ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2023, Vol. 84, No. 2, pp. 190–210.

OPTIMIZATION, SYSTEM ANALYSIS, AND OPERATIONS RESEARCH

A Makespan-Optimal Schedule for Processing Jobs with Possible Operation Preemptions as an Optimal Mixed Graph Coloring

Y. N. Sotskov

United Institute of Informatics Problems, National Academy of Sciences of Belarus, Minsk, Belarus
e-mail: sotskov48@mail.ru

Received November 4, 2021
Revised August 15, 2022
Accepted 29, 2022

Abstract—This paper establishes a relationship between optimal scheduling problems with the minimum schedule length and the problems of finding optimal (strict) colorings of mixed graph vertices, i.e., assigning a minimal set of ordered colors to the vertices $V = \{v_1, \ldots, v_{|V|}\}$ of a mixed graph $G = (V, A, E)$ so that the vertices v_i and v_j incident to an edge $[v_i, v_j] \in E$ will have different colors and the color of the vertex v_k in an arc $(v_k, v_l) \in A$ will be not greater (smaller) than that of the vertex v_l. As shown below, any optimal coloring problem for the vertices of a mixed graph G can be represented as the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ of constructing a makespan-optimal schedule for processing a partially ordered set of jobs with integer durations p_{ij} of their operations with possible preemptions. In contrast to classical scheduling problems, executing an operation in the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ may require several machines and, besides the two types of precedence relations defined on the set of operations, unit-time operations of a given subset must be executed simultaneously. The problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ is pseudopolynomially reduced to the problem of finding an optimal coloring of the vertices of a mixed graph G (the input data of the scheduling problem). Due to the assertions proved, the results obtained for the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ have analogs for the corresponding optimal coloring problems for the vertices of a mixed graph G, and vice versa.

Keywords: schedule, preemption, makespan optimality, mixed graph, optimal coloring

DOI: 10.25728/arcRAS.2023.29.68.001

1. INTRODUCTION

Production scheduling requires constructing optimal schedules for processing jobs on the available equipment (machines or processors). Optimization of production schedules is an important factor in production efficiency due to reducing production costs and job completion times and ensuring the timely supply of raw materials and necessary components to the production process.

In practice, production scheduling problems are diverse, both in terms of the conditions, constraints, and intended purpose of production and the goals achieved by implementing the constructed schedules. As a rule, special schedule optimization algorithms are developed to solve production scheduling problems considering the conditions of a particular production process. The applicability of scheduling algorithms to production scheduling can be expanded using models of more complex processing systems to uniformly represent different classes of scheduling problems and develop, based on such models, general methods for constructing optimal schedules.

As is known, constructing makespan-optimal schedules with unit-time operations is equivalent to finding optimal colorings of graph vertices. In addition to precedence relations defined on the set of operations, it may be necessary to consider the impossibility of joint execution of operations on the same equipment (machines). In this case, optimal schedules can be constructed using the colorings of mixed graph vertices. They were introduced in [1, 2].

Let $G = (V, A, E)$ denote a finite mixed graph with a non-empty vertex set $V = \{v_1, \ldots, v_{|V|}\}$, an arc set A, and an edge set E. Each arc $(v_i, v_j) \in A$ defines an ordered pair of vertices v_i and v_j, whereas each edge $[v_p, v_q] \in E$ defines an unordered pair of vertices v_p and v_q. Assume that the mixed graph under consideration $G = (V, A, E)$ contains no multiple arcs, multiple edges, and loops. If the set A is empty, we obtain a graph $(V, \emptyset, E)$. If the set E is empty, we have a directed graph $(V, A, \emptyset)$.

In [1], the colorings of a mixed graph (its vertices) were defined as follows.

Definition 1 [1]. An integer function $c : V \to \{1, \ldots, t\}$ is called a coloring $c(G)$ of a mixed graph $G = (V, A, E)$ if $c(v_i) \leqslant c(v_j)$ for each arc $(v_i, v_j) \in A$ and $c(v_p) \neq c(v_q)$ for each edge $[v_p, v_q] \in E$. A coloring $c(G)$ is optimal if it uses the minimum number $t =: \chi(G)$ of different colors $c(v_i) \in \{1, \ldots, t\}$. The value $\chi(G)$ is called the chromatic number of the mixed graph G.

If $A = \emptyset$, then a coloring $c(G)$ is a common coloring of the vertices of a graph $G = (V, \emptyset, E)$. In contrast to the coloring of a graph $(V, \emptyset, E)$ existing for any graph, the coloring $c(G)$ of a mixed graph $G = (V, A, E)$ with non-empty arc and edge sets may not exist. The following criterion for the existence of a coloring $c(G)$ of a mixed graph $G = (V, A, E)$ was proved in [1].

Theorem 1 [1]. *A coloring $c(G)$ of a mixed graph $G = (V, A, E)$ exists iff the directed subgraph $(V, A, \emptyset)$ of the same G contains no circuit with a pair of vertices adjacent in the subgraph $(V, \emptyset, E)$.*

A mixed graph $G = (V, A, E)$ will be called colorable if there exists a coloring $c(G)$ for it. Finding an optimal coloring $c(G)$ of a mixed graph is an NP-hard problem even if $A = \emptyset$ [3]. The papers [4–8] considered the relationship between the problems of finding optimal mixed graph colorings and optimal scheduling problems with the minimum schedule length (the minimum total time for processing jobs) and unit-time operations. The results published on mixed graph colorings and equivalent optimal scheduling problems with unit-time operations were surveyed in [9].

This paper demonstrates that an optimal coloring of any colorable mixed graph is found by constructing a makespan-optimal schedule for executing a partially ordered set of integer-time operations with possible preemptions. In contrast to classical scheduling problems, executing an operation in the problem under consideration may require several machines. Besides the two types of precedence relations defined on the set of operations, in this problem, unit-time operations of a given subset must be executed simultaneously. Due to the assertions proved, the results obtained for the makespan-optimal scheduling problems have analogs for the corresponding optimal coloring problems for the vertices of mixed graphs, and vice versa.

2. OPTIMAL SCHEDULES FOR PROCESSING JOBS WITH DIFFERENT ROUTES AND STRICT MIXED GRAPH COLORINGS

The considerations below involve the terminology of the monographs [10, 11] on graph theory and the monographs [12, 13] on scheduling theory.

To classify scheduling problems, we adopt the three-position notation $\alpha|\beta|\gamma$ introduced in [14], where α is the type of the processing system and the number of machines (processors), β is the characteristics of processed jobs (tasks), and γ is the objective function. Scheduling problems are classified using the parameters given in [13].

2.1. Unit-time Operations and a Strict Coloring of Mixed Graphs

We begin with posing the problem $J|p_{ij} = 1|C_{\max}$ of constructing a makespan-optimal schedule for processing a set $\mathcal{J} = \{J_1, \ldots, J_{|\mathcal{J}|}\}$ of jobs with different routes and the unit durations

$p_{ij} = 1$ of all given operations Q_{ij}. (Such an optimality criterion is denoted by $C_{\max}$, and such a multistage processing system is called a job-shop.) In the problem $J|p_{ij} = 1|C_{\max}$, the job set $\mathcal{J}$ must be processed optimally on the set $\mathcal{M} = \{M_1, \ldots, M_{|\mathcal{M}|}\}$ of dedicated (different) machines. A job $J_i \in \mathcal{J}$ is processed by executing a set $\mathcal{Q}_i = \{Q_{i,1}, \ldots, Q_{i,|\mathcal{Q}_i|}\}$ of operations in a given order: $(Q_{i,1}, \ldots, Q_{i,|\mathcal{Q}_i|})$. Each operation $Q_{ij} \in \mathcal{Q}_i$ must be executed on a dedicated machine $M_{\mu(i,j)}$ of the set $\mathcal{M} \ni M_{\mu(i,j)}$.

All jobs of the set $\mathcal{J} = \{J_1, \ldots, J_{|\mathcal{J}|}\}$ are ready to be processed at the initial time $t = 0$ of the planning horizon and preemptions are prohibited when executing any operation $Q_{ij} \in \mathcal{Q}_i$ of each job $J_i \in \mathcal{J}$. Hence, an admissible schedule for processing the jobs of the set $\mathcal{J} = \{J_1, \ldots, J_{|\mathcal{J}|}\}$ is uniquely defined by the times to start $S(Q_{ij}) \geqslant 0 = t$ or complete $C(Q_{ij}) = S(Q_{ij}) + p_{ij}$ all operations $Q_{ij} \in \mathcal{Q} := \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{Q}_i$.

Let a subset $\mathcal{Q}^{(k)}$ of the set $\mathcal{Q}$ consist of all the operations to be executed on the machine $M_k \in \mathcal{M}$. *Any pair of operations from the set* $\mathcal{Q}^{(k)}$ *may not be executed simultaneously when implementing an admissible schedule.*

From the above statement of the problem of scheduling theory, it follows that an admissible schedule for the problem $J|p_{ij} = 1|C_{\max}$ must define $|\mathcal{M}|$ linear strict orders of sets of operations $\mathcal{Q}^{(k)}$ on machine $M_k \in \mathcal{M}$.

According to this problem statement, an admissible schedule for the problem $J|p_{ij} = 1|C_{\max}$ defines $|\mathcal{M}|$ linear strict orders of executing the operation sets $\mathcal{Q}^{(k)}$ on dedicated machines $M_k \in \mathcal{M}$; moreover, the makespan-optimal schedule

$$\Big\{C(Q_{1,1}), \ldots, C(Q_{1,|\mathcal{Q}_1|}), \ldots, (C(Q_{|\mathcal{J}|,1}), \ldots, C(Q_{|\mathcal{J}|,|\mathcal{Q}_{|\mathcal{J}|}|})\Big\} =: \mathbf{S} \tag{1}$$

has the minimum length $C_{\max} := \max\{C_1, \ldots, C_{|\mathcal{J}|}\}$ among all admissible schedules for processing the job set $\mathcal{J}$. Throughout this paper, C_i denotes the completion time of a job $J_i \in \mathcal{J}$, i.e., $C_i = C(Q_{i,|\mathcal{Q}_i|})$, where $Q_{i,|\mathcal{Q}_i|}$ is the last operation of the job J_i.

When solving the problem $\alpha|\beta|C_{\max}$, an optimal schedule can be found in the set of semi-active schedules [12]: there exists a makespan-optimal schedule that is semi-active.

Definition 2. An admissible schedule for the problem $\alpha|\beta|C_{\max}$ is said to be semi-active if the execution of any operation from the set $\mathcal{Q}$ can not be started earlier without violating the order of operations in the schedule $\mathbf{S}$ or (and) another operation will be executed later than in the schedule $\mathbf{S}$.

A strict mixed graph coloring was defined in [2] as follows.

Definition 3 [2]. An integer function $c_< : V \to \{1, \ldots, t\}$ is called a strict coloring $c_<(G)$ of a mixed graph $G = (V, A, E)$ if

$$c_<(v_i) < c_<(v_j) \tag{2}$$

for each arc $(v_i, v_j) \in A$ and $c_<(v_p) \neq c_<(v_q)$ for each edge $[v_p, v_q] \in E$. A strict coloring $c_<(G)$ is optimal if it uses the minimum number $t =: \chi_<(G)$ of different colors $c_<(v_i) \in \{1, \ldots, t\}$.

A strict coloring $c_<(G)$ can be interpreted as a special case of a coloring $c(G)$ of a mixed graph $G = (V, A, E)$; see Definition 1.

Remark 1. A coloring $c(G)$ can be used instead of a strict coloring $c_<(G)$ for any mixed graph $G = (V, A, E)$ in which

$$(v_i, v_j) \in A \Rightarrow [v_i, v_j] \in E \tag{3}$$

for each arc $(v_i, v_j) \in A$.

If a mixed graph $G = (V, A, E)$ has an arc $(v_i, v_j) \in A$ without the implication (3), then we add the edge $[v_i, v_j]$ in the graph G for each such arc (v_i, v_j). Therefore, any strict coloring $c_<(G)$

of a mixed graph G can be represented as a coloring $c(G^+)$ of the mixed graph $G^+ = (V, A, E^+)$ obtained by adding all such edges.

The following corollary of Theorem 1 gives a criterion for the existence of a strict coloring $c_<(G)$ of a mixed graph $G = (V, A, E)$.

Corollary 1. *There exists a strict coloring $c_<(G)$ of a mixed graph $G = (V, A, E)$ iff the directed subgraph $(V, A, \emptyset)$ of the same mixed graph $G = (V, A, E)$ contains no circuits.*

The next result for the problem $J|p_{ij} = 1|C_{\max}$ was established in [4].

Theorem 2 [4]. *The problem $J|p_{ij} = 1|C_{\max}$ is equivalent to the problem of finding an optimal strict coloring $c_<(G)$ of a mixed graph $G = (V, A, E)$ with $V = \mathcal{Q}$ and the following conditions:*

(a) $(\mathcal{Q}, A, \emptyset) = \bigcup_{i=1}^{|\mathcal{J}|}(\mathcal{Q}_i, A_i, \emptyset)$, *where each directed subgraph $(\mathcal{Q}_i, A_i, \emptyset)$ of the mixed graph $G = (V, A, E)$ is a path passing through all vertices of the set $\mathcal{Q}_i$ and $\mathcal{Q}_i \cap \mathcal{Q}_j = \emptyset$ for all $i \neq j$.*

(b) $(\mathcal{Q}, \emptyset, E) = \bigcup_{k=1}^{|\mathcal{M}|}(\mathcal{Q}^{(k)}, \emptyset, E^{(k)})$, *where each subgraph $(\mathcal{Q}^{(k)}, \emptyset, E^{(k)})$ of the mixed graph $G = (V, A, E)$ is a complete graph and $\mathcal{Q}^{(k)} \cap \mathcal{Q}^{(l)} = \emptyset$ for all $k \neq l$.*

Considering Remark 1, Theorem 2 yields the following fact.

Corollary 2. *The problem $J|p_{ij} = 1|C_{\max}$ is equivalent to the problem of finding an optimal coloring $c(G)$ of a mixed graph $G = (V, A, E)$ with $V = \mathcal{Q}$ and conditions* (a), (b), *and* (3).

2.2. Preemptions of Integer-Time Operations

In this subsection, Theorem 2 is generalized to the problem $J|[p_{ij}], pmtn|C_{\max}$ of constructing a makespan-optimal schedule for processing the job set $\mathcal{J}$ with integer durations $p_{ij} \geqslant 1$ of the operations $Q_{ij} \in \mathcal{Q}$ with possible preemptions. In the problem notation, *pmtn* indicates the possibility of operation preemptions, whereas $[p_{ij}]$ means the integer durations of the operations. Due to possible preemptions of operations from the set $\mathcal{Q}$, the set of semi-active schedules becomes wider. In many cases, this fact may complicate finding a makespan-optimal schedule. On the other hand, the preemptions of all or some operations of the set $\mathcal{Q}$ may reduce the length $C_{\max}$ of the optimal semi-active schedule. Hence, for many problems $J|[p_{ij}], pmtn|C_{\max}$, it is desirable to limit the number of times with possible preemptions of operations from the set $\mathcal{Q}$ without losing the semi-active schedule of the smallest length $C_{\max}$.

In the sequel, an optimal schedule for the problem $J|[p_{ij}], pmtn|C_{\max}$ will be restricted to the set of semi-active schedules with possible operation preemptions at integer times only. Such a reduction of the solution set in the problem $J|[p_{ij}], pmtn|C_{\max}$ rests on the following remark.

Remark 2. All jobs of the set $\mathcal{J} = \{J_1, \ldots, J_{|\mathcal{J}|}\}$ are ready to be processed at the initial time $t = 0$ and all operations $Q_{ij} \in \mathcal{Q}$ have integer durations $p_{ij} \geqslant 1$. Therefore, there exists an optimal semi-active schedule in which the preemptions of operations occur at integer times only.

Remark 2 is valid since the preemption of an operation $Q_{ij} \in \mathcal{Q}$ may reduce the length of a semi-active schedule **S** without preempted operations only if this preemption occurs at the completion time of at least one operation. (In this case, another operation $Q_{uv} \in \mathcal{Q}$, $u \neq i$, can be started on the available machine $M_{\mu(i,j)} = M_{\mu(u,v)} \in \mathcal{M}$.) All jobs of the set $\mathcal{J} = \{J_1, \ldots, J_{|\mathcal{J}|}\}$ are ready to be processed at the time $t = 0$ and all operations $Q_{ij} \in \mathcal{Q}$ have integer durations $p_{ij} \geqslant 1$. Hence, in a semi-active schedule **S** without operation preemptions, the execution of any operation can complete at an integer time only.

In view of Remark 2, we partition the operation $Q_{1,1} \in \mathcal{Q}_1$ of an integer duration into $p_{1,1}$ unit-time operations, denoting them by $\{v_1, \ldots, v_{p_{1,1}}\}$.

Here and elsewhere, the unit-time operations are supposed linearly ordered and must be executed in ascending order of their numbers during the implementation of any admissible schedule.

Let $\{v_{p_{1,1}+1}, \ldots, v_{p_{1,1}+p_{1,2}}\}$ denote the set of unit-time operations obtained by partitioning the next operation $Q_{1,2} \in \mathcal{Q}_1$ of the job J_1. By analogy, we partition the operations of the set $\mathcal{Q}_1 \setminus \{Q_{1,1}, Q_{1,2}\}$ into unit-time operations and assign sequential numbers to them. For example, the last operation $Q_{1,|\mathcal{Q}_1|} \in \mathcal{Q}_1$ of the job J_1 will be partitioned into the following unit-time operations: $\left\{v_{\sum_{j=1}^{|\mathcal{Q}_1|-1} p_{1,j}+1}, \ldots, v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}}\right\}$.

This partition procedure of all operations from the set $\mathcal{Q}_1$ yields the set

$$\mathcal{W}_1 = \left\{v_1, \ldots, v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}}\right\} \tag{4}$$

of all linearly ordered unit-time operations of the first job $J_1 \in \mathcal{J}$ in the problem $J|[p_{ij}], pmtn|C_{\max}$.

Consider the next unit-time operation $v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+1}$, sequentially numbering all unit-time operations into which the operations of the set $\mathcal{Q}_2 = \{Q_{2,1}, \ldots, Q_{2,|\mathcal{Q}_2|}\}$ of the second job $J_2 \in \mathcal{J}$ are partitioned. As a result, we obtain the set

$$\mathcal{W}_2 = \left\{v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+1}, \ldots, v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+\sum_{j=1}^{|\mathcal{Q}_2|} p_{2,j}}\right\} \tag{5}$$

of all linearly ordered unit-time operations of the job $J_2 \in \mathcal{J}$.

Following this technique, we sequentially number the unit-time operations of the jobs from the set $\mathcal{J} \setminus \{J_1, J_2\}$ in an ascending order of the job numbers and the numbers of the resulting unit-time operations.

At the last stage of the partition procedure for the integer-time operations of the set $\mathcal{J}$, we construct the set

$$\mathcal{W}_{|\mathcal{J}|} = \left\{v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_{|\mathcal{J}|-1}|} p_{|\mathcal{J}|-1,j}+1}, \ldots, v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_{|\mathcal{J}|}|} p_{|\mathcal{J}|,j}}\right\} \tag{6}$$

of all linearly ordered unit-time operations of the last job $J_{|\mathcal{J}|}$ from the set $\mathcal{J}$.

Thus, by Remark 2, all jobs of the set $\mathcal{J}$ are processed by executing the set $\mathcal{W} := \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{W}_i$ of unit-time operations.

Let $\mathcal{J}^{(k)}$ denote a subset of jobs $\mathcal{J} \supseteq \mathcal{J}^{(k)}$ with operations executed on a machine $M_k \in \mathcal{M}$. If $J_i \in \mathcal{J}^{(k)}$, the subset $\mathcal{Q}_i^{(k)}$ of the set $\mathcal{Q}_i \supseteq \mathcal{Q}_i^{(k)}$ contains all operations of the job J_i executed on a machine $M_k \in \mathcal{M}$.

2.3. Example 1 of the Problem $J5|[p_{ij}], pmtn|C_{\max}$

We reduce the problem $J|[p_{ij}], pmtn|C_{\max}$ to finding an optimal strict coloring $c_<(G)$ of a mixed graph G for Example 1 of the problem $J5|[p_{ij}], pmtn|C_{\max}$ with the job set $\mathcal{J} = \{J_1, J_2, J_3\}$ and the machine set $\mathcal{M} = \{M_1, \ldots, M_5\}$. The input data of this example are presented in Table 1.

Table 1. The input data of Example 1 of problem $J5|[p_{ij}], pmtn|C_{\max}$

Operations $Q_{1,j}$ of job J_1	$Q_{1,1}$	$Q_{1,2}$	$Q_{1,3}$	$Q_{1,4}$	–
Machines $M_{\mu(1,j)}$	M_1	M_2	M_3	M_4	–
Durations $p_{1,j}$ of operations $Q_{1,j}$	2	4	2	1	–
Operations $Q_{2,j}$ of job J_2	$Q_{2,1}$	$Q_{2,2}$	$Q_{2,3}$	$Q_{2,4}$	$Q_{2,5}$
Machines $M_{\mu(2,j)}$	M_2	M_5	M_1	M_2	M_4
Durations $p_{2,j}$ of operations $Q_{2,j}$	3	2	2	3	1
Operations $Q_{3,j}$ of job J_3	$Q_{3,1}$	$Q_{3,2}$	$Q_{3,3}$	$Q_{3,4}$	$Q_{3,5}$
Machines $M_{\mu(3,j)}$	M_5	M_3	M_1	M_5	M_3
Durations $p_{3,j}$ of operations $Q_{3,j}$	2	1	1	3	1

Let us construct a mixed graph $G = (V, A, E)$ defining all input data of Example 1 in the network form.

The job J_1 consists of four operations $Q_{1,1}$, $Q_{1,2}$, $Q_{1,3}$, and $Q_{1,4}$ with the durations $p_{1,1} = 2$, $p_{1,2} = 4$, $p_{1,3} = 2$, and $p_{1,4} = 1$, respectively. Using the notations (4), we obtain the linearly ordered set of unit-time operations $\mathcal{W}_1 = \{v_1, \dots, v_9\}$ and include them in the desired vertex set $V \supset \mathcal{W}_1$. All operations of the job J_1 and all unit-time operations yielded by partitioning each operation from the set $\mathcal{Q}_1$ are linearly ordered when implementing any admissible schedule. Therefore, we include the following arc set into the mixed graph $G = (V, A, E) : \mathcal{A}_1 = \{(v_1, v_2), (v_2, v_3), \dots, (v_8, v_9)\}$, where $\mathcal{A}_1 \subset A$.

The job J_2 consists of five operations $Q_{2,1}$, $Q_{2,2}$, $Q_{2,3}$, $Q_{2,4}$, and $Q_{2,5}$ with the durations $p_{2,1} = 3$, $p_{2,2} = 2$, $p_{2,3} = 2$, $p_{2,4} = 3$, and $p_{2,5} = 1$, respectively. Using the notations (5), we obtain the set of unit-time operations $\mathcal{W}_2 = \{v_{10}, \dots, v_{20}\}$ and include them into the vertex set $V \supset \mathcal{W}_2$. Since all operations of the job J_2 are linearly ordered when implementing any admissible schedule, we include the following arc set into the mixed graph $G = (V, A, E) : \mathcal{A}_2 = \{(v_{10}, v_{11}), (v_{11}, v_{12}), \dots, (v_{19}, v_{20})\}$, where $\mathcal{A}_2 \subset A$.

The job J_3 consists of five operations $Q_{3,1}$, $Q_{3,2}$, $Q_{3,3}$, $Q_{3,4}$, and $Q_{3,5}$ with the durations $p_{3,1} = 2$, $p_{3,2} = 1$, $p_{3,3} = 1$, $p_{3,4} = 3$, and $p_{3,5} = 1$, respectively. Using the notations (6), we obtain the set of unit-time operations $\mathcal{W}_3 = \{v_{21}, \dots, v_{28}\}$ and include them into the vertex set $V \supset \mathcal{W}_3$. Since all operations of the job J_3 are linearly ordered when implementing any admissible schedule, we include the following arc set into the mixed graph $G = (V, A, E) : \mathcal{A}_3 = \{(v_{21}, v_{22}), (v_{22}, v_{23}), \dots, (v_{27}, v_{28})\}$, where $\mathcal{A}_3 \subset A$.

Thus, the directed subgraph $(V, A, \emptyset) = \bigcup_{i=1}^{|\mathcal{J}|} (\mathcal{W}_i, \mathcal{A}_i, \emptyset)$, $|\mathcal{J}| = 3$, of the desired mixed graph (V, A, E) has the vertex set

$$V = \mathcal{W} := \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{W}_i \tag{7}$$

and the arc set

$$A = \mathcal{A} := \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{A}_i. \tag{8}$$

We sequentially construct the edge set E of the mixed graph $G = (V, A, E)$ for each set $\mathcal{Q}^{(k)} = \bigcup_{J_i \in \mathcal{J}^{(k)}} \mathcal{Q}_i^{(k)}$ of operations executed on each machine $M_k \in \{M_1, \dots, M_5\}$.

For the machine M_1, the set $\mathcal{Q}^{(1)}$ is partitioned into two unit-time operations $\{v_1, v_2\}$ of the job J_1, two unit-time operations $\{v_{15}, v_{16}\}$ of the job J_2, and one unit-time operation v_{24} of the job J_1. None of the pairs of operations from the set $\mathcal{Q}^{(1)}$ may be executed simultaneously. Hence, it is necessary to construct a complete tripartite graph $(V_1, \emptyset, E_1)$ in which $V_1 = \{v_1, v_2; , v_{15}, v_{16}; v_{24}\}$ and $E_1 = \{[v_1, v_{15}], [v_1, v_{16}], [v_2, v_{15}], [v_2, v_{16}]; [v_1, v_{24}], [v_2, v_{24}]; [v_{15}, v_{24}], [v_{16}, v_{24}]\}$. *Here and elsewhere, the vertices belonging to different parts of the k-partite graph and the edges incident to such vertices are separated by the symbol ";" (semicolon).*

For the machine M_2, the set $\mathcal{Q}^{(2)}$ is partitioned into four unit-time operations $\{v_3, v_4, v_5, v_6\}$ of the job J_1 and six unit-time operations $\{v_{10}, v_{11}, v_{12}, v_{17}, v_{18}, v_{19}\}$ of the job J_2. None of the pairs of operations from the set $\mathcal{Q}^{(2)}$ may be executed simultaneously. Hence, it is necessary to construct a complete bipartite graph $(V_2, \emptyset, E_2)$ in which $V_2 = \{v_3, v_4, v_5, v_6; v_{10}, v_{11}, v_{12}, v_{17}, v_{18}, v_{19}\}$ and $|E_2| = |\mathcal{Q}_1^{(2)}| \cdot |\mathcal{Q}_2^{(2)}| = 4 \cdot 6 = 24$.

For the machine M_3, the set $\mathcal{Q}^{(3)}$ is partitioned into two unit-time operations $\{v_7, v_8\}$ of the job J_1 and two unit-time operations $\{v_{23}, v_{28}\}$ of the job J_3. A pair of operations from the set $\mathcal{Q}^{(3)}$

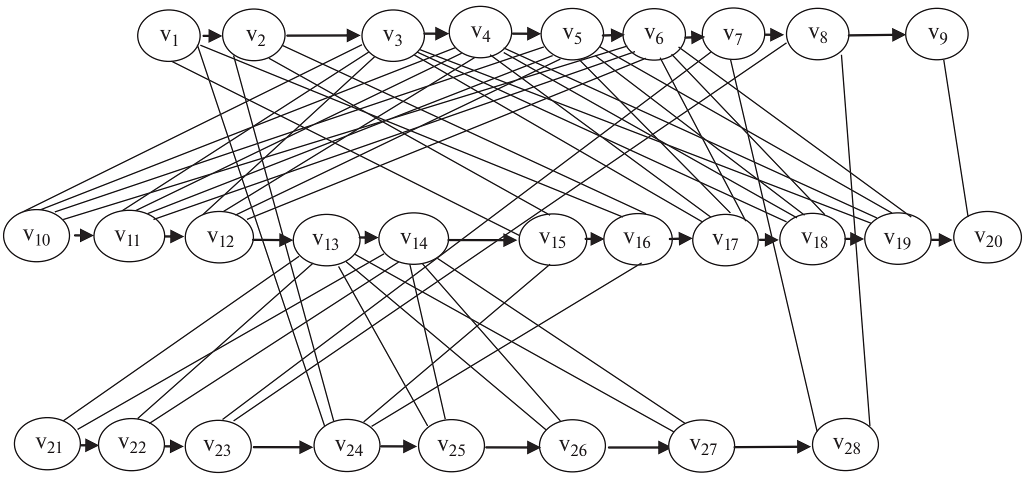

Fig. 1. Mixed graph $G = (V, A, E)$ defining all input data of Example 1 of problem $J5|[p_{ij}], pmtn|C_{\max}$.

may not be executed simultaneously. Hence, it is necessary to construct a complete bipartite graph $(V_3, \emptyset, E_3)$ in which $V_3 = \{v_7, v_8;\ v_{23}, v_{28}\}$ and $|E_3| = |\mathcal{Q}_1^{(3)}| \cdot |\mathcal{Q}_3^{(3)}| = 2 \cdot 2 = 4$.

For the machine M_4, the set $\mathcal{Q}^{(4)}$ is partitioned into one unit-time operation v_9 of the job J_1 and one unit-time operation v_{20} of the job J_2. The operations from the set $\mathcal{Q}^{(4)}$ may not be executed simultaneously. Hence, it is necessary to connect the operations v_9 and v_{20} with an edge. Thereby, we construct a trivial complete bipartite graph $(V_4, \emptyset, E_4)$ in which $V_4 = \{v_9;\ v_{20}\}$ and $E_4 = \{[v_9, v_{20}]\}$.

For the machine M_5, the set $\mathcal{Q}^{(5)}$ is partitioned into two unit-time operations $\{v_{13}, v_{14}\}$ of the job J_2 and five unit-time operations $\{v_{21}, v_{22}, v_{25}, v_{26}, v_{27}\}$ of the job J_3. None of the pairs of operations from the set $\mathcal{Q}^{(5)}$ may be executed simultaneously. Hence, it is necessary to construct a complete bipartite graph $(V_5, \emptyset, E_5)$ in which $V_5 = \{v_{13}, v_{14};\ v_{21}, v_{22}, v_{25}, v_{26}, v_{27}\}$ and $|E_5| = |\mathcal{Q}_2^{(5)}| \cdot |\mathcal{Q}_3^{(5)}| = 2 \cdot 5 = 10$.

Thus, the subgraph $(V, \emptyset, E) = \bigcup_{k=1}^{|\mathcal{M}|}(V_k, \emptyset, E_k)$, $|\mathcal{M}| = 5$, of the desired mixed graph $G = (V, A, E)$ has the edge set

$$E = \mathcal{E} := \bigcup_{i=1}^{|\mathcal{M}|} \mathcal{E}_i. \tag{9}$$

Each subgraph $(V_k, \emptyset, E_k)$ of the mixed graph G is a complete $|\mathcal{J}^{(k)}|$-partite graph and $V_k \cap V_l = \emptyset$ for all $k \neq l$.

Figure 1 presents the mixed graph $G = (V, A, E)$ defining all input data of Example 1. Let us demonstrate that an optimal strict coloring $c_<(G)$ of the mixed graph G can be found by solving Example 1.

Note that $p_{ij} = 1$ for all operations $Q_{ij} \in \mathcal{Q}$ in the problem $J|p_{ij} = 1|C_{\max}$ (subsection 2.1). Therefore, the schedule (1) for the problem $J|p_{ij} = 1|C_{\max}$ is determined by the completion times of all unit-time operations $\mathcal{Q}$ of all jobs from the set $\mathcal{J}$.

For the problem $J|[p_{ij}], pmtn|C_{\max}$ with possible preemptions of integer-time operations, an admissible schedule for processing jobs from the set $\mathcal{J} = \{J_1, \ldots, J_{|\mathcal{J}|}\}$ is determined by the completion times $C(v_j)$ of all unit-time operations $v_j \in \mathcal{W} = \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{W}_i$. The value p_{ij} is the integer duration of the operation Q_{ij}. Considering Remark 2, besides the set (1) of the completion times of all integer-time operations $\mathcal{Q}$ of jobs from the set $\mathcal{J}$, we have to determine the completion times of all unit-time operations from the set $\mathcal{W}$. Therefore, in the problem $J|[p_{ij}], pmtn|C_{\max}$ with possible preemptions of integer-time operations, an admissible schedule $\mathbf{S}$ is given by the set

$$\{C(v_1), \ldots, C(v_{|\mathcal{W}|})\} =: \mathbf{S}. \tag{10}$$

As is easily verified, for all vertices $v_i \in \mathcal{W}$ of the mixed graph $G = (V, A, E)$ in Example 1, we have

$$C(v_i) = c_<(v_i). \tag{11}$$

Due to equalities (10) and (11), the makespan-optimal schedule $\mathbf{S} = \{C(v_1) = c_<(v_1), \ldots, C(v_{28}) = c_<(v_{28})\}$ in Example 1 is determined by the following optimal strict coloring $c_<(G)$ of the mixed graph G (Fig. 1):

$$\begin{gathered}
c_<(v_1) = 1, \quad c_<(v_2) = 2, \quad c_<(v_3) = 4, \quad c_<(v_4) = 5, \quad c_<(v_5) = 6, \\
c_<(v_6) = 7, \quad c_<(v_7) = 8, \quad c_<(v_8) = 9, \quad c_<(v_9) = 10, \quad c_<(v_{10}) = 1, \\
c_<(v_{11}) = 2, \quad c_<(v_{12}) = 3, \quad c_<(v_{13}) = 4, \quad c_<(v_{14}) = 5, \quad c_<(v_{15}) = 6, \\
c_<(v_{16}) = 7, \quad c_<(v_{17}) = 8, \quad c_<(v_{18}) = 9, \quad c_<(v_{19}) = 10, \quad c_<(v_{20}) = 11, \\
c_<(v_{21}) = 1, \quad c_<(v_{22}) = 2, \quad c_<(v_{23}) = 3, \quad c_<(v_{24}) = 4, \quad c_<(v_{25}) = 6, \\
c_<(v_{26}) = 7, \quad c_<(v_{27}) = 8, \quad c_<(v_{28}) = 10.
\end{gathered}$$

We calculate the optimal schedule length in Example 1:

$$C_{\max} = \max\{C_1, C_2, C_3\} = \max\{c_<(v_9), c_<(v_{20}), c_<(v_{28})\} = \max\{10, 11, 10\} = 11.$$

The strict coloring $c_<(G)$ is optimal according to the following considerations: there exists a path $(v_{10}, \ldots, v_{20})$ of length 11 in the subgraph $(V, A, \emptyset)$ of the constructed mixed graph $G = (V, A, E)$; hence, $\chi_<(G) \geqslant 11$.

2.4. The Problem $J|[p_{ij}], pmtn|C_{\max}$ and the Corresponding Problem of Finding a Strict Coloring of Mixed Graph Vertices with Complete k-partite Subgraphs

Using the introduced notations, we formulate the following result based on condition (a) of Theorem 2.

Theorem 3. *The problem $J|[p_{ij}], pmtn|C_{\max}$ is pseudopolynomially reduced to the problem of finding an optimal strict coloring $c_<(G)$ of a mixed graph $G = (V, A, E)$ with the vertex set $V = \mathcal{W}$, condition* (a) *and the additional condition*

(c) $(V, \emptyset, E) = \bigcup_{k=1}^{|\mathcal{M}|} (V_k, \emptyset, E_k)$, *where each graph $(V_k, \emptyset, E_k)$ is a complete $|\mathcal{J}^{(k)}|$-partite graph and $V_k \cap V_l = \emptyset$ for all $k \neq l$.*

Proof. For an arbitrary problem $J|[p_{ij}], pmtn|C_{\max}$, we prove the existence of a mixed graph $G = (V, A, E)$ with the vertex set $V = \mathcal{W}$ and an optimal strict coloring determining an optimal schedule for the problem $J|[p_{ij}], pmtn|C_{\max}$ under conditions (a) and (c).

According to Remark 2, an optimal schedule for the problem $J|[p_{ij}], pmtn|C_{\max}$ can be found in the class of semi-active schedules with possible preemptions at integer times only. Hence, the problem $J|[p_{ij}], pmtn|C_{\max}$ can be represented as the problem $J|p_{ij} = 1|C_{\max}$ of constructing an optimal schedule for processing a given set $\mathcal{W} = \{v_1, \ldots, v_{|\mathcal{W}|}\} = \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{W}_i$ of unit-time jobs. The job set $\mathcal{W}$ for the problem $J|p_{ij} = 1|C_{\max}$ is obtained by the sequential partition of the operation sets $\mathcal{Q}_i$ of all jobs $J_i \in \mathcal{J}$ in the problem $J|[p_{ij}], pmtn|C_{\max}$ into unit-time operations. For details, see subsection 2.2: formulas (4), (5), and (6) for the operations of the sets $\mathcal{Q}_1$, $\mathcal{Q}_2$, and $\mathcal{Q}_{|\mathcal{J}|}$, respectively.

Obviously, Theorem 3 follows from Theorem 2: condition (b) for a mixed graph $G = (V, A, E)$ with the vertex set $V = \mathcal{Q}$ turns into condition (c) for the mixed graph $(\mathcal{W}, \mathcal{A}, \mathcal{E})$ with the vertex set $\mathcal{W} = \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{W}_i$ (formula (7)), the arc set $\mathcal{A} = \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{A}_i$ (formula (8)), and the edge set $\mathcal{E} = \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{E}_i$ (formula (9)). The proof of Theorem 3 is complete.

In view of Remark 1, we arrive at the following result.

Corollary 3. *The problem* $J|[p_{ij}], pmtn|C_{\max}$ *is pseudopolynomially reduced to the problem of finding an optimal coloring* $c(G)$ *of a mixed graph* $G = (V, A, E)$ *with conditions* (a) *and* (c) *and* $V = \mathcal{W}$.

3. AN OPTIMAL SCHEDULE FOR MULTI-PROCESSOR OPERATIONS AND THE CORRESPONDING COLORING OF MIXED GRAPH VERTICES

In Section 2, we have provided Theorem 2 and proved Theorem 3 on the reduction of the classical scheduling problems $J|p_{ij} = 1|C_{\max}$ and $J|[p_{ij}], pmtn|C_{\max}$ to the problems of finding optimal strict colorings $c_<(G)$ of mixed graphs $G = (V, A, E)$ with conditions (a) and (b) and conditions (a) and (c), respectively. For an arbitrary mixed graph $G = (V, A, E)$, the problem of finding an optimal strict coloring $c_<(G)$ is not generally reduced to the problem $J|p_{ij} = 1|C_{\max}$ or the problem $J|[p_{ij}], pmtn|C_{\max}$. This fact can be easily verified.

The problem of finding an optimal coloring $c(G)$ of any mixed graph $G = (V, A, E)$ can be reduced to some generalizations of the problems $J|p_{ij} = 1|C_{\max}$ and $J|[p_{ij}], pmtn|C_{\max}$. We present them below.

3.1. Unit-Time Durations of Multiprocessor Operations

In the classical problems $J|\beta|C_{\max}$ of scheduling theory, each operation is executed on one machine from the set $\mathcal{M}$. In contrast, processing systems with multiprocessor operations (jobs) may require either a single machine or several dedicated machines from the set $\mathcal{M}$ to execute an operation $Q_{ij} \in \mathcal{Q}$ over the entire duration $p_{ij} \geqslant 0$ [13]. Like in all scheduling problems $\alpha|\beta||\gamma$, when implementing any admissible schedule, no pair of operations requiring at least one common machine $M_k \in \mathcal{M}$ may be executed simultaneously.

The monograph [13, pp. 264–283] described the results of studying the problems $GMPT||C_{\max}$, where MPT denotes MultiProcessor Tasks and G is a processing system (a general shop) with arbitrary precedence relations defined on the set of multiprocessor operations $\mathcal{Q}$. In the problem $GMPT|p_{ij} = 1|C_{\max}$, the completion time $C(Q_{ij})$ of an operation $Q_{ij} = v_{k_{ij}}$ must precede the starting time $S(Q_{rq})$ of an operation $Q_{rq} = v_{k_{rq}}$. Such a completion–start precedence relation of operations will be written as $v_{k_{ij}} \to v_{k_{rq}}$. Obviously, the mixed graph $G = (V, A, E)$ defining the input data of the problem $GMPT|p_{ij} = 1|C_{\max}$ must contain an arc $(v_{k_{ij}}, v_{k_{rq}})$ and an edge $[v_{k_{ij}}, v_{k_{rq}}]$ as well; see Definition 1.

Note that intensive research on the problems $GMPT|\beta|\gamma$ has been ongoing for several decades [15–23] since such problems arise in many real production scheduling systems. The research results on the problems $GMPT|\beta|\gamma$ before 1996 were presented in the survey [19]. The problems $GMPT|p_{ij} = 1|\gamma$ with unit-time multiprocessor operations were considered in [19–23].

Consider the problem $G_cMPT|p_{ij} = 1|C_{\max}$, which is equivalent to the problem of finding an optimal coloring $c(G)$ of a mixed graph $G = (V, A, E)$; for details, see [24]. The problem $GMPT|p_{ij} = 1|C_{\max}$ is a special case of the problem $G_cMPT|p_{ij} = 1|C_{\max}$, whereas the problem $J|p_{ij} = 1|C_{\max}$ is a special case of the problem $GMPT|p_{ij} = 1|C_{\max}$.

The problem $G_cMPT|p_{ij} = 1|C_{\max}$ differs from the problem $GMPT|p_{ij} = 1|C_{\max}$ [13, pp. 264–268] in the following:

1) Besides the completion–start precedence relations $v_{k_{ij}} \to v_{k_{rq}}$, the problem $G_cMPT|p_{ij} = 1|C_{\max}$ may include start–start precedence relations on the operation set $\mathcal{Q}$. (In other words, the starting time $S(Q_{ij})$ of an operation $Q_{ij} = v_{k_{ij}}$ must precede the starting time $S(Q_{rq})$ of an operation $Q_{rq} = v_{k_{rq}}$.)

2) Also, the problem $G_cMPT|p_{ij} = 1|C_{\max}$ may include the subsets of unit-time operations $\{v_{h_1}, \ldots, v_{h_{|V(h)|}}\} =: V(h)$ of the set V that must be executed simultaneously in any admissible schedule.

Let us describe the start–start precedence relations as a generalization of the problem $J|p_{ij} = 1|C_{\max}$ provided that the input data of this problem are represented as a mixed graph $G = (V, A, E)$. (Hence, the problem $J|p_{ij} = 1|C_{\max}$ is reduced to the problem of finding an optimal coloring of the mixed graph G; see Corollary 2.)

When describing the precedence relations on the set $\mathcal{Q}$, we use the notations $v_{k_{ij}} \in \mathcal{W}$ of unit-time operations $Q_{ij} \in \mathcal{Q}$, which have been introduced in subsection 2.2 for integer-time operations of the problem $J|[p_{ij}], pmtn|C_{\max}$. This is acceptable since all operations of the problem $J|p_{ij} = 1|C_{\max}$ have unit durations. A bijection between the elements of the sets $\mathcal{Q}$ and $\mathcal{W}$ is defined by equalities (4), (5), ..., (6) for the operations of the jobs $J_1, J_2, \ldots, J_n \in \mathcal{J}$, respectively.

Let the problem $G_cMPT|p_{ij} = 1|C_{\max}$ require that the completion time $C(Q_{ij})$ of an operation $Q_{ij} = v_{k_{ij}}$ precede the starting time $S(Q_{rq})$ of an operation $Q_{rq} = v_{k_{rq}}$. Then the mixed graph $G = (V, A, E)$ defining the input data of the problem $G_cMPT|p_{ij} = 1|C_{\max}$ must contain the arc $(v_{k_{ij}}, v_{k_{rq}})$ and the edge $[v_{k_{ij}}, v_{k_{rq}}]$ as well; see Definition 1. In addition to defining the completion–start relation $v_{k_{ij}} \to v_{k_{rq}}$, the problem $G_cMPT|p_{ij} = 1|C_{\max}$ may require that the starting time $S(Q_{uv})$ of an operation $Q_{uv} = v_{k_{uv}}$ precede the starting time $S(Q_{el})$ of an operation $Q_{el} = v_{k_{el}}$. Then the mixed graph $G = (V, A, E)$ defining the input data of the problem $G_cMPT|p_{ij} = 1|C_{\max}$ contains the arc $(v_{k_{uv}}, v_{k_{el}})$ and does not contain the edge $[v_{k_{uv}}, v_{k_{el}}]$. Such a start–start precedence relation will be written as $v_{k_{uv}} \mapsto v_{k_{el}}$.

In addition to the precedence relation $v_{k_{ij}} \to v_{k_{rq}}$ and $v_{k_{uv}} \mapsto v_{k_{el}}$, the problem $G_cMPT|p_{ij} = 1|C_{\max}$ may require that a given subset of unit-time operations $\{v_{h_1}, \ldots, v_{h_{|V(h)|}}\} =: V(h)$ of the set V be executed simultaneously in any admissible schedule. To define such a condition, the directed subgraph $(V, A, \emptyset)$ of the mixed graph $G = (V, A, E)$ must contain a circuit $(v_{h_1}, v_{h_2}, \ldots, v_{h_{|V(h)|}}, v_{h_1})$, i.e., the set A must contain the following subset of arcs:

$$\left\{ \left(v_{h_1}, v_{h_2}\right), \left(v_{h_2}, v_{h_3}\right), \ldots, \left(v_{h_{|V(h)|-1}}, v_{h_{|V(h)|}}\right), \left(v_{h_{|V(h)|}}, v_{h_1}\right) \right\} \subseteq A.$$

Consider the problem $G_cMPT|p_{ij} = 1|C_{\max}$ with w given subsets $V(1), \ldots, V(w)$ of unit-time operations. The operations of each subset $V(h) = \{v_{h_1}, \ldots, v_{h_{|V(h)|}}\} \subseteq V$ must be executed simultaneously in any admissible schedule, $h \in \{1, \ldots, w\}$. Then the directed subgraph $(V, A, \emptyset)$ of the mixed graph $G = (V, A, E)$ must contain the following subset of arcs:

$$\mathcal{A}_0 = \bigcup_{h=1}^{w} \left\{ \left(v_{h_1}, v_{h_2}\right), \left(v_{h_2}, v_{h_3}\right), \ldots, \left(v_{h_{|V(h)|-1}}, v_{h_{|V(h)|}}\right), \left(v_{h_{|V(h)|}}, v_{h_1}\right) \right\}. \tag{12}$$

Since the mixed graph $G = (V, A, E)$ defines the input data of an individual problem (example) $G_cMPT|p_{ij} = 1|C_{\max}$, this example will be called the problem $G_cMPT|p_{ij} = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$. Unlike classical scheduling problems $J|p_{ij} = 1|C_{\max}$ and $J|[p_{ij}], pmtn|C_{\max}$, which have solution under any input data, there are examples of $G_cMPT|p_{ij} = 1|C_{\max}$ without admissible schedules. The following criterion for the existence of an admissible schedule in the problem $G_cMPT|p_{ij} = 1|C_{\max}$ on a mixed graph $G = (V, A, E)$ was proved in [24, p. 76].

Theorem 4 [24]. *There exists an admissible solution of the problem $G_cMPT|p_{ij} = 1|C_{\max}$ on a mixed graph $G = (V, A, E)$ iff the directed subgraph $(V, A, \emptyset)$ of the same mixed graph $G = (V, A, E)$ contains no circuit with adjacent vertices of the subgraph $(V, \emptyset, E)$.*

The next result was also established in [24, p. 76].

Lemma 1 [24]. *The solvable problem $G_cMPT|p_{ij} = 1|C_{\max}$ on a mixed graph $G = (V, A, E)$ is equivalent to the problem of finding an optimal coloring $c(G)$ of the same mixed graph $G = (V, A, E)$.*

As is easily verified, not all problems $G_cMPT|p_{ij}=1|C_{\max}$ are reduced to optimal strict colorings $c_<(G)$ of mixed graphs $G=(V,A,E)$. Indeed, the strict inequality (2) cannot be used to define the start–start precedence relation $v_{k_{ij}} \mapsto v_{k_{rq}}$; hence, the directed subgraph $(V,A,\emptyset)$ of the mixed graph G with a strict coloring $c_<(G)$ contains no circuits. (For clarity, Theorems 1 and 4 should be compared with Corollary 1.)

3.2. Preemptions of Integer-Time Multiprocessor Operations

Theorem 5 generalizes Lemma 1 to the problem $G_cMPT|[p_{ij}],pmtn|C_{\max}$ of finding a makespan-optimal schedule for processing the job set $\mathcal{J}$ with integer durations $p_{ij} \geqslant 1$ of all operations $Q_{ij} \in \mathcal{Q}$ with their possible preemptions.

Theorem 5. *The solvable problem $G_cMPT|[p_{ij}],pmtn|C_{\max}$ is pseudopolynomially reduced to the problem of finding an optimal coloring $c(G)$ of a mixed graph $G=(V,A,E)$. For any colorable mixed graph $G=(V,A,E)$, there exists a problem $G_cMPT|[p_{ij}],pmtn|C_{\max}$ on the same mixed graph $G=(V,A,E)$ that is equivalent to the problem of finding an optimal coloring $c(G)$ of the mixed graph $G=(V,A,E)$.*

Proof. Consider the solvable problem $G_cMPT|[p_{ij}],pmtn|C_{\max}$ of finding a makespan-optimal schedule for processing the job set $\mathcal{J}=\{J_1,\ldots,J_{|\mathcal{J}|}\}$ on given dedicated machines $\mathcal{M}=\{M_1,\ldots,M_{|\mathcal{M}|}\}$. Let us construct a mixed graph $G=(V,A,E)$ whose optimal coloring determines the solution of this problem.

Due to Remark 2, we will find a makespan-optimal schedule for the solvable problem $G_cMPT|[p_{ij}],pmtn|C_{\max}$ in the class of semi-active schedules with possible preemptions at integer times only. To this effect, we define the set of unit-time operations $\mathcal{W}=\{v_1,\ldots,v_{|\mathcal{W}|}\}=\bigcup_{i=1}^{|\mathcal{J}|}\mathcal{W}_i$ by partitioning sequentially the sets of integer-time operations $\mathcal{Q}_i$ of all jobs $J_i \in \mathcal{J}$ in the problem $G_cMPT|[p_{ij}],pmtn|C_{\max}$ into unit-time operations (see subsection 2.2): equality (4) defines the subset $\mathcal{W}_1$ of unit-time operations for the operations $\mathcal{Q}_1$ of the first job, whereas equality (5) defines the subset $\mathcal{W}_2$ of unit-time operations for the operations $\mathcal{Q}_2$ of the second job. Similar equalities for the sets $\mathcal{Q}_3,\ldots,\mathcal{Q}_{|\mathcal{J}|-1}$ sequentially define the subsets $\mathcal{W}_3,\ldots,\mathcal{W}_{|\mathcal{J}|-1}$ of unit-time operations. For example, equality (6) defines the subset $\mathcal{W}_{|\mathcal{J}|}$ of unit-time operations for the operations $\mathcal{Q}_{|\mathcal{J}|}$ of the last job $\mathcal{J}_{|\mathcal{J}|}$. The sequential partition of all integer-time operations into unit-time operations yields a subgraph $(\mathcal{W},\emptyset,\emptyset)$ of the desired mixed graph $G=(V,A,E)$ with the vertex set $V=\mathcal{W}=\bigcup_{i=1}^{|\mathcal{J}|}\mathcal{W}_i$.

In any semi-active schedule, a job $J_i \in \mathcal{J}$ consists of a linearly ordered set $\mathcal{Q}_i \subset \mathcal{Q}$ of integer-time operations, which is associated with a linearly ordered set of unit-time operations of the form

$$\mathcal{W}_i=\left\{v_{\sum_{j=1}^{|\mathcal{Q}_1|}p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_{i-1}|}p_{i-1,j}+1},\ldots,v_{\sum_{j=1}^{|\mathcal{Q}_1|}p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_i|}p_{i,j}}\right\}.$$

The arc set

$$\mathcal{A}_i=\left\{\left(v_{\sum_{j=1}^{|\mathcal{Q}_1|}p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_{i-1}|}p_{i-1,j}},\ v_{\sum_{j=1}^{|\mathcal{Q}_1|}p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_{i-1}|}p_{i-1,j}+1}\right),\ldots,\right.$$
$$\left.\left(v_{\sum_{j=1}^{|\mathcal{Q}_1|}p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_i|}p_{i,j}},\ v_{\sum_{j=1}^{|\mathcal{Q}_1|}p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_i|}p_{i,j}+1}\right)\right\}$$

and the edge set

$$\mathcal{E}_i = \left\{ \left[v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_{i-1}|} p_{i-1,j}},\ v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_{i-1}|} p_{i-1,j}+1} \right], \ldots, \left[v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_i|} p_{i,j}},\ v_{\sum_{j=1}^{|\mathcal{Q}_1|} p_{1,j}+\ldots+\sum_{j=1}^{|\mathcal{Q}_i|} p_{i,j}+1} \right] \right\}$$

determine a linear order of executing the operation set $\mathcal{W}_i$ when implementing an admissible schedule for the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$.

In addition to the completion–start precedence relations between the operations of the set $\mathcal{Q}_i$ for processing the same job $J_i \in \mathcal{J}$, we assume that the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ includes the following sets:

—the set $\mathcal{R}_{\rightarrow}$ of completion–start relations between the operations of different jobs, given by

$$\mathcal{R}_{\rightarrow} = \{v_{r_1} \to v_{r_2}, \ldots, v_{r_{n-1}} \to v_{r_n}\}; \tag{13}$$

—the set $\mathcal{R}_{\mapsto}$ of start–start precedence relations between the operations of different jobs, given by

$$\mathcal{R}_{\mapsto} = \{v_{l_1} \mapsto v_{l_2}, \ldots, v_{l_{m-1}} \to v_{l_m}\}. \tag{14}$$

According to Definition 1, we introduce the precedence relations (13) by adding the arc set $\mathcal{A}_{|\mathcal{J}|+1} := \{(v_{r_1}, v_{r_2}), \ldots, (v_{r_{n-1}}, v_{r_n})\}$ and the edge set $\mathcal{E}_{|\mathcal{J}|+1} := \{[v_{r_1}, v_{r_2}], \ldots, [v_{r_{n-1}}, v_{r_n}]\}$ in the desired mixed graph $G = (V, A, E)$. For introducing the precedence relations (14), it suffices to add the arc set $\mathcal{A}_{|\mathcal{J}|+2} := \{(v_{l_1}, v_{l_2}), \ldots, (v_{l_{m-1}}, v_{l_m})\}$ in the mixed graph $G = (V, A, E)$.

In the problem $G_cMPT|p_{ij} = 1|C_{\max}$, we can also define subsets $V(1), \ldots, V(w)$ of unit-time operations from the set $\mathcal{Q}$ such that all operations of the subset $V(h) = \{v_{h_1}, \ldots, v_{h_{|V(h)|}}\} \subseteq V$ must be executed simultaneously in any admissible schedule, $h \in \{1, \ldots, w\}$. In the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ with such a condition, the set of precedence relations $\mathcal{R}_{\mapsto}$ (14) must contain the following subset of precedence relations: $\bigcup_{h=1}^{w} \left\{ v_{h_1} \mapsto v_{h_2}, v_{h_2} \mapsto v_{h_3}, \ldots, v_{h_{|V(h)|-1}} \mapsto v_{h_{|V(h)|}}, v_{h_{|V(h)|}} \mapsto v_{h_1} \right\}$. Then the constructed arc set $\mathcal{A}_{|\mathcal{J}|+2}$ must also contain the set $\mathcal{A}_0 \subseteq A$ defined in (12).

Let us denote $\mathcal{A} := \bigcup_{i=1}^{|\mathcal{J}|+2} \mathcal{A}_i$ and $\mathcal{E} := \bigcup_{i=1}^{|\mathcal{J}|+1} \mathcal{E}_i$. Thus, we have constructed the subgraph $(\mathcal{W}, \mathcal{A}, \mathcal{E})$ of the desired mixed graph $G = (V, A, E)$ in which $V = \mathcal{W}$ and $A = \mathcal{A}$.

We define the set $E \setminus \mathcal{E}$ of other edges of the mixed graph G so that it is impossible to execute simultaneously any pair of unit-time operations from the set $\mathcal{Q}^{(k)}$, $k \in \{1, \ldots, |\mathcal{M}|\}$, on a machine $M_k \in \mathcal{M}$ in any admissible schedule of the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$. For each machine $M_k \in \mathcal{M}$, the set $\mathcal{Q}^{(k)}$ of integer-time operations executed on this machine determines the set V_k of all unit-time operations executed on the machine M_k.

The cardinality of the set V_k is the total duration of all operations for processing the jobs of the set $\mathcal{J}^{(k)} =: \left\{ J_{k_1}, \ldots, J_{|\mathcal{J}^{(k)}|} \right\}$, i.e., $|V_k| = \sum_{J_i \in \mathcal{J}^{(k)}} p_{ik}$. We partition the set V_k into $|\mathcal{J}^{(k)}|$ subsets $\mathcal{V}_k^j$ of unit-time operations for processing jobs $J_{k_j} \in \mathcal{J}^{(k)}$:

$$V_k = \mathcal{V}_k^1 \bigcup \ldots \bigcup \mathcal{V}_k^{|\mathcal{J}^{(k)}|}, \ \ \mathcal{V}_k^j \neq \emptyset, \ \ \mathcal{V}_k^j \bigcap \mathcal{V}_k^l = \emptyset, \ \ k \neq l.$$

Obviously, the prohibition to execute simultaneously any pair of unit-time operations from the set V_k in an admissible schedule is defined by a complete $|\mathcal{J}^{(k)}|$-partite graph $(V_k, \emptyset, E_k)$ with the sets $\mathcal{V}_k^1, \ldots, \mathcal{V}_k^{|\mathcal{J}^{(k)}|}$ as the parts and the edge set E_k of cardinality $\prod_{J_{k_j} \in \mathcal{J}^{(k)}} |\mathcal{V}_k^j|$.

Well, we have defined the edge set $E \setminus \mathcal{E} = \bigcup_{k=1}^{|\mathcal{M}|} E_k$ and have constructed the mixed graph $G = (V, A, E)$ in which $V = \mathcal{W}$, $A = \mathcal{A}$, and $E = \mathcal{E} \bigcup E_1 \bigcup \ldots \bigcup E_{|\mathcal{M}|}$. By the construction of the mixed graph $G = (V, A, E)$, all precedence relations of the operations in the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ are defined by the subgraph $(\mathcal{W}, \mathcal{A}, \mathcal{E})$ of the mixed graph $G = (V, A, E)$; the prohibition to execute simultaneously any pair of operations from the set V_k on the same machine $M_k \in \mathcal{M}$ is defined by the subgraph $(\mathcal{W}, \bigcup_{i=1}^{|\mathcal{J}|} \mathcal{A}_i, \{\bigcup_{k=1}^{|\mathcal{M}|} E_k\} \bigcup \{\bigcup_{i=1}^{|\mathcal{J}|} \mathcal{E}_i\})$ of the mixed graph $G = (V, A, E)$. Hence, for the solvable problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on the mixed graph $G = (V, A, E)$, there exists an admissible schedule

$$\mathbf{S} = \{C(v_1), \ldots, C(v_{|\mathcal{W}|})\} \tag{15}$$

defining a coloring $c(G)$ of the mixed graph $G = (V, A, E)$ in which $c(v_i) = C(v_i)$ for all vertices $v_i \in \mathcal{W}$. Evidently, the makespan-optimal schedule defines the optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. The proof of the first part of Theorem 5 is complete.

Note that the makespan-optimal schedule for the problem $G_cMPT|p_{ij} = 1|C_{\max}$ coincides with the solution of the problem $G_cMPT|p_{ij} = 1, pmtn|C_{\max}$: by Remark 2, if all operations $Q_{ij} \in \mathcal{Q}$ have the unit durations $p_{ij} = 1$, the preemptions of a certain operation from the set $\mathcal{Q}$ will not reduce the length $C_{\max}$ of the desired makespan-optimal semi-active schedule. Therefore, the possibility of preempting unit-time operations in the problem $G_cMPT|p_{ij} = 1, pmtn|C_{\max}$ can be simply neglected when finding the makespan-optimal semi-active schedule. Hence, the problem $G_cMPT|p_{ij} = 1|C_{\max}$ can be solved as a particular case $G_cMPT|p_{ij} = 1, pmtn|C_{\max}$ of the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$; see Theorem 5. Thus, the second part of Theorem 5 directly follows from Lemma 2.

Lemma 2 [24]. *For any colorable mixed graph $G = (V, A, E)$, there exists a problem $G_cMPT|p_{ij} = 1|C_{\max}$ on the same mixed graph $G = (V, A, E)$ that is equivalent to the problem of finding an optimal coloring $c(G)$.*

The proof of Lemma 2, provided in the paper [24, pp. 78–79], contain an algorithm as follows: for a given colorable mixed graph $G = (V, A, E)$, it constructs the problem $G_cMPT|p_{ij} = 1|C_{\max}$ on the same mixed graph G that is equivalent to the problem of finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. The proof of Theorem 5 is complete.

4. EXAMPLE 2 OF THE PROBLEM $G_cMPT|[p_{ij}], pmtn|C_{\max}$

Let us illustrate the first part of Theorem 5 by Example 2 of the solvable problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ with five jobs $\mathcal{J} = \{J_1, \ldots, J_5\}$ and nine machines $\mathcal{M} = \{M_1, \ldots, M_9\}$. Table 2 presents the operations of the sets $\mathcal{Q}$, their durations, and the machines of the sets $\mathcal{M}_{\mu(i,j)} \subseteq \mathcal{M}$ required to execute an operation $Q_{ij} \in \mathcal{Q}$. Prior to defining the precedence relations between the operations of different jobs for Example 2, we construct a subgraph $G' = (V, A', E')$ of the desired mixed graph $G = (V, A, E)$.

The mixed graph $G' = (V, A', E')$ will represent the part of the input data of Example 2 shown in Table 2. The entire mixed graph $G = (V, A, E)$ will be constructed using the following fact.

Remark 3. Since the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on the mixed graph $G = (V, A, E)$ is solvable, by Theorem 4 the directed subgraph $(V, A, \emptyset)$ of the mixed graph $G = (V, A, E)$ contains no circuit with adjacent vertices of the subgraph $(V, \emptyset, E)$.

Table 2. Some input data of Example 2 of the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$

Operations	$Q_{1,1}$	$Q_{1,2}$	$Q_{2,1}$	$Q_{2,2}$	$Q_{2,3}$	$Q_{3,1}$	$Q_{3,2}$	$Q_{3,3}$	$Q_{4,1}$	$Q_{4,2}$	$Q_{4,3}$	$Q_{5,1}$	$Q_{5,2}$
Machines sets $\mathcal{M}_{\mu(i,j)}$	M_1 M_6	M_7	M_1 M_2	M_6	M_7	M_2 M_3	M_5 M_7	M_3 M_8	M_3 M_4 M_5	M_8	M_9	M_4	M_5 M_9
p_{ij}	5	1	3	1	3	2	1	2	4	2	2	4	2

The job J_1 consists of two operations $Q_{1,1}$ and $Q_{1,2}$ with the durations $p_{1,1} = 5$ and $p_{1,2} = 1$, respectively. Using the notations (4), we obtain the linearly ordered set of unit-time operations $\mathcal{W}_1 = \{v_1, \ldots, v_6\}$ and include them in the vertex set $V \supset \mathcal{W}_1$ of the mixed graph G'. All operations of the job J_1 are linearly ordered when implementing any admissible schedule. Therefore, we include the arc set $\mathcal{A}_1 = \{(v_1, v_2), (v_2, v_3), \ldots, (v_5, v_6)\}$, $\mathcal{A}_1 \subset A'$, and the edge set $\mathcal{E}_1 = \{[v_1, v_2], [v_2, v_3], \ldots, [v_5, v_6]\}$, $\mathcal{E}_1 \subset E'$, into the mixed graph G'.

The job J_2 consists of three operations $Q_{2,1}$, $Q_{2,2}$, and $Q_{2,3}$ with the durations $p_{2,1} = 3$, $p_{2,2} = 1$, and $p_{2,3} = 3$, respectively. Using the notations (5), we obtain the set of unit-time operations $\mathcal{W}_2 = \{v_7, \ldots, v_{13}\}$ and include them in the vertex set $V \supset \mathcal{W}_2$. All operations of the job J_2 are linearly ordered when implementing any admissible schedule. Therefore, we include the arc set $\mathcal{A}_2 = \{(v_7, v_8), (v_8, v_9), \ldots, (v_{12}, v_{13})\}$, $\mathcal{A}_2 \subset A'$, and the edge set $\mathcal{E}_2 = \{[v_7, v_8], [v_8, v_9], \ldots, [v_{12}, v_{13}]\}$, $\mathcal{E}_2 \subset E'$, into the mixed graph G'.

The job J_3 consists of three operations $Q_{3,1}$, $Q_{3,2}$, and $Q_{3,3}$ with the durations $p_{3,1} = 2$, $p_{3,2} = 1$, and $p_{3,3} = 2$, respectively. Using the notations (6), we obtain the set of unit-time operations $\mathcal{W}_3 = \{v_{14}, \ldots, v_{18}\}$ and include them in the vertex set $V \supset \mathcal{W}_3$. All operations of the job J_3 are linearly ordered when implementing any admissible schedule. Therefore, we include the arc set $\mathcal{A}_3 = \{(v_{14}, v_{15}), (v_{15}, v_{16}), (v_{16}, v_{17}), (v_{17}, v_{18})\}$, $\mathcal{A}_3 \subset A'$, and the edge set $\mathcal{E}_3 = \{[v_{14}, v_{15}], [v_{15}, v_{16}], [v_{16}, v_{17}], [v_{17}, v_{18}]\}$, $\mathcal{E}_3 \subset E'$, into the mixed graph G'.

The job J_4 consists of three operations $Q_{4,1}$, $Q_{4,2}$, and $Q_{4,3}$ with the durations $p_{4,1} = 4$, $p_{4,2} = 2$, and $p_{4,3} = 2$, respectively. Continuing the sequential numbering of unit-time operations, we obtain the set of unit-time operations $\mathcal{W}_4 = \{v_{19}, \ldots, v_{26}\}$ and include them in the vertex set $V \supset \mathcal{W}_4$. All operations of the job J_4 are linearly ordered when implementing any admissible schedule. Therefore, we include the arc set $\mathcal{A}_4 = \{(v_{19}, v_{20}), (v_{20}, v_{21}), \ldots, (v_{25}, v_{26})\}$, $\mathcal{A}_4 \subset A'$, and the edge set $\mathcal{E}_4 = \{[v_{19}, v_{20}], [v_{20}, v_{21}], \ldots, [v_{25}, v_{26}]\}$, $\mathcal{E}_4 \subset E'$, into the mixed graph G'.

The job J_5 consists of two operations $Q_{5,1}$ and $Q_{5,2}$ with the durations $p_{5,1} = 4$ and $p_{5,2} = 1$, respectively. Using the notations (6), we obtain the set of unit-time operations $\mathcal{W}_5 = \{v_{27}, \ldots, v_{32}\}$ and include them in the desired vertex set $V \supset \mathcal{W}_5$. All operations of the job J_5 are linearly ordered when implementing any admissible schedule. Therefore, we include the arc set $\mathcal{A}_5 = \{(v_{27}, v_{28}), (v_{28}, v_{29}), \ldots, (v_{31}, v_{32})\}$, $\mathcal{A}_5 \subset A'$, and the edge set $\mathcal{E}_5 = \{[v_{27}, v_{28}], [v_{28}, v_{29}] \ldots, [v_{31}, v_{32}]\}$, $\mathcal{E}_5 \subset E'$, into the mixed graph G'. Thus, the subgraph $G' = (V, A', E')$ of the desired mixed graph $G = (V, A, E)$ has the vertex set $V = \mathcal{W} := \bigcup_{i=1}^{5} \mathcal{W}_i$, the arc set $A' := \bigcup_{i=1}^{5} \mathcal{A}_i$, and the edge set $E' := \bigcup_{i=1}^{5} \mathcal{E}_i$.

In addition to the precedence relations between the operations of the set $\mathcal{Q}_i$ for processing the same job $J_i \in \mathcal{J} = \{J_1, \ldots, J_5\}$, we assume that Example 2 includes the following sets:

—the set $\mathcal{R}_{\rightarrow}$ of precedence relations between the operations of different sets $\mathcal{Q}_i \subseteq \mathcal{Q}$ (i.e., $v_{k_{ij}} \in \mathcal{Q}_i, v_{k_{rq}} \in \mathcal{Q}_r, r \neq i$), given by

$$\mathcal{R}_{\rightarrow} = \{v_1 \rightarrow v_7,\ v_{22} \rightarrow v_{14},\ v_{22} \rightarrow v_{30},\ v_{27} \rightarrow v_{19}\}; \tag{16}$$

—the set $\mathcal{R}_{\mapsto}$ of precedence relations $v_{k_{ij}} \mapsto v_{k_{rq}}$, given by

$$\mathcal{R}_{\mapsto} = \{v_9 \mapsto v_{19},\ v_{10} \mapsto v_{23},\ v_{23} \mapsto v_{15},\ v_{15} \mapsto v_{10},\ v_{17} \mapsto v_{25}\}. \tag{17}$$

Note that the operations of the set $V(h) = \{v_{10}, v_{15}, v_{23}\}$ must be executed simultaneously when implementing any admissible schedule since the set (17) contains the precedence relations $v_{10} \mapsto v_{23}$, $v_{23} \mapsto v_{15}$, and $v_{15} \mapsto v_{10}$. Hence, the directed subgraph $(V, A, \emptyset)$ of the desired mixed graph $G = (V, A, E)$ must contain the circuit $(v_{10}, v_{23}, v_{15}, v_{10})$.

We introduce the precedence relations $v_{k_{ij}} \rightarrow v_{k_{rq}}$ (16) by adding the arc set $\mathcal{A}_6 = \{(v_1, v_7), (v_{22}, v_{14}), (v_{22}, v_{30}), (v_{27}, v_{19})\}$, $\mathcal{A}_6 \subset A'$, and the edge set $\mathcal{E}_6 = \{[v_1, v_7], [v_{14}, v_{22}], [v_{22}, v_{30}], [v_{19}, v_{27}]\}$, $\mathcal{E}_6 \subset E'$, in the constructed mixed graph $G' = (V, A', E')$.

Also, we introduce the precedence relations $v_{k_{ij}} \mapsto v_{k_{rq}}$ (17) by adding the arc set $\mathcal{A}_7 = \{(v_9, v_{19}), (v_{10}, v_{23}), (v_{23}, v_{15}), (v_{15}, v_{10}), (v_{17}, v_{25})\}$, $\mathcal{A}_7 \subset \{A' \cup \mathcal{A}_6\}$, in the constructed mixed graph.

Thus, we have constructed the directed subgraph $(V, A, \emptyset)$ of the desired mixed graph $G = (V, A, E)$ and the subset $\mathcal{E}_6 \cup E'$ of the edge set E.

The other edges of the set $E \setminus \{\mathcal{E}_6 \cup E'\}$ of the desired mixed graph $G = (V, A, E)$ will be constructed sequentially for all sets $\mathcal{Q}^{(k)} = \bigcup_{J_i \in \mathcal{J}^{(k)}} \mathcal{Q}_i^{(k)}$ of operations executed on machines $M_k \in \{M_1, \dots, M_9\}$.

For the machine M_1, the set $\mathcal{Q}^{(1)} = \{Q_{1,1}, Q_{2,1}\}$ of integer-time operations defines the set of eight unit-time operations $\{v_1, \dots, v_5, v_7, v_8, v_9\} =: V_1$, which is partitioned into five unit-time operations $\{v_1, \dots, v_5\}$ of the job J_1 and three unit-time operations $\{v_7, v_8, v_9\}$ of the job J_2. The prohibition to execute simultaneously any pair of operations from the set V_1 is defined by a complete bipartite graph $(V_1, \emptyset, E_1')$ in which $V_1 = \{v_1, \dots, v_5;\ v_7, v_8, v_9\}$. Like in subsection 2.3, the vertices of different parts of the k-partite graph are separated by the semicolon. Due to Remark 3, the edges $[v_1, v_7], [v_1, v_8]$, and $[v_1, v_9]$ can be eliminated from the complete bipartite graph $(V_1, \emptyset, E_1')$ since the order of executing the operations v_1 and v_i, $i \in \{7, 8, 9\}$, is determined by a path in the directed graph $(V, A, \emptyset)$ and a chain in the graph $(V, \emptyset, \{\mathcal{E}_6 \cup E'\})$ between the vertices v_1 and v_i. Therefore, we construct the bipartite graph $(V_1, \emptyset, E_1)$ with $E_1 = \{[v_2, v_7], \dots, [v_5, v_7], [v_2, v_8], \dots, [v_5, v_8], [v_2, v_9], \dots, [v_5, v_9]\}$ instead of the complete bipartite graph $(V_1, \emptyset, E_1')$ with $|E_1'| = |\mathcal{Q}_1^{(1)}| \cdot |\mathcal{Q}_2^{(1)}| = 5 \cdot 3 = 15$.

For the machine M_2, the set $\mathcal{Q}^{(2)}$ of integer-time operations defines the set V_2 of five unit-time operations, which is partitioned into three unit-time operations $\{v_7, v_8, v_9\}$ of the job J_2 and two unit-time operations $\{v_{14}, v_{15}\}$ of the job J_3. The prohibition to execute simultaneously any pair of operations from the set V_2 is defined by a complete bipartite graph $(V_2, \emptyset, E_2')$ in which $V_2 = \{v_7, v_8, v_9;\ v_{14}, v_{15}\}$ and $E_2 = \{[v_7, v_{14}], [v_7, v_{15}], [v_8, v_{14}], [v_8, v_{15}], [v_9, v_{14}], [v_9, v_{15}]\}$.

For the machine M_3, the set $\mathcal{Q}^{(3)}$ of integer-time operations defines the set V_3 of eight unit-time operations, which is partitioned into four unit-time operations $\{v_{14}, v_{15}, v_{17}, v_{18}\}$ of the job J_3 and four unit-time operations $\{v_{19}, \dots, v_{22}\}$ of the job J_4. The set $\mathcal{R}_{\rightarrow}$ contains the precedence relation $v_{22} \rightarrow v_{14}$, and the operations of the sets $\mathcal{Q}_3$ and $\mathcal{Q}_4$ for processing the jobs J_3 and J_4 are ordered. Hence, it is unnecessary to add edges in the desired mixed graph for prohibiting the simultaneous execution of a pair of operations from the set $\mathcal{Q}^{(2)}$. In this case, let $E_3 = \emptyset$.

For the machine M_4, the set $\mathcal{Q}^{(4)}$ of integer-time operations defines the set V_4 of eight unit-time operations, which is partitioned into four unit-time operations $\{v_{19}, \dots, v_{22}\}$ of the job J_4 and four unit-time operations $\{v_{27}, \dots, v_{30}\}$ of the job J_5. The prohibition to execute simultaneously any pair of operations from the set V_4 is defined by a complete bipartite graph $(V_4, \emptyset, E_4')$ in which $V_4 = \{v_{19}, \dots, v_{22};\ v_{27}, \dots, v_{30}\}$. Due to Remark 3, the edges of the set $\{[v_{19}, v_{28}], [v_{19}, v_{29}], [v_{19}, v_{30}]\}$ can be eliminated from the complete bipartite graph $(V_4, \emptyset, E_4')$ since the order of executing the operations v_{19} and v_i, $i \in \{28, 29, 30\}$, is determined by a path in the directed graph $(V, A, \emptyset)$ and a chain in the graph $(V, \emptyset, \{\mathcal{E}_6 \cup E'\})$ between the vertices v_{19} and v_i. Since the set $\mathcal{R}_{\rightarrow}$ contains the precedence relation $v_{22} \rightarrow v_{30}$, we eliminate the edges of the set $\{[v_{19}, v_{30}], [v_{20}, v_{30}], [v_{21}, v_{30}]\}$ and construct the bipartite graph $(V_4, \emptyset, E_4)$ with $E_4 = \{[v_{20}, v_{28}], [v_{20}, v_{29}], [v_{21}, v_{28}], [v_{21}, v_{29}]\}$ instead of the complete bipartite graph $(V_4, \emptyset, E_4')$ with $|E_4'| = |\mathcal{Q}_4^{(4)}| \cdot |\mathcal{Q}_5^{(4)}| = 4 \cdot 4 = 16$.

For the machine M_5, the set $\mathcal{Q}^{(5)}$ of integer-time operations defines the set V_5 of seven unit-time operations, which is partitioned into one unit-time operation v_{16} of the job J_3, four unit-time operations $\{v_{19}, \dots, v_{22}\}$ of the job J_4, and two operations $\{v_{31}, v_{32}\}$ of the job J_5. The prohibition to execute simultaneously any pair of operations from the set V_5 is defined by a complete tripartite graph $(V_5, \emptyset, E_5')$ in which $V_5 = \{v_{16}; v_{19}, \dots, v_{22};\ v_{31}, v_{32}\}$. Due to Remark 3, we eliminate the edges of the set $\{[v_{16}, v_{19}], \ldots, [v_{16}, v_{22}]\}$ from the complete tripartite graph $(V_5, \emptyset, E'_5)$ since the set $\mathcal{R}_\rightarrow$ contains the precedence relation $v_{22} \rightarrow v_{14}$. Also, the set $\mathcal{R}_\rightarrow$ contains the precedence relation $v_{22} \rightarrow v_{30}$, and we eliminate the edges of the set $\{[v_{19}, v_{31}], [v_{19}, v_{32}], [v_{20}, v_{31}], [v_{20}, v_{32}], [v_{21}, v_{31}], [v_{21}, v_{32}], [v_{22}, v_{31}], [v_{22}, v_{32}]\}$ as well. Therefore, we construct the tripartite graph $(V_5, \emptyset, E_5)$ with $E_5 = \{[v_{16}, v_{31}], [v_{16}, v_{32}]\}$ instead of the complete tripartite graph $(V_4, \emptyset, E'_4)$ with $|E'_5| = |\mathcal{Q}_3^{(5)}| \cdot |\mathcal{Q}_4^{(5)}| \cdot |\mathcal{Q}_5^{(5)}| = 1 \cdot 4 \cdot 2 = 8$.

For the machine M_6, the set $\mathcal{Q}^{(6)}$ of integer-time operations defines the set V_6 of six unit-time operations, which is partitioned into five unit-time operations $\{v_1, \ldots, v_5\}$ of the job J_1 and one unit-time operation v_{10} of the job J_2. The prohibition to execute simultaneously any pair of operations from the set V_6 is defined by a complete bipartite graph $(V_6, \emptyset, E_6)$ in which $V_6 = \{v_1, \ldots, v_5;\ v_{10}\}$ and $E_6 = \{[v_1, v_{10}], \ldots, [v_5, v_{10}]\}$.

For the machine M_7, the set $\mathcal{Q}^{(7)}$ of integer-time operations defines the set V_7 of five unit-time operations, which is partitioned into one unit-time operation v_6 of the job J_1, three unit-time operations $\{v_{11}, v_{12}, v_{13}\}$ of the job J_2, and one unit-time operation v_{16} of the job J_3. The prohibition to execute simultaneously any pair of operations from the set V_7 is defined by a complete tripartite graph $(V_7, \emptyset, E_7)$ in which $V_7 = \{v_6;\ v_{11}, v_{12}, v_{13};\ v_{16}\}$ and $E_6 = \{[v_6, v_{11}], [v_6, v_{12}], [v_6, v_{13}]; [v_6, v_{16}];\ [v_{11}, v_{16}], [v_{12}, v_{16}], [v_{13}, v_{16}]\}$.

For the machine M_8, the set $\mathcal{Q}^{(8)}$ of integer-time operations defines the set V_8 of four unit-time operations, which is partitioned into two unit-time operations $\{v_{17}, v_{18}\}$ of the job J_3 and two unit-time operations $\{v_{23}, v_{24}\}$ of the job J_4. The prohibition to execute simultaneously any pair of operations from the set V_8 is defined by a complete bipartite graph $(V_8, \emptyset, E_8)$ in which $V_8 = \{v_{17}, v_{18};\ v_{23}, v_{24}\}$ and $E_8 = \{[v_{17}, v_{23}], [v_{17}, v_{24}], [v_{18}, v_{23}], [v_{18}, v_{24}]\}$.

For the machine M_9, the set $\mathcal{Q}^{(9)}$ of integer-time operations defines the set V_9 of four unit-time operations, which is partitioned into two unit-time operations $\{v_{25}, v_{26}\}$ of the job J_4 and two unit-time operations $\{v_{31}, v_{32}\}$ of the job J_5. The prohibition to execute simultaneously any pair of operations from the set V_9 is defined by a complete bipartite graph $(V_9, \emptyset, E_9)$ in which $V_9 = \{v_{25}, v_{26};\ v_{31}, v_{32}\}$ and $E_9 = \{[v_{25}, v_{31}], [v_{25}, v_{32}], [v_{26}, v_{31}], [v_{26}, v_{32}]\}$.

Thus, we have constructed the subgraph $(V, \emptyset, E \setminus \{\mathcal{E}_6 \cup E'\}) = \bigcup_{k=1}^{9} (V_k, \emptyset, E_k)$ of the mixed graph (V, A, E) in which each graph $(V_k, \emptyset, E_k)$ is a $|\mathcal{J}^{(k)}|$-partite graph. Figure 2 shows the constructed mixed graph $G = (V, A, E)$ defining a input data of Example 2 of the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ with five jobs $\mathcal{J} = \{J_1, \ldots, J_5\}$ and nine machines $\mathcal{M} = \{M_1, \ldots, M_9\}$.

According to Theorem 5, the individual problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on the mixed graph $G = (V, A, E)$ (Example 2) has been reduced to the problem of finding an optimal coloring $c(G)$ of the same mixed graph $G = (V, A, E)$. An admissible schedule $\mathbf{S}$ for the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ with possible preemptions of operations $\mathcal{Q}$ is defined by the set (15) of the completion times of unit-time operations from the set $\mathcal{W}$. A semi-active schedule is defined by a coloring $c(G)$ of a mixed graph G in which $c(v_i) = C(v_i)$ for all vertices $v_i \in \mathcal{W}$.

A makespan-optimal semi-active schedule for Example 2 is defined by the following optimal coloring $c(G)$ of the mixed graph G in Fig. 2:

$$
\begin{gathered}
c(v_1) = 1, \quad c(v_2) = 5, \quad c(v_3) = 6, \quad c(v_4) = 7, \quad c(v_5) = 8, \quad c(v_6) = 9, \quad c(v_7) = 2, \\
c(v_8) = 3, \quad c(v_9) = 4, \quad c(v_{10}) = 9, \quad c(v_{11}) = 11, \quad c(v_{12}) = 12, \quad c(v_{13}) = 13, \quad c(v_{14}) = 8, \\
c(v_{15}) = 9, \quad c(v_{16}) = 10, \quad c(v_{17}) = 11, \quad c(v_{18}) = 12, \quad c(v_{19}) = 4, \quad c(v_{20}) = 5, \\
c(v_{21}) = 6, \quad c(v_{22}) = 7, \quad c(v_{23}) = 9, \quad c(v_{24}) = 10, \quad c(v_{25}) = 12, \quad c(v_{26}) = 13, \quad c(v_{27}) = 1, \\
c(v_{28}) = 2, \quad c(v_{29}) = 3, \quad c(v_{30}) = 8, \quad c(v_{31}) = 9, \quad c(v_{32}) = 11.
\end{gathered}
$$

The coloring $c(G)$ is optimal due to the inequality $\chi(G) \geqslant 13$, holding since the directed subgraph $(V, A, \emptyset)$ of the mixed graph G contains the path $(v_1, v_7, v_8, v_9, v_{19}, v_{20}, v_{21}, v_{22}, v_{14}, v_{15}, v_{16}, v_{17}, v_{18},$

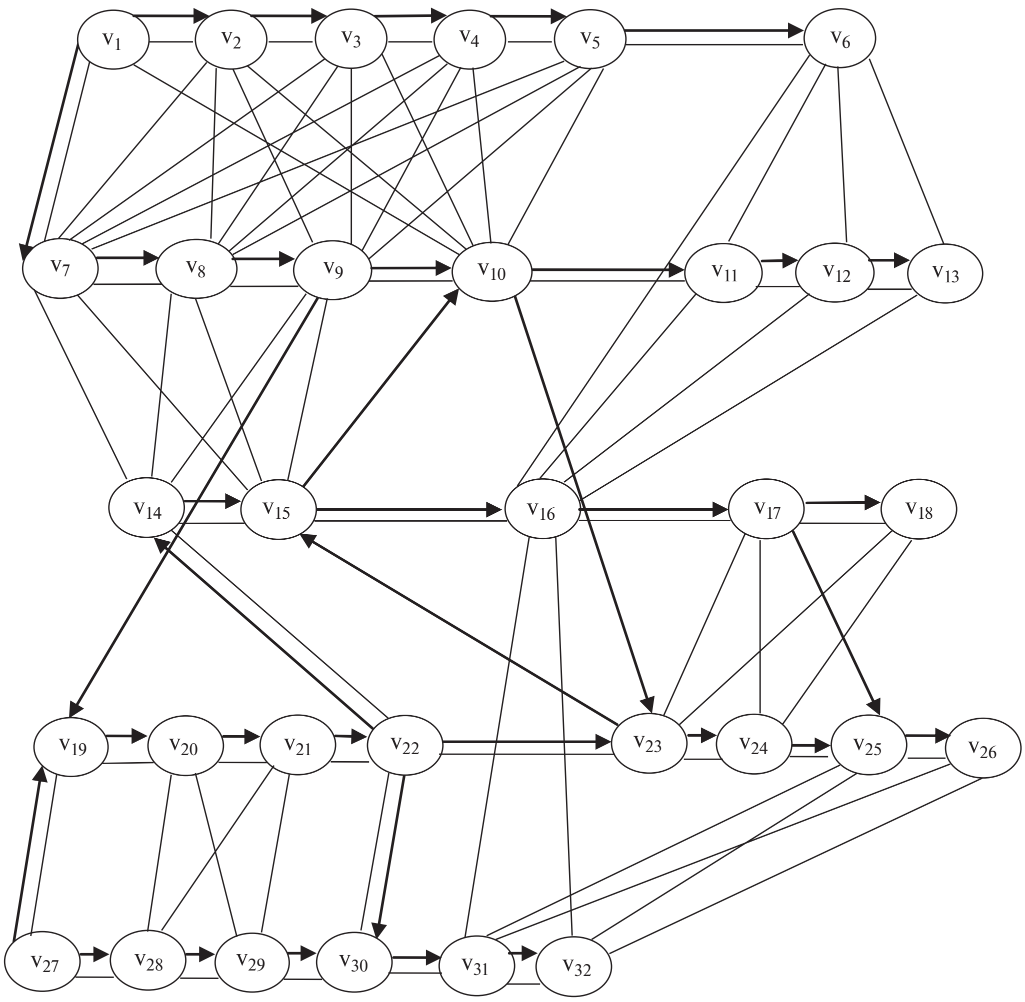

Fig. 2. Mixed graph $G = (V, A, E)$ defining the input data of Example 2 of problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$.

$v_{25}, v_{26})$ with a weight of 13. Here, the weight of a path in a mixed graph $G = (V, A, E)$ is the sum of the weights $w(v_i, v_j)$ of all arcs (v_i, v_j) in this path; the weight of an arc $(v_i, v_j) \in A$ is 1 if $[v_i, v_j] \in E$ and $w(v_i, v_j) = 0$ otherwise.

5. SEMI-ACTIVE SCHEDULES FOR THE PROBLEM $G_cMPT|[p_{ij}], pmtn|C_{\max}$ AND THE CORRESPONDING MINIMAL COLORINGS OF MIXED GRAPH VERTICES

Let E^* denote the subset of all edges $[v_i, v_j]$ of a set E for which the vertices v_i and v_j are not adjacent in a directed graph $(V, A, \emptyset)$. By Definition 1, any coloring $c(G)$ of a mixed graph $G = (V, A, E)$ defines different colors $c(v_i) \neq c(v_j)$ for each edge $[v_i, v_j] \in E^*$. If $c(v_i) < c(v_j)$, we add the arc (v_i, v_j) in the mixed graph G; if $c(v_i) > c(v_j)$, we add the symmetric arc (v_j, v_i) in this graph. After adding these arcs for all edges $[v_i, v_j] \in E^*$, the mixed graph $G = (V, A, E)$ becomes a mixed graph $G(c) = (V, A \cup A(c), E)$, $|E^*| = |A(c)|$ where

$$[v_p, v_q] \in E \Rightarrow (v_p, v_q) \in A \cup A(c) \tag{18}$$

for each edge $[v_p, v_q] \in E$.

According to (18), any coloring $c(G)$ of a mixed graph $G = (V, A, E)$ defines an order of colors $c(v_i)$ for all vertices $v_i \in V$. Hence, it is possible to define the following set of minimal colorings $c(G)$ of a mixed graph G.

Definition 4. A coloring $c(G)$ of a mixed graph $G = (V, A, E)$ is said to be minimal if none of the colors $c(v_i)$, $v_i \in V$, can be reduced without violating the order of colors $c(v_i)$ defined by this coloring $c(G)$ for all vertices $v_i \in V$ and (or) some vertex $v_j \in V \setminus \{v_i\}$ would be required to assign a color greater than the color $c(v_j)$.

An optimal coloring of a mixed graph $G = (V, A, E)$ can be found in the set of minimal colorings: obviously, there exists an optimal coloring of a given colorable mixed graph G that is minimal. Note that the optimal coloring $c(G)$ of the mixed graph G in Example 2 is minimal.

A minimal coloring $c(G)$ defines the optimal scheduling (15) *for the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on a mixed graph $G = (V, A, E)$; moreover, this schedule is semi-active.* By Lemma 2 and Definitions 2 and 4, any semi-active schedule $\mathbf{S}$ existing for the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on a mixed graph $G = (V, A, E)$ uniquely defines a minimal coloring of its vertices, and vice versa. Thus, we arrive at the following result.

Theorem 6. *There exists a bijection between the set $\mathbf{C}(G)$ of all minimal colorings $c(G)$ of a colorable mixed graph $G = (V, A, E)$ and the set $\mathbf{S}(G)$ of all semi-active schedules existing for the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on the same mixed graph $G = (V, A, E)$.*

According to Theorems 5 and 6, it is possible to reduce the dimension of the set of admissible schedules compared when solving the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ (hence, reduce the dimension of the set of compared colorings $c(G)$ when finding an optimal one).

It follows from the second part of Theorem 5 that the problem of finding an optimal coloring $c(G)$ of any colorable mixed graph $G = (V, A, E)$ has the same asymptotic complexity as the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on the mixed graph $G = (V, A, E)$ and, moreover, $|\mathbf{C}(G)| = |\mathbf{S}(G)|$.

According to the first part of Theorem 5, the reduction of the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ to finding an optimal coloring $c(G)$ of a mixed graph G is pseudopolynomial. Such a reduction is reasonable in real production scheduling systems if the durations c_{ij} of operations $Q_{ij} \in \mathcal{Q}$ are not high. For the problem with long integer-time operations of the set $\mathcal{Q}$, we can calculate the greatest common divisor D of their integer durations.

If the value D exceeds 1, the original problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ should be replaced by its analog with the modified durations $\frac{c_{ij}}{D}$ for all operations $Q_{ij} \in \mathcal{Q}$.

The dimension of the mixed graph $G = (V, A, E)$ induced by the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ can also be reduced through a suitable decomposition of the original problem into smaller-dimension subproblems based on planned preemptions of the production process, e.g., when a lunch break begins or a work shift ends. During such interruptions, the updated and supplemented data can be included in the input data of the next subproblem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ to obtain a more effective schedule of the production process.

Note that by a common assumption of scheduling theory, an operation is preempted with any cost and instantaneously, like its subsequent resumption. Such ideal preemptions are not typical of many real production scheduling systems. In practice, some time is required to interrupt a production operation; also, it is necessary to consider the machine changeover time after a preemption and the machine set-up time for resuming the operation preempted. For this purpose, special operations (machine changeover and set-up) should be included into the input data of the problems $G_cMPT|[p_{ij}], pmtn|C_{\max}$. The durations of changeover and set-up operations of machines $M_{\mu(ij)} \in \mathcal{M}$ should be less than the durations of operations $Q_{ij} \in \mathcal{Q}$ reasonable to be preempted.

6. DISCUSSION OF THE RESULTS AND POTENTIAL USE

The assertions in Sections 1–5 can be used to construct network models in the form of mixed graphs $G = (V, A, E)$ for numerous scheduling problems $\alpha|[p_{ij}], pmtn|C_{\max}$ with possible operation preemptions and scheduling problems $\alpha|p_{ij} = 1|C_{\max}$ without them. The constructed network models may serve for developing algorithms and computer programs to minimize the length $C_{\max}$ of semi-active schedules. Such algorithms would involve finding the optimal colorings $c(G)$ or strict colorings $c_<(G)$ of mixed graphs $G = (V, A, E)$ defining the input data of the problem $\alpha|p_{ij} = 1|C_{\max}$ or the solvable problem $\alpha|[p_{ij}], pmtn|C_{\max}$.

As established in the paper, the generalizations of classical scheduling problems (see above) are related to the problems of finding optimal (strict) colorings of mixed graph vertices. This relationship allows solving any solvable problems $\alpha|[p_{ij}], pmtn|C_{\max}$ and any problems $\alpha|[p_{ij}]|C_{\max}$, as well as their numerous particular cases, using only terms of graph theory without using special terms of scheduling theory associated with real production scheduling systems. As it turned out, graph theory terms are sufficient for investigating any problem $\alpha|[p_{ij}], pmtn|C_{\max}$ or $\alpha|p_{ij} = 1|C_{\max}$ and developing solution algorithms for scheduling problems by finding optimal (strict) colorings of mixed graphs $G = (V, A, E)$ defining the problem input data.

To illustrate the advantages of using the terminology of graph theory when solving problems $\alpha|[p_{ij}], pmtn|C_{\max}$ and $\alpha|p_{ij} = 1|C_{\max}$, we list the graph theory terms employed in this paper: *(general) work shop, (multistage) processing or service system, job-shop, general shop, (service or dedicated) machine, equipment, processor, (semi-active, admissible, optimal) schedule, optimality criterion for schedules, implementing an admissible schedule, makespan-optimal schedule, (first, second, last) job, processing jobs, the time of job readiness to processing, planning horizon, scheduling length, (multiprocessor, preempted) operation, (integer) preemption, job processing route, first (last) job operation, unit- or integer-time operation, assignment of operation to machine, the prohibition of executing operations simultaneously, joint execution of (unit-time) operations, operation start, operation completion, start (completion) time of operation, operation preemption, schedule time, precedence relation of operations (completion–start, start–start), job realization time, production conditions (efficiency), and (integer) time.* Fewer graph theory terms have been adopted to describe and prove the same results of the paper in the terminology of colorings of mixed graph vertices, namely: *(incident, adjacent) vertex, arc, (incident) edge, (finite, directed, mixed, k-partite, complete, colorable) graph, subgraph, (strict, minimal, optimal) coloring, vertex color, chromatic number, path, path length (weight), chain, and circuit.*

The following assertions, proved above, are important both for scheduling theory and graph theory:

1. The problems $\alpha|p_{ij} = 1|C_{\max}$ on a mixed graph $G = (V, A, E)$ are equivalent to the problems of finding an optimal $c(G)$ or strict $c_<(G)$ coloring of the same mixed graph $G = (V, A, E)$.

2. The problem of finding an optimal coloring $c(G)$ of any colorable mixed graph $G = (V, A, E)$ is polynomially reduced to the problem $\alpha|[p_{ij}], pmtn|C_{\max}$ on the same mixed graph $G = (V, A, E)$.

3. Any solvable problem $\alpha|[p_{ij}], pmtn|C_{\max}$ on a mixed graph $G = (V, A, E)$ is pseudopolynomially reduced to the problem of finding an optimal coloring $c(G)$ on the same mixed graph $G = (V, A, E)$.

The results of this paper justify the graph-theoretic approach to solving the scheduling problems $\alpha|p_{ij} = 1|C_{\max}$ and $\alpha|[p_{ij}], pmtn|C_{\max}$ by reducing them to the corresponding problems of finding optimal $c(G)$ or optimal strict $c_<(G)$ colorings of mixed graphs $G = (V, A, E)$ defining the conditions and constraints of scheduling problems under considerartion. The conditions and constraints (input data) of an individual problem $\alpha|p_{ij} = 1|C_{\max}$ or $G_cMPT|[p_{ij}], pmtn|C_{\max}$ are defined by the corresponding mixed graph $G = (V, A, E)$. For the problem $\alpha|p_{ij} = 1|C_{\max}$, such a mixed graph $G = (V, A, E)$ is uniquely defined. For the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$, it is defined up to the redundant edges from the set E, which can be found and removed by the rank-based partition of the vertices of the directed subgraph $G = (V, A, \emptyset)$ of the mixed graph $G = (V, A, E)$; for details, see Example 2 in Section 4. The results of this paper can be used to investigate the colorings of special classes of mixed graphs induced by the scheduling problems $\alpha|p_{ij} = 1|C_{\max}$ and

$\alpha|[p_{ij}], pmtn|C_{\max}$, which may involve graph theory experts in solving the problems $\alpha|p_{ij} = 1|C_{\max}$ and $G_cMPT|[p_{ij}], pmtn|C_{\max}$ and their numerous particular cases.

As is easily verified, different problems $\alpha|p_{ij} = 1|C_{\max}$ or $\alpha|[p_{ij}], pmtn|C_{\max}$ can be defined by the same mixed graph $G = (V, A, E)$. Therefore, the results obtained for a coloring $c(G)$ or strict coloring $c_<(G)$ of a particular mixed graph $G = (V, A, E)$ are generally applicable to the entire set of scheduling problems $\alpha|p_{ij} = 1|C_{\max}$ or $\alpha|[p_{ij}], pmtn|C_{\max}$, respectively. Hence, coloring properties and algorithms for finding optimal colorings of mixed graphs $G = (V, A, E)$ of special form can be used to solve all problems $\alpha|p_{ij} = 1|C_{\max}$ or $\alpha|[p_{ij}], pmtn|C_{\max}$, respectively, with the input data defined by the same mixed graphs.

7. CONCLUSIONS

We have studied the relationship between optimal vertex coloring problems of a mixed graph $G = (V, A, E)$ and the problems $G_cMPT|[p_{ij}], pmtn|C_{\max}$ of constructing a makespan-optimal schedule for executing a partially ordered set of integer-time operations with possible preemptions with two types of precedence relations on the operation set (completion–start and start–start) and the simultaneous execution of a subset of unit-time operations. According to Theorem 5, the problem of finding an optimal coloring of the vertices of a mixed graph $G = (V, A, E)$ is polynomially reduced to the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on the same mixed graph G. The constructive proof of this theorem indicates the following: for many assertions proved for optimal colorings of mixed graph vertices (e.g., see [1–9, 24]), there are similar assertions for the problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ and its particular cases. Also, it has been established that any solvable problem $G_cMPT|[p_{ij}], pmtn|C_{\max}$ on a mixed graph $G = (V, A, E)$ is pseudopolynomially reduced to the problem of finding an optimal coloring $c(G)$ of vertices of the same mixed graph G. Therefore, assertions for the problems of finding optimal colorings of mixed graph vertices can be derived directly from the assertions proved for the problems $G_cMPT|[p_{ij}], pmtn|C_{\max}$ and its numerous particular cases [2, 5, 12, 13, 15–24].

8. ACKNOWLEDGMENTS

The author is grateful to the reviewers for careful reading of the manuscript and helpful remarks.

FUNDING

This work was supported by the Belarusian Republican Foundation for Fundamental Research, projects nos. F21-010 and F23RNF-017.

REFERENCES

1. Sotskov, Y.N. and Tanaev, V.S., A Chromatic Polynomial of a Mixed Graph, *Proc. Nat. Acad. Sci. Belarus. Phys. Math. Ser.*, 1976, no. 6, pp. 20–23.

2. Hansen, P., Kuplinsky, J., and de Werra, D., Mixed Graph Colorings, *Math. Methods Oper. Res.*, 1997, vol. 45, pp. 145–160.

3. Karp, R.M., Reducibility among Combinatorial Problems, in *Complexity of Computer Computations*, Miller, R.E. and Thatcher, J.W., Eds., New York: Plenum Press, 1972, pp. 85–103.

4. Sotskov, Y.N., Dolgui, A., and Werner, F., Mixed Graph Coloring for Unit-Time Job-Shop Scheduling, *Int. J. of Math. Algorithms*, 2001, vol. 2, pp. 289–323.

5. Sotskov, Y.N., Tanaev, V.S., and Werner, F., Scheduling Problems and Mixed Graph Colorings, *Optimization*, 2002, vol. 51, no. 3, pp. 597–624.

6. Al-Anzi, F.S., Sotskov, Y.N., Allahverdi, A., and Andreev, G.V., Using Mixed Graph Coloring to Minimize Total Completion Time in Job Shop Scheduling, *Appl. Math. Comput.*, 2006, vol. 182, pp. 1137–1148.

7. Kouider, A., Ait Haddadene, H., Ourari, S., and Oulamara, A., Mixed Graph Coloring for Unit-Time Scheduling, *Int. J. Product. Res.*, 2017, vol. 55, no. 6, pp. 1720–1729.

8. Kouider, A., Ait Haddadene, H., and Oulamara, A., On Minimization of Memory Usage in Branch-and-Bound Algorithm for the Mixed Graph Coloring: Application to the Unit-Time Job Shop Scheduling, *Comput. Oper. Res.*, 2019, vol. 4967, pp. 1001–1008.

9. Sotskov, Y.N., Mixed Graph Colorings: A Historical Review, *Mathematics*, 2020, vol. 8, pp. 1–24.

10. Harary, F., *Graph Theory*, Reading: Addison-Wesley, 1969.

11. Thulasiraman, K. and Swamy, M.N.S., *Graphs: Theory and Algorithms*, Toronto: John Wiley & Sons, 1992.

12. Tanaev, V.S., Sotskov, Y.N., and Strusevich, V.A., *Scheduling Theory: Multi-Stage Systems*, Dordrecht: Kluwer Academic Publishers, 1994.

13. Brucker, P., *Scheduling Algorithms*, Berlin: Springer, 1995.

14. Graham, R.E., Lawler, E.L., Lenstra, J.K., and Rinnooy Kan, A.H.G., Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey, *Ann. Discret. Math.*, 1979, vol. 5, pp. 287–326.

15. Hoogeveen, J.A., van de Velde, S.L., and Veltman, B., Complexity of Scheduling Multiprocessor Tasks with Prespecified Processor Allocations, *Discret. Appl. Math.*, 1994, vol. 55, pp. 259–272.

16. Brucker, P. and Kramer, A., Shop Scheduling Problems with Multiprocessor Tasks on Dedicated Processors, *Ann. Oper. Res.*, 1995, vol. 57, pp. 13–27.

17. Chou, F.D., Particle Swarm Optimization with Cocktail Decoding Method for Hybrid Flow Shop Scheduling Problems with Multiprocessor Tasks, *Int. J. Prod. Econom.*, 2013, vol. 141, pp. 137–145.

18. Kurdi, M., Ant Colony System with a Novel Non-Daemon Actions Procedure for Multiprocessor Task Scheduling in Multistage Hybrid Flow Shop, *Swarm Evol. Comput.*, 2019, vol. 44, pp. 987–1002.

19. Drozdowski, M., Scheduling Multiprocessor—an Overview, *Eur. J. Oper. Res.*, 1996, vol. 94, pp. 215–230.

20. Baptiste, P., A Note on Scheduling Multiprocessor Tasks with Identical Processing Times, *Comput. Oper. Res.*, 2003, vol. 30, pp. 2071–2078.

21. Zinder, Y., Dob, V.H., and Oguz, C., Computational Complexity of Some Scheduling Problems with Multiprocessor Tasks, *Discret. Optimization*, 2005, vol. 2, pp. 391–408.

22. Kis, T., Scheduling Multiprocessor UET Tasks of Two Sizes, *Theor. Comput. Sci.*, 2009, vol. 410, pp. 4864–4873.

23. Giaro, K., Kubale, M., and Obszarski, P., A Graph Coloring Approach to Scheduling of Multiprocessor Tasks on Dedicated Machines with Availability Constraints, *Discret. Appl. Math.*, 2009, vol. 157, pp. 3625–3630.

24. Sotskov, Y.N., Mixed Graph Coloring as Scheduling Multi-Processor Tasks with Equal Processing Times, *J. Belarusian State Univ. Math. Inform.*, 2021, vol. 2, pp. 67–81.

This paper was recommended for publication by P.Yu. Chebotarev, a member of the Editorial Board