A_G)$ denote the following mincost flow problem:

$$\sum_{(i,j)\in A_G} x_{ij} - \sum_{(j,i)\in A_G} x_{ji} = 0, \quad i \in V_G,$$
$$l_{ij} \le x_{ij} \le u_{ij}, \quad (i, j) \in A_G,$$
$$x_{ij} \ge 0, \quad (i, j) \in A_G,$$
$$\sum_{(i,j)\in A_G} e_{ij} x_{ij} \to \min .$$

Denote by *Graph* the set of all directed graphs and define the class of mincost flow problems as follows:

$$W_{Graph} = \{ v(G, l_{ij}, u_{ij}, e_{ij}, (i, j) \in A_G) | l_{ij}, u_{ij} \in Z_+, e_{ij} \in Z, (i, j) \in A_G, G \in Graph \} .$$

Definition 2. Let W be the class of LPs with two-sided linear inequality constraints. We say that the class W is $t_1|t_2 - equal|t_3 - edge$ reducible to the class W_{Graph}, if the class W is $t_1|t_2|t_3$ reducible to the class W_{Graph}; moreover, if $v = v(G; l_{ij}, u_{ij}, e_{ij}, (i, j) \in A_G) \in W_{Graph}$ is the problem corresponding to $w = w(A, b^-, b^+, c) \in W$, then the following conditions hold. There exist injective functions $\alpha : \{1, 2, \ldots, row(A)\} \to A_G$ and $\beta : \{1, 2, \ldots, col(A)\} \to A_G$ such that

—$l_{\alpha(i)} = b_i^-$, $u_{\alpha(i)} = b_i^+$, $i = \overline{1, row(A)}$; $l_{(u,v)} = 0$, $u_{uv} = b^*$, $(u, v) \in A_G \backslash \{\alpha(i) | i = \overline{1, row(A)}\}$, where b^* is a sufficiently large number, for definiteness, let $b^* = \sum\limits_{k=1}^{row(A)} b_k^+$;

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

—if x_{ij}, $(i, j) \in A_G$, is the optimal (feasible) solution of v, then $y = (x_{\beta(1)}, x_{\beta(2)}, \ldots, x_{\beta(col(A))})$ is the optimal (feasible) solution of w.

Therefore, by Definition 2, if the class W is $t_1|t_2 - equal|t_3 - edge$ reducible to the class W_{Graph}, $w \in W$, and $v = v(G; l_{ij}, u_{ij}, c_{ij}, (i, j) \in A_G) \in W_{Graph}$ is the problem corresponding to w, then it is guaranteed that in the construction of the mincost flow problem v, the resulting capacities and costs of the arcs are defined from the coefficients of the problem w, and the solution of w is found via a subset of the components of the solution of v. Hence, a solution algorithm can be devised for the problem w; it is based on the solution of the corresponding problem v and has 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 the mincost flow problem for the network with n nodes and m arcs. A survey on the estimates of computational complexity of known flow algorithms can be found in [22, 23]. In the exposition to follow, we consider conditions under which the class $W(M)$ is $t_1|t_2 - equal|t_3 - edge$ reducible to the class W_{Graph}.

4. REDUCIBILITY CONDITIONS FOR MULTI-INDEX PROBLEMS

The form of LP problems in the class $W(M)$ is defined by the specified set M. Hence, the goal is to find conditions imposed on the set M such that solutions of problems in the class $W(M)$ could be found via use of flow algorithms. Let us show that under the reducibility to mincost flow problems, linear programs possess certain properties. In particular, these properties will be applied to the analysis of reducibility of three-index classes $W(M)$.

Theorem 1. *If the class W is $P|P - equal|P - edge$ reducible to the class W_{Graph}, then the class W_Z of integer linear programs is decidable in polynomial time.*

Corollary. *If the class W_Z of integer linear programs is NP-hard, then the class W of linear programs is not $P|P - equal|P - edge$ reducible to the class W_{Graph}; otherwise* P = NP.

Lemma 1. *Let $M \subseteq 2^{N(s)}$. If the class $W(M)$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}, then $W(M')$ is also $P|P - equal|P - edge$ reducible to W_{Graph}, where $M' \subseteq M$.*

Lemma 2. *Let $M \subseteq 2^{N(s)}$. If the class $W(M)$ is not $P|P - equal|P - edge$ reducible to the class W_{Graph}, then the class $W(M')$ is also $P|P - equal|P - edge$ non-reducible to the class W_{Graph}, where $M \subseteq M' \subseteq 2^{N(s)}$.*

Lemma 3. *Let $M \subseteq 2^{N(s)}$. If the class $W(M)$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}, then $W(M \cup \{\varnothing\})$ is also $P|P - equal|P - edge$ reducible to W_{Graph}.*

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

Earlier, the following sufficient conditions of reducibility were found.

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

In case the set M is 2-nested, for every problem $w \in W(M)$, a corresponding problem $v = v(G; l_{ij}, u_{ij}, e_{ij}, (i,j) \in A_G) \in W_{Graph}$ can be composed by exploiting the constructive proof of Theorem 2 proposed in [1]. Moreover, it holds $|V_G| = O(|E_{N(s)}|)$, $|A_G| = O(|E_{N(s)}|)$, where the quantity $|E_{N(s)}|$ actually coincides with the number of variables in the problem w. By definition, in case of $L|L - equal|L - edge$ reducibility, the construction of the corresponding problem and finding a solution of the original problem from that of the corresponding problem requires only linear time. Let $\mu(n, m)$ be the computational complexity of the algorithm for solving the mincost flow problem in the network with n nodes and m arcs. Then, if the set M is 2-nested, any of the problems $w(A, b^-, b^+, c) \in W(M)$ is solvable in time $O(\mu(|E_{N(s)}|, |E_{N(s)}|))$.

We next consider the case of three-index problems of linear programming.

Theorem 3. *Let $M \subseteq 2^{N(s)}$, $s \leq 3$. For the class $W(M)$ to be $P|P - equal|P - edge$ reducible to the class W_{Graph}, it is necessary and sufficient that the set M is 2-nested; otherwise* P = NP.

Therefore, the 2-nesting condition is necessary and sufficient for the $P|P - equal|P - edge$ reducibility of the class of three-index LPs of the transportation type to the mincost flow class, otherwise the well-known hypothesis on the non-equivalence of the classes P and NP is not true.

5. NUMERICAL EXAMPLES

In this section, we illustrate the results obtained above via a number of multi-index linear programs of the transportation type.

Example 1. $s = 3$, $N(s) = \{1, 2, 3\}$, $M = \{\{1, 2, 3\}, \{1, 2\}, \{1, 3\}, \{3\}\}$. In that case, the problems in the class $W(M)$ are defined as follows:

$$a^- \le \sum_{j_1 \in J_1} \sum_{j_2 \in J_2} \sum_{j_3 \in J_3} x_{j_1 j_2 j_3} \le a^+;$$

$$b^-_{j_3} \le \sum_{j_1 \in J_1} \sum_{j_2 \in J_2} x_{j_1 j_2 j_3} \le b^+_{j_3}, \quad j_3 \in J_3;$$

$$c^-_{j_2} \le \sum_{j_1 \in J_1} \sum_{j_3 \in J_3} x_{j_1 j_2 j_3} \le c^+_{j_2}, \quad j_2 \in J_2;$$

$$d^-_{j_1 j_2} \le \sum_{j_3 \in J_3} x_{j_1 j_2 j_3} \le d^+_{j_1 j_2}, \quad j_1 \in J_2, \quad j_2 \in J_2;$$

$$x_{j_1 j_2 j_3} \ge 0, \quad j_1 \in J_1, \quad j_2 \in J_2, \quad j_3 \in J_3.$$

It can be seen that in this example, the set M is 2-nested, since there exists a partition $M_1 = \{\{1, 2, 3\}, \{1, 2\}\}$, $M_2 = \{\{1, 3\}, \{3\}\}$. By Theorem 2, the class $W(M)$ is $L|L - equal|L - edge$ reducible to the class W_{Graph}.

Example 2. $s = 5$, $N(s) = \{1, 2, 3, 4, 5\}$, $M = \{\{1, 2, 3, 4, 5\}, \{2, 3, 4, 5\}, \{1, 2, 3\}, \{2, 3\}, \{2\}, \{5\}\}$. It is seen that the set M is 2-nested, since there is a partition $M_1 = \{\{1, 2, 3, 4, 5\}, \{2, 3, 4, 5\}, \{5\}\}$, $M_2 = \{\{1, 2, 3\}, \{2, 3\}, \{2\}\}$. Hence, by Theorem 2, the class $W(M)$ is $L|L - equal|L - edge$ reducible to the class W_{Graph}.

Example 3. $s = 3$, $N(s) = \{1, 2, 3\}$, $M = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$. The set M is not 2-nested. In the three-index case, by Theorem 3, the 2-nesting condition is necessary and sufficient for reducibility of multi-index problems to min-cost flow problems, otherwise P = NP. Then either the class $W(M)$ is not $P|P - equal|P - edge$ reducible to the class W_{Graph}, or P = NP.

Example 4. $s = 4$, $N(s) = \{1, 2, 3, 4\}$, $M = \{\{1\}, \{2\}, \{3\}, \{4\}, \varnothing\}$. The set M is not 2-nested. It is seen that the class $W_Z(M)$ of integer linear programs is NP-hard, since the NP-hard class of planar three-index assignment problems (the NP-hardness of this class of assignment problems is proved in [18]) is reducible to $W_Z(M)$. Hence, by Corollary 1 the class $W(M)$ is not $P|P-equal|P-edge$ reducible to the class W_{Graph}; alternatively, P = NP.

6. CONCLUSION

This work is a continuation of the research initiated in [1]. A formalization of the reducibility concept of linear programs is proposed, which can be used in the classification of the results in this area. In particular, this concept is applied to the formalization of reducibility of linear programs to mincost flow problems and the solution of the initial problem is defined as a subset of the components in the solution of the flow problem.

It was shown that 2-nesting is necessary and sufficient for the $P|P-equal|P-edge$ reducibility of three-index linear programs of the transportation type to the mincost flow problems under the validity of the well-known P $\neq$ NP hypothesis. On the other hand (see [1]), the 2-nesting condition is sufficient for $L|L-equal|L-edge$ reducibility of multi-index problems to mincost flow problems. Hence, with the use of the $P|P-equal|P-edge$ reducibility concept in the three-index case, only those classes of multi-index problems can be reduced to mincost flow problems, which would be reducible with the application of the $P|P-equal|P-edge$ reducibility concept (provided P $\neq$ NP). Therefore, in the three-index case, the reduction scheme of Theorem 2 can be in a sense considered as the most efficient and comprehensive one since the growth of the computational burden associated with the reduction (in the case of polynomial complexity) does not lead to the enhancement of the class of reducible multi-index problems.

Further research relates to checking the hypothesis on the necessity and sufficiency of 2-nesting for $L|L-equal|L-edge$ reducibility in the case of arbitrary number of indices. It is also of interest to develop new concepts of reducibility of multi-index LPs to flow algorithms in order to enlarge the class of multi-index problems which admit flow algorithms as solution tools.

APPENDIX

Proof of Theorem 1. Let the conditions of the theorem be satisfied. Then for every instance problem $w \in W(M)$, the associated problem $v \in W_{Graph}$ can be constructed. By the $P|P-equal|P-edge$ reducibility, the number of nodes and arcs in the associated flow problem v depends polynomially on the size of the problem w. By Definition 2, the arc capacities in the problem v are defined via integer-valued free coefficients of the two-sided constraints in the problem w and hence, are also integer-valued.

The problem v is polynomial-time solvable (see [8]), since it is a particular case of the polynomial-time solvable linear program (specific polynomial-time algorithms for solving flow problems can be used, see [22, 23]).

It is well known that the matrix of the constraints in the problem v is absolutely unimodular, see [31]. The arc capacities in v are integer-valued; hence, a solution of v found in polynomial time (provided v is feasible) is integer-valued. Then, by Definition 2, integer-valuedness also holds for any solution of w found in polynomial time. Hence, this solution will also serve as a solution to w_Z. Therefore, any problem in the class W_Z can be solved in polynomial time. The theorem is proved.

Proof of Corollary. Assume that the class W_Z be NP-hard and the class W is $P|P-equal|P-edge$ reducible to the class W_{Graph}. Then by Theorem 1, the class W_Z is polynomial-time decidable. If the NP-hard class W_Z of optimization problems is polynomial-time decidable, then by the NP-hardness concept (e.g., see [17]) it follows that P = NP. The corollary is proved.

Proof of Lemma 1. Let the conditions of the lemma be satisfied. Consider the problem $w' = (A', b'^-, b'^+, c') \in W(M')$. Since $M' \subseteq M$, there exists a problem $w = (A, b^-, b^+, c) \in W(M)$ such that

—$col(A) = col(A')$;

—the multi-index variables of the problem w coincide with those of the problem w'; then, without loss of generality, the columns of the matrices A and A' having equal numbers can be associated with the same multi-index variables of the problems w and w';

—the problem w contains all the constraints of problem w'; then, without loss of generality, we let $A_{ij} = A'_{ij}$, $b_i^- = b_i'^-$, $b_i^+ = b_i'^+$, $i = \overline{1, row(A')}$, $j = \overline{1, col(A')}$;

—the left and right free coefficients of the constraints of w, which do not appear in the constraints of the problem w', are defined as follows: $b_i^- = 0$, $b_i^+ = \sum_{k=1}^{row(A')} b_k'^+$, $i = \overline{row(A') + 1, row(A)}$.

—$c = c'$.

Since the class $W(M)$ is $P|P{-}equal|P{-}edge$ reducible to the class W_{Graph}, by Definition 2, there exists a problem $v = v(G; l_{ij}, u_{ij}, e_{ij}, (i,j) \in A_G) \in W_{Graph}$ that corresponds to the problem w. Moreover, there exist functions $\alpha : \{1, 2, \ldots, row(A)\} \to A_G$ and $\beta : \{1, 2, \ldots, col(A)\} \to A_G$ which satisfy the conditions of Definition 2. Clearly, the problem $v \in W_{Graph}$ also corresponds to the problem w', and the related functions $\alpha' : \{1, 2, \ldots, row(A')\} \to A_G$ and $\beta' : \{1, 2, \ldots, col(A')\} \to A_G$ are defined as follows: $\alpha'(i) = \alpha(i)$, $i = \overline{1, row(A')}$, $\beta'(i) = \beta(i)$, $i = \overline{1, col(A')}$. Hence, by Definition 2 the class $W(M')$ is $t_1|t_2 - equal|t_3 - edge$ reducible to the class W_{Graph}, where the functions t_1, t_2, t_3 are polynomials in the size of the instance problem w.

Note that the functions t_1, t_2, t_3 which are used in the reduction of the problem w' to v are indeed polynomials in the size of the instance problem w, since they are defined by the functions exploited in the reduction of w to v. Below, it is shown that these functions can as well be considered as polynomials in the size of the instance w'.

There exists a set of values $n_l \in N$, $n_l \geq 2$, $l \in N(s)$, such that $A \in D(M, n_1, \ldots, n_s)$, $A' \in D(M', n_1, \ldots, n_s)$. The number of variables in w and w' is equal to $col(A) = col(A') = n_1 \ldots n_s$, and the numbers of constraints in w and w', which are equal to $row(A)$ and $row(A')$, respectively, are bounded from above by $row(D(2^{N(s)}, n_1, \ldots, n_s))$. Moreover, we have

$$row\left(D(2^{N(s)}, n_1, \ldots, n_s)\right) = \sum_{i=0}^{s} \sum_{\{j_1,\ldots,j_i\} \in 2^{N(s)}} n_{j_1} \ldots n_{j_s}$$

$$\leq \sum_{i=0}^{s} \sum_{\{j_1,\ldots,j_i\} \in 2^{N(s)}} n_1 \ldots n_s = n_1 \ldots n_s \sum_{i=0}^{s} C_i^s = n_1 \ldots n_s 2^s \leq (n_1 \ldots n_s)^2.$$

Therefore,

$$n_1 \ldots n_s \leq row(A)col(A) \leq (n_1 \ldots n_s)^3,$$
$$n_1 \ldots n_s \leq row(A')col(A') \leq (n_1 \ldots n_s)^3.$$

In other words, letting h and h' denote the sizes of the instance problems w and w', respectively, we obtain $h = O(h'^3)$. Hence, the functions t_1, t_2, t_3 are polynomial in the size of the instance w', which shows that the class $W(M')$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}. The lemma is proved.

Proof of Lemma 2. Let us prove Lemma 2 by contradiction. Assume that $W(M)$ is not $P|P - equal|P - edge$ reducible to the class W_{Graph}, and there exists a set M' such that $M \subseteq M' \subseteq 2^{N(s)}$ and $W(M')$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}. Then by Lemma 1, the class $W(M)$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}. Hence, we arrive at the contradiction, which proves the lemma.

Proof of Lemma 3. Let the conditions of the lemma be satisfied. If $\varnothing \in M$, then $M \cup \{\varnothing\} = M$, and the lemma is proved. Let $\varnothing \notin M$ and consider the problem $w' = w(A', b'^-, b'^+, c') \in W(M \cup \{\varnothing\})$. Since $M \subseteq M \cup \{\varnothing\}$, there exists a problem $w = w(A, b^-, b^+, c) \in W(M)$ such that

—$col(A) = col(A')$, $\quad row(A) + col(A') = row(A')$;

—the multi-index variables of the problem w coincide with those of the problem w'; then, without loss of generality, the columns of the matrices A, A' having equal numbers can be associated with the same multi-index variables of the problems w and w';

—the problem w' contains all the constraints of the problem w; then, without loss of generality, we let $A_{ij} = A'_{ij}$, $b_i^- = b_i'^-$, $b_i^+ = b_i'^+$, $i = \overline{1, row(A)}$, $j = \overline{1, col(A)}$;

—$c = c'$.

Moreover, without loss of generality, we assume that

—$A'_{row(A)+i,i} = 1, \quad i = \overline{1, col(A')}$,

—$A'_{row(A)+i,j} = 0, \quad j \in \overline{1, col(A')}\backslash\{i\}, \quad i = \overline{1, col(A')}$.

In other words, the constraints in the problem w', which are defined by the element $\varnothing \in M \cup \{\varnothing\}$ and represent two-sided constraints on the variables of the problem w', are specified by the lower rows of the matrix A'.

Since $W(M)$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}, Definition 2 implies the existence of a problem $v = v(G; l_{ij}, u_{ij}, e_{ij}, (i,j) \in A_G) \in W_{Graph}$, which corresponds to the problem w. Moreover, there exists a pair of functions α, β that satisfy the conditions of Definition 2. We next schematically design the problem $v' \in W_{Graph}$ that corresponds to the problem w'. The only difference between the problems w' and w is that the former one contains two-sided constraints on the variables, and the function β defines the arcs of the problem v, which are associated with the variables of the original problem. Hence, each of these arcs can be replaced with the pair of consecutive arcs such that the first one corresponds to the variable, and the second one to the two-sided constraint on this variable. Therefore, to construct $v' = v(G'; l'_{ij}, u'_{ij}, e'_{ij}, (i,j) \in A_{G'})$, we appropriately modify the problem v. Namely, let initially $G' = G$; then, for each of the arcs (u,v) such that $\beta(i) = (u,v)$, we transform the graph G' as follows: $V_{G'} := V_{G'} \cup \{p_i\}$, $A_{G'} := A_{G'}\backslash\{(u,v)\} \cup \{(u,p_i),(p_i,v)\}$, where p_i is a new node and $i = \overline{1, col(A)}$. Let us define the functions $\alpha' : \{1, 2, \ldots, row(A')\} \to A_{G'}$, $\beta' : \{1, 2, \ldots, col(A')\} \to A_{G'}$ as follows:

$$\alpha'(i) = \begin{cases} \alpha(i), \text{ if } \alpha(i) \neq \beta(j), \text{ for all } j = \overline{1, col(A)} \\ (u, p_j), \text{ if there exists } j \in \overline{1, col(A)}, \text{ such that } \alpha(i) = \beta(j) = (u,v), \end{cases} \quad i = \overline{1, row(A)};$$

$$\alpha'(row(A) + i) = (p_i, v), \quad \text{where} \quad \beta(i) = (u,v), \quad i = \overline{1, col(A')};$$
$$\beta'(i) = (u, p_i), \quad \text{where} \quad \beta(i) = (u,v), \quad i = \overline{1, col(A')}.$$

The functions α', β' thus specified and the graph G' define the problem v', which corresponds to the problem w'. Hence, the class $W(M \cup \{\varnothing\})$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}. The lemma is proved.

Proof of Theorem 3. For $s \leq 2$, any set $M \subseteq 2^{N(s)}$ is 2-nested and by Theorem 2, the class $W(M)$ is $L|L - equal|L - edge$ reducible to the class W_{Graph}, hence, it is $P|P - equal|P - edge$ reducible to the class W_{Graph}.

Let $s = 3$. If the set M is 2-nested then by Theorem 2, the class $W(M)$ is $L|L - equal|L - edge$ reducible to the class W_{Graph}, hence, it is $P|P - equal|P - edge$ reducible to the class W_{Graph}. Next, let the set M be 2-reducible; then, at least one of the following two conditions is satisfied: $\{\{1\},\{2\},\{3\}\} \subseteq M$ or $\{\{1,2\},\{1,3\},\{2,3\}\} \subseteq M$. Assume that the class $W(M)$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}. Then by Lemma 1, the class $W(\{\{1\},\{2\},\{3\}\})$ or the class $W(\{\{1,2\},\{1,3\},\{2,3\}\})$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}. Hence, by Lemma 3, the class $W(\{\varnothing, \{\{1\},\{2\},\{3\}\})$ or the class $W(\{\varnothing, \{1,2\},\{1,3\},\{2,3\}\})$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}. Therefore, by Theorem 1, at least one of the classes of integer linear programs $W_Z(\{\varnothing, \{1\},\{2\},\{3\}\})$ or $W_Z(\{\varnothing, \{1,2\},\{1,3\},\{2,3\}\})$ is polynomial-time decidable. On the other hand, the problem classes $W_Z(\{\varnothing, \{1\},\{2\},\{3\}\})$ and $W_Z(\{\varnothing, \{1,2\},\{1,3\},\{2,3\}\})$ are NP-hard, since they contain the planar three-index assignment problem and the axial three-index assignment problem, respectively, as particular cases, both known to be NP-hard (e.g., see [29]). It then follows that if the class $W(M)$ is $P|P - equal|P - edge$ reducible to the class W_{Graph}, then P = NP. The theorem is proved.

REFERENCES

1. Afraimovich, L.G. and Prilutskii, M.Kh., Multiindex Resource Distributions for Hierarchical Systems, *Autom. Remote Control*, 2006, no. 6, pp. 1007–1016.
2. Afraimovich, L.G. and Prilutskii, M.Kh., Multi-commodity Flows in Tree Networks, *Izv. Ross. Akad. Nauk, Teor. Sist. Upravlen.*, 2008, no. 2, pp. 57–63.
3. Prilutskii, M.Kh., Multicriterial Multi-index Problems of Spatial Scheduling, *Izv. Ross. Akad. Nauk, Teor. Sist. Upravlen.*, 2007, no. 1, pp. 78–82.
4. Prilutskii, M.Kh., Multicriteria Distribution of a Homogeneous Resource in Hierarchical System, *Autom. Remote Control*, 1996, no. 2, pp. 266–271.
5. Pusztaszeri, J., Rensing, P., and Liebling, T., Tracking Elementary Particles Near their Primary Vertex: A Combinatorial Approach, *J. Global Optim.*, 1996, vol. 9, no. 1, pp. 41–64.
6. 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.
7. Schrijver, A., *Theory of Linear and Integer Programming*, New York: Wiley, 1986. Translated under the title *Teoriya lineinogo i tselochislennogo programmirovaniya*, Moscow: Mir, 1991.
8. 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.
9. Gale, D., *The Theory of Linear Economic Models*, New York: McGraw-Hill, 1960. Translated under the title *Teoriya lineinykh ekonomicheskikh modelei*, Moscow: Mir, 1969.
10. Raskin, L.G. and Kirichenko, I.O., *Mnogoindeksnye zadachi lineinogo programmirovaniya* (Multi-index Problems of Linear Programming), Moscow: Radio i Svyaz', 1982.
11. Emelichev, V.A., Kovalev, M.M., and Kravtsov, M.K., *Mnogogranniki, grafy, optimizatsiya* (Polyhedra, Graphs, Optimization), Moscow: Nauka, 1981.
12. Bandopadhyaya, L. and Puri, M.C., Impaired Flow Multi-index Transportation Problem with Axial Constraints, *J. Austral. Math. Soc., Ser. B*, 1988, vol. 29, no. 3, pp. 296–309.
13. Junginger, W., On Representatives of Multi-index Transportation Problems, *Eur. J. Oper. Res.*, 1993, vol. 66, no. 3, pp. 353–371.
14. Verkhovskii, B.S., Multidimensional Linear Programs of the Transportation Type, *Dokl. Akad. Nauk SSSR*, 1963, vol. 151, no. 3, pp. 515–518.
15. 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.
16. Kravtsov, M.A. and Krachkovskii, A.P., On the Properties of Three-index Transportation Polyhedra, *Discrete Math.*, 1999, vol. 11, no. 3, pp. 109–125.
17. 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.
18. Frieze, A.M., Complexity of a 3-dimensional Assignment Problem, *Eur. J. Oper. Res.*, 1983. vol. 13, no. 2, pp. 161–164.
19. 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.
20. Huang, G. and Lim, A., A Hybrid Genetic Algorithm for the Three-Index Assignment Problem, *Eur. J. Oper. Res.*, 2006, vol. 172, pp. 249–257.
21. Ahuja, R.K., Magnati, T.L., and Orlin, J.B., *Network Flows: Theory, Algorithms, and Applications*, Upper Saddle River: Prentice Hall, 1993.

22. Galil, Z. and Tardos, E., An $O(n^2(m + n \log n) \log n)$ Min-cost Flow Algorithm, *J. ACM*, 1988, vol. 35, no. 2, pp. 374–386.

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

24. Litvak, B.G. and Rappoport, A.M., Linear Programs that Admit for a Network Formulation, *Ekon. Mat. Metody*, 1970, vol. 6, no. 4, pp. 594–604.

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

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

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

28. Afraimovich, L.G., Cycle-based Reducibility of Multi-index Systems of Linear Inequalities of the Transportation Type, *Izv. Ross. Akad. Nauk, Teor. Sist. Upravlen.*, 2010, no. 4, pp. 83–90.

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

30. Chen, B., Potts, C.N., and Woeginger, G.J., A Review of Machine Scheduling: Complexity, Algorithms and Approximability, in *Handbook of Combinatorial Optimization*, Du, D.Z. and Pardalos, P.M., Eds., Dordrecht: Kluwer, 1998, pp. 21–169.

31. Hoffman, A.D. and Kruskal, J.B., *Tselochislennye granichnye tochki vypuklykh mnogogrannikov. Lineinye neravenstva i smezhnye voprosy* (Integral Boundary Points of Convex Polyhedra. Linear Inequalities and Related Systems). Moscow: Inostrannaya Literatura, 1959, pp. 325–347.

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