


Article

Scheduling Multiprocessor Tasks with Equal Processing Times as a Mixed Graph Coloring Problem

Yuri N. Sotskov ^{1,*}  and Evangelina I. Mihova ²

¹ United Institute of Informatics Problems of the National Academy of Sciences of Belarus, 6 Surganov Street, 220012 Minsk, Belarus

² Oskar-Maria-Graf-Gymnasium, Keltenweg 5, 85375 Neufahrn bei Freising, Germany; evangelinamihova@gmail.com

* Correspondence: sotskov48@mail.ru; Tel.: +375-17-249-61-20

Abstract: This article extends the scheduling problem with dedicated processors, unit-time tasks, and minimizing maximal lateness L_{\max} for integer due dates to the scheduling problem, where along with precedence constraints given on the set $V = \{v_1, v_2, \dots, v_n\}$ of the multiprocessor tasks, a subset of tasks must be processed simultaneously. Contrary to a classical shop-scheduling problem, several processors must fulfill a multiprocessor task. Furthermore, two types of the precedence constraints may be given on the task set V . We prove that the extended scheduling problem with integer release times $r_i \geq 0$ of the jobs V to minimize schedule length C_{\max} may be solved as an optimal mixed graph coloring problem that consists of the assignment of a minimal number of colors (positive integers) $\{1, 2, \dots, t\}$ to the vertices $\{v_1, v_2, \dots, v_n\} = V$ of the mixed graph $G = (V, A, E)$ such that, if two vertices v_p and v_q are joined by the edge $[v_p, v_q] \in E$, their colors have to be different. Further, if two vertices v_i and v_j are joined by the arc $(v_i, v_j) \in A$, the color of vertex v_i has to be no greater than the color of vertex v_j . We prove two theorems, which imply that most analytical results proved so far for optimal colorings of the mixed graphs $G = (V, A, E)$, have analogous results, which are valid for the extended scheduling problems to minimize the schedule length or maximal lateness, and vice versa.

Keywords: unit-time schedule; multiprocessor task; makespan; lateness; mixed graph coloring



Citation: Sotskov, Y.N.; Mihova, E.I. Scheduling Multiprocessor Tasks with Equal Processing Times as a Mixed Graph Coloring Problem. *Algorithms* **2021**, *14*, 246. <https://doi.org/10.3390/a14080246>

Academic Editor: Jesper Jansson

Received: 16 July 2021

Accepted: 13 August 2021

Published: 19 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The world of manufacturing includes the following two main groups: custom manufacturing and mass production. In custom manufacturing, the quantity of the produced items may be small. Such production is highly flexible to enable a customization to the specific needs of the clients. Consequently, humans mostly perform custom manufacturing. Conversely, high automation characterizes scenarios of mass production, which consists of a fixed order of operations, uniform (equal) operation durations, a lack of the process flexibility, and an absorption of costs derived from the defective production units [1].

In our article, we focus on the above mass production, which presupposes the scheduling problems with unit processing times of the jobs to minimize either makespan C_{\max} (the schedule length) or maximal lateness L_{\max} . Scheduling models with equal (unit) processing times of the jobs are an approximation for coping with the mass-industrial productions and manufactures of similar items, particularly for job-shop manufacturing that allows a manager to personalize an individual item [2].

Unit-time shop-scheduling problems to minimize the makespan are equivalent to an optimal graph coloring problem that consists of assigning a minimal number of colors to the vertices of a graph such that no two adjacent vertices have the same color.

If a unit-time, shop-scheduling problem requires both precedence and incompatibility constraints; a mixed graph coloring [3] allows modeling such a shop-scheduling problem. Since publishing article [3], many studies on the unit-time shop-scheduling problems with minimizing the makespan are based on mixed graph colorings.

Let $G = (V, A, E)$ denote a mixed graph with the set $V = \{v_1, v_2, \dots, v_n\} \neq \emptyset$ of the vertices placed at the first position in parenthesis, the set A of the arcs at the second position, and the set E of the edges at the third position. The arc $(v_i, v_j) \in A$ is an ordered pair of vertices v_i and v_j , while the edge $[v_p, v_q] \in E$ is an unordered pair of vertices v_p and v_q . If the equality $A = \emptyset$ holds, one has a graph $G = (V, \emptyset, E)$. If the equality $E = \emptyset$ holds, one has a digraph $G = (V, A, \emptyset)$. A mixed graph $G = (V, A, E)$ under consideration contains no multiple arcs, no multiple edges, and no loops.

Article [3] introduces an optimal mixed graph coloring problem as follows.

Definition 1 ([3]). An integer-valued function $c : V \rightarrow \{1, 2, \dots, t\}$ is called a coloring $c(G)$ of the mixed graph $G = (V, A, E)$ if the non-strict inequality $c(v_i) \leq c(v_j)$ holds for each arc $(v_i, v_j) \in A$ and the relation $c(v_p) \neq c(v_q)$ holds for each edge $[v_p, v_q] \in E$. A mixed graph coloring $c(G)$ is optimal if it uses a minimal number $\chi(G)$ of different colors $c(v_i) \in \{1, 2, \dots, t\}$; such a minimal number $\chi(G)$ of the used colors being called a chromatic number of the mixed graph $G = (V, A, E)$.

If the equality $A = \emptyset$ holds, the coloring $c(G)$ is a usual coloring of the vertices of the graph $G = (V, \emptyset, E)$. Contrary to coloring of the vertices of the graph $G = (V, \emptyset, E)$ that exists for any graph $G = (V, \emptyset, E)$, a mixed graph $G = (V, A, E)$ may be un-colorable; see the following criterion proved in [3].

Theorem 1 ([3]). A coloring $c(G)$ of the mixed graph $G = (V, A, E)$ exists if and only if the digraph (V, A, \emptyset) has no circuit containing adjacent vertices in the graph (V, \emptyset, E) .

The mixed graph $G = (V, A, E)$ is colorable if there exists a coloring $c(G)$ of the mixed graph $G = (V, A, E)$; otherwise, the mixed graph $G = (V, A, E)$ is un-colorable.

In articles [4,5], it is shown that a job-shop scheduling problem with unit processing times of the operations and the minimization of the makespan may be represented as an optimal coloring $c(G)$ of the specified mixed graph $G = (V, A, E)$.

Finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ is an NP-hard problem even if the equality $A = \emptyset$ holds [6].

In article [7], it is shown that any job-shop scheduling problem with unit processing times of the operations to minimize the total completion time $\sum C_i$ may be represented as a mixed graph coloring with the minimization of the sum of colors of path-endpoints in the mixed graph $G = (V, A, E)$; see [8,9]. Hereafter, C_i denotes a completion time of the job J_i . The unit-time scheduling problem with the minimization of the makespan is NP-hard even for three dedicated machines [10]. The unit-time job-shop scheduling problem with the minimization of the total completion time $\sum C_i$ is also NP-hard [11].

Articles [12–15] investigate the complexity of the job-shop scheduling problem with a fixed number of jobs and a fixed number of machines, provided that a job may be processed several times by the same machine. Articles [13,16] investigate the complexity of the job-shop scheduling problems with any fixed regular objective function. Articles [17–23] study different types of connections between mixed graph colorings and unit-time shop-scheduling problems. Article [24] is a survey on the mixed graph coloring problems and the equivalent unit-time shop-scheduling problems.

In this article, we show that any optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ is equivalent to finding an optimal schedule for the corresponding partially ordered multiprocessor tasks with unit processing times and the minimization of maximal lateness L_{\max} . Contrary to a classical shop-scheduling problem, several dedicated machines must process a multiprocessor task. Along with two types of the precedence constraints given on the set $V = \{v_1, v_2, \dots, v_n\}$ of the multiprocessor tasks, it is necessary to process a specified subset of the tasks simultaneously.

The equivalence of the considered scheduling problems and the corresponding mixed graph coloring problems implies that most analytical results and algorithms developed for

a wide class of the shop-scheduling problems generate analogous results and algorithms for optimal mixed graph colorings $c(G)$, and vice versa.

In what follows, we use the terminology from books [25,26] for graph theory and that from books [27,28] for scheduling theory.

2. Closed Results Published on the Mixed Graph Coloring Problems and the Equivalent Unit-Time Shop-Scheduling Problems to Minimize the Makespan

To classify different scheduling problems, one can use a three-field notation $\alpha|\beta|\gamma$ introduced in [29], where field α specifies a processing system along with a machine number, field β job characteristics, and field γ an objective function [28].

2.1. A Unit-Time Minimum-Length Job-Shop Scheduling Problem

We first consider the following job-shop scheduling problem $J|t_i = 1|C_{\max}$ with unit processing times of the given operations and the objective criterion C_{\max} to minimize the makespan. In the job-shop scheduling problem $J|t_i = 1|C_{\max}$, a set of the jobs $J = \{J_1, J_2, \dots, J_{|J|}\}$ must be optimally processed on the different (dedicated) machines $M = \{M_1, M_2, \dots, M_{|M|}\}$. A job $J_k \in J$ consists of the set $V^{(k)} = \{v_{k_1}, v_{k_2}, \dots, v_{k_{|V^{(k)}|}}\}$ of the linearly ordered operations: $(v_{k_1}, v_{k_2}, \dots, v_{k_{|V^{(k)}|}})$. A specified machine from the set M is required to process (without preemptions) the operation v_i from the set $V = \bigcup_{k=1}^{|J|} V^{(k)}$. The processing time t_i of each operation v_i in the set V is equal to 1.

Let the set $V_i = \{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\} \subseteq V$ denote a set of all operations processed by machine $M_i \in M$. Any pair of the operations processed by the same machine cannot be processed simultaneously in any feasible schedule.

In order to solve the job-shop scheduling problem $J|t_i = 1|C_{\max}$, it is necessary to find a non-preemptive schedule (each operation must be processed without preemptions), whose length $C_{\max} = \max\{C_1, C_2, \dots, C_{|J|}\}$ (the makespan) is minimum among lengths of all feasible schedules, where the completion time C_k of the job $J_k \in J$ is equal to the completion time $c_{k_{|V^{(k)}|}}$ of the last operation $v_{k_{|V^{(k)}|}}$ of the job $J_k \in J$.

In articles [4,5,8,9], it is shown that the job-shop scheduling problem $J|t_i = 1|C_{\max}$ may be represented as an optimal coloring $c(G)$ of the specified mixed graph $G = (V, A, E)$. Based on Definition 1, one can represent every job $J_k \in J$ as a union of the path $(v_{k_1}, v_{k_2}, \dots, v_{k_{r_k}})$ in the directed subgraph (V, A, \emptyset) of the mixed graph $G = (V, A, E)$ and the chain $(v_{k_1}, v_{k_2}, \dots, v_{k_{r_k}})$ in the subgraph (V, \emptyset, E) . As a result, we determine the vertex set $V = \bigcup_{k=1}^{|J|} V^{(k)}$ of the mixed graph $G = (V, A, E)$, the set of the arcs $A = \bigcup_{k=1}^{|J|} \{(v_{k_1}, v_{k_2}), (v_{k_2}, v_{k_3}), \dots, (v_{k_{r_k}-1}, v_{k_{r_k}})\}$, and the subset E^* of the edge set E , which are defined based on the following implications:

$$(v_p, v_q) \in A \Rightarrow [v_p, v_q] \in E^* \tag{1}$$

Due to Definition 1, the remaining subset $E \setminus E^*$ of the edges in the subgraph $(V, \emptyset, E \setminus E^*)$ of the mixed graph $G = (V, A, E)$ is determined by $|M|$ cliques $\{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\}$, where the set $V_i = \{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\}$ consists of all operations processed by machine $M_i \in M$. Due to this, any pair of operations from the set $V_i \subseteq V$ cannot be processed simultaneously. It is clear that the constructed mixed graph $G = (V, A, E)$ determines the complete input data for the job-shop scheduling problem $J|t_i = 1|C_{\max}$; see articles [4,5] for details. Thus, one can call such a scheduling problem as a job-shop scheduling problem $J|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$.

Article [4] shows that if the mixed graph $G = (V, A, E)$ determines the input data for the job-shop scheduling problem $J|t_i = 1|C_{\max}$, then this mixed graph $G = (V, A, E)$ must possess the following mandatory properties.

Property 1. The following partition: $(V, A, \emptyset) = (V^{(1)}, A^{(1)}, \emptyset) \cup (V^{(2)}, A^{(2)}, \emptyset) \cup \dots \cup (V^{(r)}, A^{(r)}, \emptyset)$ holds, where each directed subgraph $(V^{(k)}, A^{(k)}, \emptyset)$ of the mixed graph $G = (V, A, E)$ is a path $(v_{k_1}, v_{k_2}, \dots, v_{k_{r_k}})$ for each index $k \in \{1, 2, \dots, r\}$.

Property 2. Edge $[v_i, v_j]$ belongs to the set E^* if and only if the implication (1) holds.

Property 3. The following partition: $(V, \emptyset, E \setminus E^*) = (V_1, \emptyset, E_1) \cup (V_2, \emptyset, E_2) \cup \dots \cup (V_m, \emptyset, E_m)$ holds, where each subgraph (V_k, \emptyset, E_k) of the mixed graph $G = (V, A, E)$ is a complete graph for each index $k \in \{1, 2, \dots, m\}$.

Article [4] contains the proof of the following theorem.

Theorem 2 ([4]). Any job-shop scheduling problem $J|t_i = 1|C_{\max}$ on mixed graph $G = (V, A, E)$ is equivalent to optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. Conversely, any optimal coloring $c(G)$ of mixed graph $G = (V, A, E)$ possessing properties 1–3 is equivalent to a job-shop scheduling problem $J|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$

To illustrate Theorem 2, we consider the following Example 1 of the job-shop scheduling problem $J|t_i = 1|C_{\max}$ with four jobs and five machines; see Figure 1.

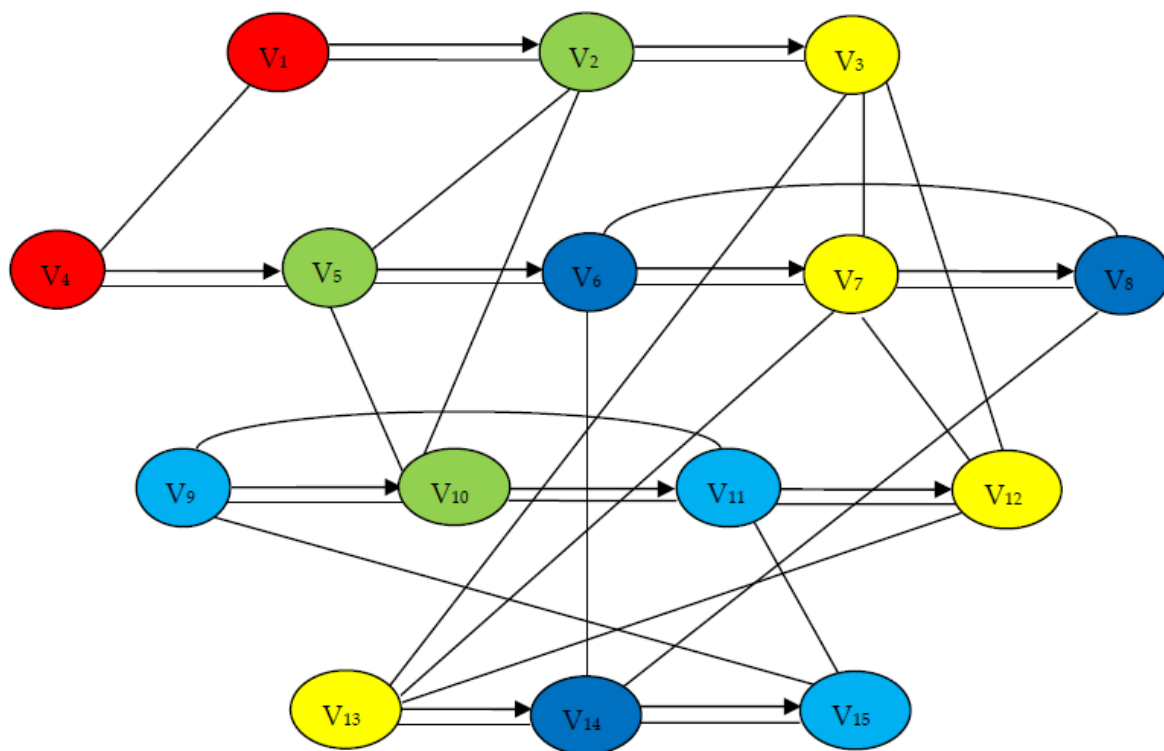


Figure 1. Mixed graph $G = (V, A, E)$ determining the input data for Example 1 of the job-shop scheduling problem $J|t_i = 1|C_{\max}$, optimal mixed graph coloring $c(G)$ being equivalent to Example 1.

Example 1. The machine set $\{M_1, M_2, \dots, M_5\} = M$ have to process the job set $\{J_1, J_2, J_3, J_4\} = J$, where the job $J_1 \in J$ includes the set $V^{(1)} = \{v_1, v_2, v_3\}$ of the linearly ordered operations: (v_1, v_2, v_3) . The job $J_1 \in J$ is represented by a union of the path (v_1, v_2, v_3) in the digraph (V, A, \emptyset) and the chain (v_1, v_2, v_3) in the graph (V, \emptyset, E) . The job $J_2 \in J$ includes the set $V^{(2)} = \{v_4, v_5, v_6, v_7, v_8\}$ of the linearly ordered operations: $(v_4, v_5, v_6, v_7, v_8)$. The job $J_2 \in J$ is represented by a union of the path $(v_4, v_5, v_6, v_7, v_8)$ in the digraph (V, A, \emptyset) and the chain $(v_4, v_5, v_6, v_7, v_8)$ in the graph (V, \emptyset, E) . The job $J_3 \in J$ includes the set $V^{(3)} = \{v_9, v_{10}, v_{11}, v_{12}\}$ of the linearly ordered operations: $(v_9, v_{10}, v_{11}, v_{12})$. The job $J_3 \in J$

is represented by a union of the path $(v_9, v_{10}, v_{11}, v_{12})$ in the digraph (V, A, \emptyset) and the chain $(v_9, v_{10}, v_{11}, v_{12})$ in the graph (V, \emptyset, E) . The job $J_4 \in J$ includes the set $V^{(4)} = (v_{13}, v_{14}, v_{15})$ of the linearly ordered operations: (v_{13}, v_{14}, v_{15}) . The job $J_4 \in J$ is represented by a union of the path (v_{13}, v_{14}, v_{15}) in the digraph (V, A, \emptyset) and the chain (v_{13}, v_{14}, v_{15}) in the graph (V, \emptyset, E) . The set $V_1 = \{v_1, v_4\}$ is a set of operations processed by machine M_1 .

In Figure 1, the forbiddance to process the operations from the set V_1 simultaneously is determined by the clique $\{v_1, v_4\}$ in the graph (V, \emptyset, E) . Machine M_2 processes the operations $V_2 = \{v_2, v_5, v_{10}\}$. The forbiddance to process a pair of operations from the set V_2 simultaneously is determined by the clique $\{v_2, v_5, v_{10}\}$. Machine M_3 processes the operations $V_3 = \{v_3, v_7, v_{12}, v_{13}\}$. The forbiddance to process operations from the set V_2 simultaneously is determined by the clique $\{v_3, v_7, v_{12}, v_{13}\}$. Machine M_4 processes the operations $V_3 = \{v_9, v_{11}, v_{15}\}$. The forbiddance to process a pair of operations from the set V_2 simultaneously is determined by the clique $\{v_9, v_{11}, v_{15}\}$. Machine M_5 processes the operations $V_5 = \{v_6, v_8, v_{14}\}$. The forbiddance to process a pair of operations from the set V_2 simultaneously is determined by the clique $\{v_6, v_8, v_{14}\}$.

In Figure 1, a specified color is used to indicate all operations $V_i = \{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\} \subseteq V$ processed by the same machine $M_i \in M$.

Note that the mixed graph $G = (V, A, E)$ depicted in Figure 1 possesses properties 1–3. An optimal coloring $c(G)$ of this mixed graph $G = (V, A, E)$ is determined as follows: $c(v_1) = 2, c(v_2) = 4, c(v_3) = 6, c(v_4) = 1, c(v_5) = 2, c(v_6) = 3, c(v_7) = 4, c(v_8) = 5, c(v_9) = 1, c(v_{10}) = 3, c(v_{11}) = 4, c(v_{12}) = 5, c(v_{13}) = 1, c(v_{14}) = 2, c(v_{15}) = 3$. Due to Theorem 2, this coloring $c(G)$ of the mixed graph $G = (V, A, E)$ determines an optimal schedule for the job-shop scheduling problem $J|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$ depicted in Figure 1. Thus, the equalities $\chi(G) = 6 = C_{\max}$ hold.

2.2. A General Shop Unit-Time Scheduling Problem to Minimize the Makespan

The general shop scheduling problem $G|t_i = 1|C_{\max}$ is a generalization of the job-shop scheduling problem $J|t_i = 1|C_{\max}$ considered in Section 2.1. In the former problem $G|t_i = 1|C_{\max}$, along with the linear orders given on the sets $V^{(k)}, J_k \in J$, there are also given the precedence constraints between operations belonging to different jobs in the set J . We next consider two types of the precedence constraints provided that the problem $G|t_i = 1|C_{\max}$ be modeled as a coloring $c(G)$ of the mixed graph $G = (V, A, E)$.

If the completion time c_{k_p} of the operation v_{k_p} of the job $J_k \in J$ has to precede the start time s_{l_q} of the operation v_{l_q} of the job $J_l \in J$, where $k \neq l$, then the mixed graph $G = (V, A, E)$ must contain both arc $(v_{k_p}, v_{l_q}) \in A$ and edge $[v_{k_p}, v_{l_q}] \in E$. In what follows, we denote this type of the precedence constraints as: $v_{k_p} \rightarrow v_{l_q}$. Hence, if the precedence constraint $v_p \rightarrow v_q$ holds for the general shop scheduling problem $G|t_i = 1|C_{\max}$, then the implication (1) must hold for the arc $(v_p, v_q) \in A$, i.e., $(v_p, v_q) \in A \Rightarrow [v_p, v_q] \in E^*$, where $E^* \subseteq E$.

If the start time s_{k_p} of the operation v_{k_p} of the job $J_k \in J$ has to precede the start time s_{l_q} of the operation v_{l_q} of the job $J_l \in J$, where $k \neq l$, then the mixed graph $G = (V, A, E)$ must contain only arc $(v_{k_p}, v_{l_q}) \in A$ as an addition. In what follows, we denote this type of the precedence constraints as: $v_{k_p} \mapsto v_{l_q}$. Hence, if the precedence constraint $v_p \mapsto v_q$ holds for the general shop scheduling problem $G|t_i = 1|C_{\max}$, then the implication (1) does not hold for the arc $(v_p, v_q) \in A$.

Article [23] contains the proof of Theorem 3 on the connection between the mixed graph coloring problem and the general shop scheduling problem $G|t_i = 1|C_{\max}$.

Theorem 3 ([23]). Any general shop scheduling problem $G|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$ is represented as an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.

Note that the inverse claim to Theorem 3 is not correct, i.e., there exists a problem of optimal coloring of the mixed graph, which cannot be represented as a general shop

scheduling problem $G|t_i = 1|C_{\max}$. We next consider Example 2 of the general shop scheduling problem $G|t_i = 1|C_{\max}$ on the mixed graph $G^* = (V, A^*, E^*)$, which defers from the mixed graph $G = (V, A, E)$ depicted in Figure 1 by three arcs and one edge.

Example 2. The general shop scheduling problem $G|t_i = 1|C_{\max}$ on the mixed graph $G^* = (V, A^*, E^*)$ is the same as Example 1 of the problem $J|t_i = 1|C_{\max}$ with only exception that there are precedence constraints between operations belonging to different jobs J_k from the set J . These additional precedence constraints are given as follows: $v_7 \mapsto v_1$; $v_{12} \mapsto v_5$; and $v_{14} \mapsto v_9$. Thus, for the mixed graph $G^* = (V, A^*, E^*)$, the following equalities hold: $A^* = A \cup \{(v_7, v_1), (v_{12}, v_5), (v_{14}, v_9)\}$; $E^* = E \cup \{[v_9, v_{14}]\}$.

An optimal schedule for Example 2 is determined by the following optimal coloring $c(G^*)$ of the mixed graph $G^* = (V, A^*, E^*)$: $c(v_1) = 8, c(v_2) = 9, c(v_3) = 10, c(v_4) = 1, c(v_5) = 6, c(v_6) = 7, c(v_7) = 8, c(v_8) = 9, c(v_9) = 3, c(v_{10}) = 4, c(v_{11}) = 5, c(v_{12}) = 6, c(v_{13}) = 1, c(v_{14}) = 2, c(v_{15}) = 4$. This coloring $c(G^*)$ is optimal; $\chi(G^*) = 10$. The optimality of the schedule determined by the coloring $c(G^*)$ follows from the fact that the mixed graph $G^* = (V, A^*, E^*)$ includes the path $(v_{13}, v_{14}, v_9, v_{10}, v_{11}, v_{12}, v_5, v_6, v_7, v_1, v_2, v_3)$, whose length is equal to 10. Thus, the inequality $\chi(G^*) \geq 10$ must hold.

2.3. Scheduling Multiprocessor Tasks with Unit Durations

Contrary to the above classical problems $J|t_i = 1|C_{\max}$ and $G|t_i = 1|C_{\max}$, where a single machine (processor) fulfills an operation, in the processing system with multiprocessor tasks [28], a task (operation) requires either one machine or several machines during the fulfillment of the multiprocessor task (MPT for short). As usual, two tasks requiring at least one common machine cannot be processed simultaneously.

Chapter 10 of the book [28] (pp. 264–283) presents the complexity results for the general shop minimum-length scheduling problem $GMPT|t_i = 1|C_{\max}$ with multiprocessor tasks. To solve the scheduling problem $GMPT|t_i = 1|C_{\max}$ means to construct an optimal schedule for processing the partially ordered multiprocessor tasks (operations) $V = \{v_1, v_2, \dots, v_n\}$ by the machines $M = \{M_1, M_2, \dots, M_{|M|}\}$.

There has been increasing interest in multiprocessor scheduling, i.e., in scheduling models, where tasks require several processors simultaneously. Many scheduling problems fit in this model and a large amount of research has been carried on theoretical multiprocessor scheduling. Due to a wide practical importance scheduling problems with multiprocessor tasks have attracted considerable attention from researchers; see [30–41]. The main part of article [34] is a presentation of the results in multiprocessor tasks scheduling both for parallel and for dedicated processors. The problems $GMPT|t_i = 1|\gamma$, with unit processing times have been considered in articles [30–34].

The general shop scheduling problem $G|t_i = 1|C_{\max}$ is a special case of the scheduling problem $GMPT|t_i = 1|C_{\max}$. Article [23] studies more general shop problem $GcMPT|t_i = 1|C_{\max}$ including the problem $GMPT|t_i = 1|C_{\max}$ as a special case.

Let two types of the precedence constraints $v_{k_p} \rightarrow v_{l_q}$ and $v_{k_p} \mapsto v_{l_q}$ be given in the problem $GcMPT|t_i = 1|C_{\max}$. Furthermore, it is required that a subset $V(k) = \{v_{k_1}, v_{k_2}, \dots, v_{k_{|V(k)|}}\}$ of the tasks $V = \{v_1, v_2, \dots, v_n\} \supseteq V(k)$ must be processed simultaneously. In order to present the latter requirement, a circuit $(v_{k_1}, v_{k_2}, \dots, v_{k_{|V(k)|}}, v_{k_1})$ is included to the directed subgraph (V, A, \emptyset) of the mixed graph $G = (V, A, E)$, where the following relations hold: $A'_c = \{(v_{k_1}, v_{k_2}), (v_{k_2}, v_{k_3}), \dots, (v_{k_{|V(k)|-1}}, v_{k_{|V(k)|}}), (v_{k_{|V(k)|}}, v_{k_1})\} \subseteq A$.

Let the input data of the general shop problem $GcMPT|t_i = 1|C_{\max}$ include w subsets $V(1), V(2), \dots, V(w)$ of the set $V = \{v_1, v_2, \dots, v_n\}$ such that all tasks in the set

$\{v_{k_1}, v_{k_2}, \dots, v_{k_{|V(k)|}}\} = V(k) \subseteq V$, must be processed simultaneously in any feasible schedule. We obtain the following A_c subset of the set A of arcs:

$$A_c = \bigcup_{k=1}^w \{(v_{k_1}, v_{k_2}), (v_{k_2}, v_{k_3}), \dots, (v_{k_{|V(k)|-1}}, v_{k_{|V(k)|}}), (v_{k_{|V(k)|}}, v_{k_1})\} \subseteq A \quad (2)$$

Every example of the general shop scheduling problem $GcMPT|t_i = 1|C_{\max}$ uniquely determines a mixed graph $G = (V, A, E)$ with $A_c \subseteq A$, which presents the input data for this example. Thus, to determine an example of the general shop scheduling problem $GcMPT|t_i = 1|C_{\max}$, it is sufficient to determine the mixed graph $G = (V, A, E)$ for this example. Such an example is called a problem $GcMPT|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$. Article [23] contains the proofs of the following two theorems.

Theorem 4 ([23]). *A feasible schedule for the general shop scheduling problem $GcMPT|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$ exists if and only if the digraph (V, A, \emptyset) has no circuit containing adjacent vertices in the graph (V, \emptyset, E) .*

Theorem 5 ([23]). *Any solvable general shop scheduling problem $GcMPT|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$ is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. Conversely, for any colorable mixed graph $G = (V, A, E)$, there exists a general shop scheduling problem $GcMPT|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$, which is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.*

To illustrate Theorems 4 and 5, we consider Example 3 of the problem $GcMPT|t_i = 1|C_{\max}$ on the mixed graph $G = (V, A, E)$ presented in Figure 2. The job $J_1 \in J$ includes the set $V^{(1)} = \{v_1, v_2, v_3\}$ of the linearly ordered operations: (v_1, v_2, v_3) . In Figure 2, the job $J_1 \in J$ is presented by a union of the path (v_1, v_2, v_3) in the digraph (V, A, \emptyset) and the chain (v_1, v_2, v_3) in the graph (V, \emptyset, E) . The job $J_2 \in J$ includes the set $V^{(2)} = \{v_4, v_5, v_6\}$ of the linearly ordered operations: (v_4, v_5, v_6) . The job $J_2 \in J$ is presented by a union of the path (v_4, v_5, v_6) in the digraph (V, A, \emptyset) and the chain (v_4, v_5, v_6) in the graph (V, \emptyset, E) . The job $J_3 \in J$ includes the set $V^{(3)} = \{v_7, v_8, v_9\}$ of the linearly ordered operations: (v_7, v_8, v_9) . The job $J_3 \in J$ is presented by a union of the path (v_7, v_8, v_9) in the digraph (V, A, \emptyset) and the chain (v_7, v_8, v_9) in the graph (V, \emptyset, E) . The job $J_4 \in J$ includes the set $V^{(4)} = \{v_{10}, v_{11}, v_{12}\}$ of the linearly ordered operations: (v_{10}, v_{11}, v_{12}) . The job $J_4 \in J$ is presented by a union of the path (v_{10}, v_{11}, v_{12}) in the digraph (V, A, \emptyset) and the chain (v_{10}, v_{11}, v_{12}) in the graph (V, \emptyset, E) .

Machine M_1 processes the operations $V_1 = \{v_1, v_4\}$. In Figure 2, the forbiddance to process the operations from the set V_1 simultaneously is determined by the edge $[v_1, v_4]$ in the graph (V, \emptyset, E) . Machine M_2 processes the operations $V_2 = \{v_2, v_5\}$. The forbiddance to process the operations from the set V_2 simultaneously is determined by the edge $[v_2, v_5]$. Machine M_3 processes the operations $V_3 = \{v_2, v_4\}$. The forbiddance to process any pair of operations from the set V_3 simultaneously is determined by the edge $[v_2, v_4]$. Machine M_4 processes the operations $V_4 = \{v_5, v_8, v_{10}\}$. The forbiddance to process any pair of operations from the set V_4 simultaneously is determined by the clique $\{v_5, v_8, v_{10}\}$ in the graph (V, \emptyset, E) . Machine M_5 processes the operations $V_5 = \{v_6, v_9, v_{11}\}$. The forbiddance to process any pair of operations from the set V_5 simultaneously is determined by the clique $\{v_6, v_9, v_{11}\}$. Machine M_6 processes the operations $V_6 = \{v_7, v_9\}$. The forbiddance to process the operations from the set V_6 simultaneously is determined by the edge $[v_7, v_9]$. Machine M_7 processes the operations $V_7 = \{v_2, v_8\}$. The forbiddance to process the operations from the set V_7 simultaneously is determined by the edge $[v_2, v_8]$. Machine M_8 processes the operations $V_8 = \{v_9, v_{10}\}$. The forbiddance to process the operations from the set V_8 simultaneously is determined by the edge $[v_9, v_{10}]$. Machine M_9 processes the operations $V_9 = \{v_7, v_{12}\}$. The forbiddance to process the operations from the set V_9 simultaneously is determined by the edge $[v_7, v_{12}]$. Machine M_{10} processes one operation

$V_{10} = \{v_3\}$. The machines, which are used to process the task $v_i \in V$, are presented near the vertex v_i in Figure 2.

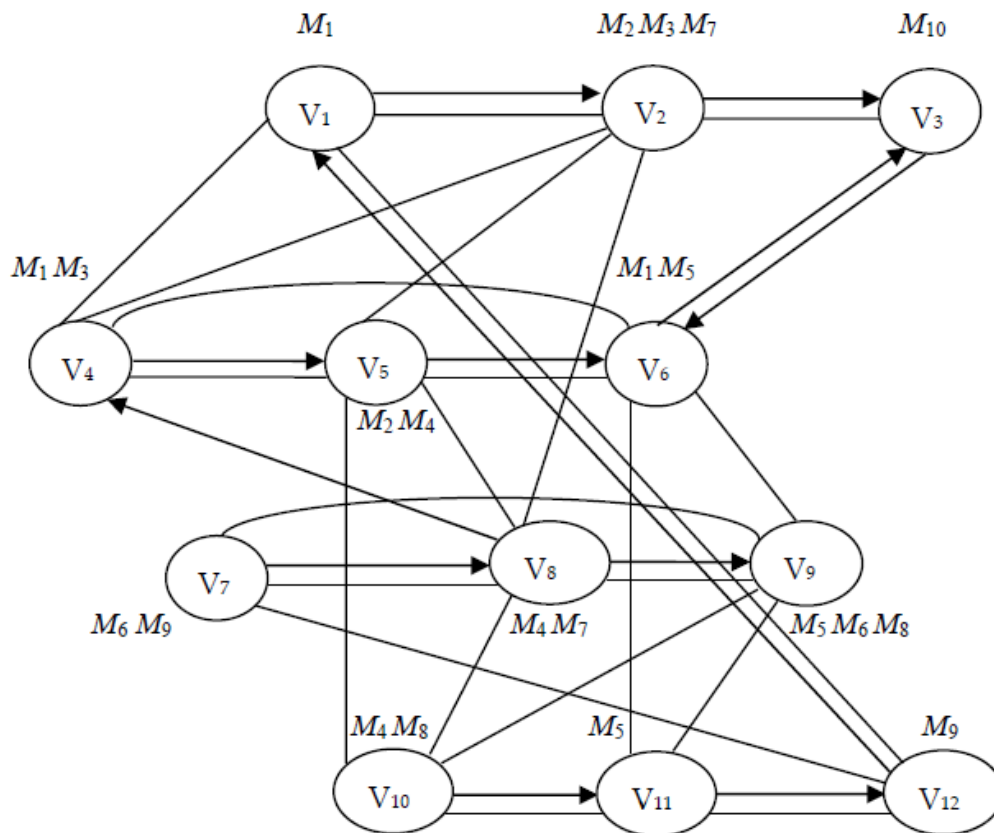


Figure 2. Mixed graph $G = (V, A, E)$ determining Example 3 of the general shop scheduling problem $GcMPT|t_i = 1|C_{max}$ with 4 jobs and 10 machines, optimal mixed graph coloring $c(G)$ being equivalent to Example 3.

The precedence relations between operations belonging to different jobs are given as follows: $v_3 \mapsto v_6$; $v_6 \mapsto v_3$; $v_8 \mapsto v_4$; and $v_{12} \mapsto v_1$. These precedence constraints are presented by arcs $\{(v_3, v_6), (v_6, v_3), (v_8, v_4), (v_{12}, v_1)\} \subset A$ and edge $[v_1, v_{12}] \in E$.

The digraph (V, A, \emptyset) has only one circuit (v_3, v_6, v_3) and this circuit has no adjacent vertices in the graph (V, \emptyset, E) . Since the digraph (V, A, \emptyset) has no circuit containing adjacent vertices in the graph (V, \emptyset, E) , there exists a feasible schedule for Example 3 of the problem $GcMPT|t_i = 1|C_{max}$ on the mixed graph $G = (V, A, E)$; see Theorem 4.

Due to Theorem 5, an optimal schedule for Example 3 is determined by the following optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$: $c(v_1) = 4$, $c(v_2) = 5$, $c(v_3) = 6$, $c(v_4) = 2$, $c(v_5) = 3$, $c(v_6) = 6$, $c(v_7) = 1$, $c(v_8) = 2$, $c(v_9) = 3$, $c(v_{10}) = 1$, $c(v_{11}) = 2$, $c(v_{12}) = 3$. The coloring $c(G)$ is optimal; $\chi(G) = 6$. The optimality of the schedule determined by the coloring $c(G)$ follows from the fact that the mixed graph $G = (V, A, E)$ includes the path $(v_{10}, v_{11}, v_{12}, v_1, v_2, v_3)$, whose length is equal to 6. Therefore, the non-strict inequality $\chi(G) \geq 6$ must hold.

3. Two Equivalent Problems of Scheduling Unit-Time Multiprocessing Tasks as Optimal Colorings of the Mixed Graphs

We next prove that any general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ with non-negative integer release times $r_i \geq 0$ of the jobs $J_i \in J$ is equivalent to an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. Since all release times r_i are non-negative integer numbers, the input data for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ may be represented by the mixed graph $G = (V, A, E)$. Thus, Theorem 4 is correct for this scheduling problem as well.

3.1. Scheduling Unit-Time Multiprocessing Tasks to Minimize the Makespan as an Optimal Mixed Graph Coloring Problem

In the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$, a multiprocessor task $v_i \in V$ may be regarded as a job J_i . The job J_i may include either one task (operation) $v_i \in V$ or more than one task (several operations).

Let a simple job be a job consisting of a single task (operation).

Theorem 6. Any solvable general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. Conversely, for any colorable mixed graph $G = (V, A, E)$, there exists a general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$, which is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.

Proof. Every solvable general shop scheduling problem $GcMPT|t_i = 1|C_{max}$ defines a mixed graph $G' = (V', A', E')$ presenting the input data for this problem. Using this mixed graph $G' = (V', A', E')$, we construct a mixed graph $G = (V, A, E)$ determining the input data for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$. To this end, for each job $J_i \in J$, we add vertices $v_{i_1^0}, \dots, v_{i_{r_i-1}^0}$ to the set V' and path $(v_{i_1^0}, \dots, v_{i_{r_i-1}^0}, v_{i_1})$, whose length is equal to r_i , to the directed subgraph (V', A', \emptyset) of the mixed graph G' . We also add the chain $(v_{i_1^0}, \dots, v_{i_{r_i-1}^0}, v_{i_1})$, whose length is equal to r_i , to the subgraph (V', \emptyset, E') of the mixed graph G' . Let $G = (V, A, E)$ denote a mixed graph obtained from the mixed graph $G' = (V', A', E')$, where $V = V' \cup \left\{ \bigcup_{i=1}^n \left\{ v_{i_1^0}, \dots, v_{i_{r_i-1}^0} \right\} \right\} =: V' \cup V^0$. It is easy to see that the constructed mixed graph $G = (V, A, E)$ determines the input data for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$. Since there is a feasible schedule for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$, the directed subgraph (V, A, \emptyset) of the mixed graph $G = (V, A, E)$ has no circuit containing the adjacent vertices in the graph (V, \emptyset, E) (due to Theorems 1 and 4). \square

Any solvable example of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ due to the correspondence of the terms used in the optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ and those used in the optimal schedule existing for this example of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$. Table 1 presents the correspondence of these terms.

Due to the existence of the clique $\{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\}$ in the graph (V, \emptyset, E) , each machine $M_i \in M$ in the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ can process at most one task within a unit-time semi-interval from the following set:

$$\{[0, 1], (1, 2], (2, 3], \dots, (t - 1, t]\} \tag{3}$$

Furthermore, it is easy to see that an optimal coloring $c : V \rightarrow \{1, 2, \dots, \chi(G)\}$ of the mixed graph $G = (V, A, E)$ determines the optimal assignment of the tasks V to the following minimal number of the unit-time semi-intervals:

$$\{[0, 1], (1, 2], (2, 3], \dots, (\chi(G) - 1, \chi(G)]\} \tag{4}$$

Such an assignment of the tasks (operations) V to the minimal number of the unit-time semi-intervals (4) is optimal since it determines a feasible schedule for processing the tasks (operations) V , whose length is equal to the chromatic number $\chi(G)$ of the mixed graph $G = (V, A, E)$ that determines the input data for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$.

Due to the correspondence of terms used in the coloring $c(G)$ of the mixed graph $G = (V, A, E)$ and those used in the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$

on the mixed graph $G = (V, A, E)$ (see Table 1), one can conclude that *any solvable example of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.*

Table 1. The correspondence of terms used in mixed graph coloring $c(G)$ and those used in the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$.

Terms of Mixed Graph Coloring $c(G)$	Corresponding Terms of General Shop Scheduling Problem $GcMPT t_i=1,[r_i] C_{max}$
Vertex $v_i \in V \setminus V^0$	Unit-time task $v_i \in V \setminus V^0$ (operation)
Vertices on the path (chain) $(v_{k_1}, v_{k_2}, \dots, v_{k_{ V(k) }})$ in the digraph (V, A, \emptyset) (in the graph (V, \emptyset, E))	A set $V^{(k)} = \{v_{k_1}, v_{k_2}, \dots, v_{k_{ V(k) }}\}$ of the linearly ordered tasks $(v_{k_1}, v_{k_2}, \dots, v_{k_{ V(k) }})$ of the job $J_k \in J$
A path (chain) $(v_{i_1^0}, \dots, v_{i_{r_i-1}^0}, v_{i_1})$ of the length r_i in the digraph (V, A, \emptyset) (in the graph (V, \emptyset, E))	A release time $r_i \geq 0$ of the job $J_i \in J$
A clique $\{v_{i_1}, v_{i_2}, \dots, v_{i_{ V_i }}\}$ in the graph (V, \emptyset, E)	Tasks $V_i = \{v_{i_1}, v_{i_2}, \dots, v_{i_{ V_i }}\}$ processed by machine $M_i \in M$
Arc (v_i, v_j) in the digraph (V, A, \emptyset)	A precedence constraint $v_i \mapsto v_j$ determined between operations belonging to different jobs
Arc (v_p, v_q) in the digraph (V, A, \emptyset) and edge $[v_p, v_q]$ in the graph (V, \emptyset, E)	A precedence constraint $v_p \rightarrow v_q$ determined between tasks (operations) belonging to different jobs
A circuit $(v_{k_1}, v_{k_2}, \dots, v_{k_{ V(k) }}, v_{k_1})$ in the digraph (V, A, \emptyset) , where $A_c \subseteq A$	Tasks $V(k) = \{v_{k_1}, v_{k_2}, \dots, v_{k_{ V(k) }}\} \subseteq V$, which must be processed simultaneously
A coloring $c(G)$ of the mixed graph $G = (V, A, E)$	A schedule for the problem $GcMPT t_i = 1, [r_i] C_{max}$
An optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$	An optimal schedule for the problem $GcMPT t_i = 1, [r_i] C_{max}$ on the mixed graph $G = (V, A, E)$
The chromatic number $\chi(G)$	The minimal value of makespan C_{max}

Next, we prove the following converse claim: *for any colorable mixed graph $G = (V, A, E)$, there exists a general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph G , which is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.* To this end, we detect a set Ω of all circuits existing in the directed subgraph (V, A, \emptyset) of the mixed graph $G = (V, A, E)$ and consider two possible cases.

Case I. Let the set Ω be empty; $\Omega = \emptyset$.

Then, the desired general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ may be determined using the following Algorithm 1.

Algorithm 1. Constructing the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$, which is equivalent to the optimal coloring $c(G)$

Input: A mixed graph $G = (V, A, E)$ without circuits in the digraph (V, A, \emptyset) .

Output: A general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$, which is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.

Step 1: Partition the graph (V, \emptyset, E) into maximum connected components: $(V, \emptyset, E) = (V_1, \emptyset, E_1) \cup \dots \cup (V_m, \emptyset, E_m) \cup (V_{m+1}, \emptyset, \emptyset) \dots \cup (V_{m+r}, \emptyset, \emptyset)$, where the subgraph (V_k, \emptyset, E_k) is a maximum (with respect to the inclusion) connected component of the graph (V, \emptyset, E) for each index $k \in \{1, \dots, m\}$ such that $|V_k| \geq 2$. Let the subgraph $(V_j, \emptyset, \emptyset)$ determine an isolated vertex for each index $j \in \{m + 1, \dots, m + r\}$. Denote this isolated vertex as: $\{v_{j_1}\} = V_j$. Set $M = \emptyset, k = 1, i = 0$, and $l_0 = 0$.

Step 2: **IF** $k = m + 1$ **GOTO** step 5 **ELSE** find all maximum (with respect to the inclusion) complete vertex-induced subgraphs $(V_k^1, \emptyset, E_k^1), \dots, (V_k^k, \emptyset, E_k^k)$ of the graph (V_k, \emptyset, E_k) . Set $r = 1$ and $i := i + l_{k-1} + 1$.

Step 3: **FOR** index i , supplement machine M_i to the already constructed machine set; $M := M \cup \{M_i\}$. Establish that all tasks in the clique V_k^r of the connected graph (V_k, \emptyset, E_k) must be processed by machine M_i ; $V_k^r = V_i = \{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\}$, where all tasks $\{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\}$ must be processed by machine M_i in any feasible schedule. Set $i := i + 1$.

Step 4: **IF** $i = \sum_{h=0}^k l_h$ **THEN** set $k := k + 1$ **GOTO** step 2 **ELSE** set $r := r + 1$ **GOTO** step 3.

Step 5: **FOR** each index $j \in \{m + 1, \dots, m + r\}$, supplement machine M_{i+j-m} to the already constructed machine set M ; $M := M \cup \{M_{i+j-m}\}$. Establish that task v_{j_1} , $V_j = \{v_{j_1}\}$, which is isolated in the graph (V, \emptyset, E) , must be processed by machine M_{i+j-m} . Establish that machine M_{i+j} must process only task v_{j_1} . Set $M := M \cup \{M_{i+1}, \dots, M_{i+r}\}$.

Step 6: **FOR** each arc $(v_p, v_q) \in A$ such that the implication (1) holds and $E^* \subseteq E$, determine the precedence constraint $v_p \rightarrow v_q$, which means that processing the task v_p must be completed before starting the task v_q in any feasible schedule.

Step 7: **FOR** each arc $(v_p, v_q) \in A$ such that the implication (1) does not hold, determine the precedence constraint $v_p \mapsto v_q$, which means that processing the task v_p must be started before the start time of the task v_q in any feasible schedule.

Step 8: The desired general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$ is constructed, where the precedence constraints on the task set V are determined at step 6 and step 7. Further, the set M of the machines is determined at step 3 and step 5 **STOP**.

Case II. Let the set Ω be not empty; $\Omega \neq \emptyset$.

Since the mixed graph $G = (V, A, E)$ is colorable, each circuit $(v_{k_1}, v_{k_2}, \dots, v_{k_{|V(k)|}}, v_{k_1})$ in the set Ω has no adjacent vertices in the graph (V, \emptyset, E) (due to Theorem 1). Thus, all tasks $\{v_{k_1}, v_{k_2}, \dots, v_{k_{|V(k)|}}\} = V(k)$ must be processed simultaneously in any feasible schedule existing for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$, where the circuit $(v_{k_1}, v_{k_2}, \dots, v_{k_{|V(k)|}}, v_{k_1})$ exists in the directed subgraph (V, A, \emptyset) .

Let $\Omega = \bigcup_{k=1}^w V(k) = \bigcup_{k=1}^w \{(v_{k_1}, v_{k_2}, v_{k_3}, \dots, v_{k_{|V(k)|-1}}, v_{k_{|V(k)|}}, v_{k_1})\}$. We delete all arcs A_c determined in (2) from the mixed graph $G = (V, A, E)$ and apply Algorithm 1 to the obtained circuit-free mixed graph $G^0 = (V, A \setminus A_c, E)$. As a result, we obtain the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G^0 = (V, A \setminus A_c, E)$, which is equivalent to finding an optimal coloring $c(G^0)$ of the mixed graph $G^0 = (V, A \setminus A_c, E)$. Thus, the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$ is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.

It is easy to see that Algorithm 1 described in the proof of Theorem 6 shows that for any colorable mixed graph $G = (V, A, E)$, one can construct the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$, which is equivalent to finding an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$, and all jobs in the set J are simple. To illustrate Theorem 6, we consider Example 4 of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$ depicted in Figure 3.

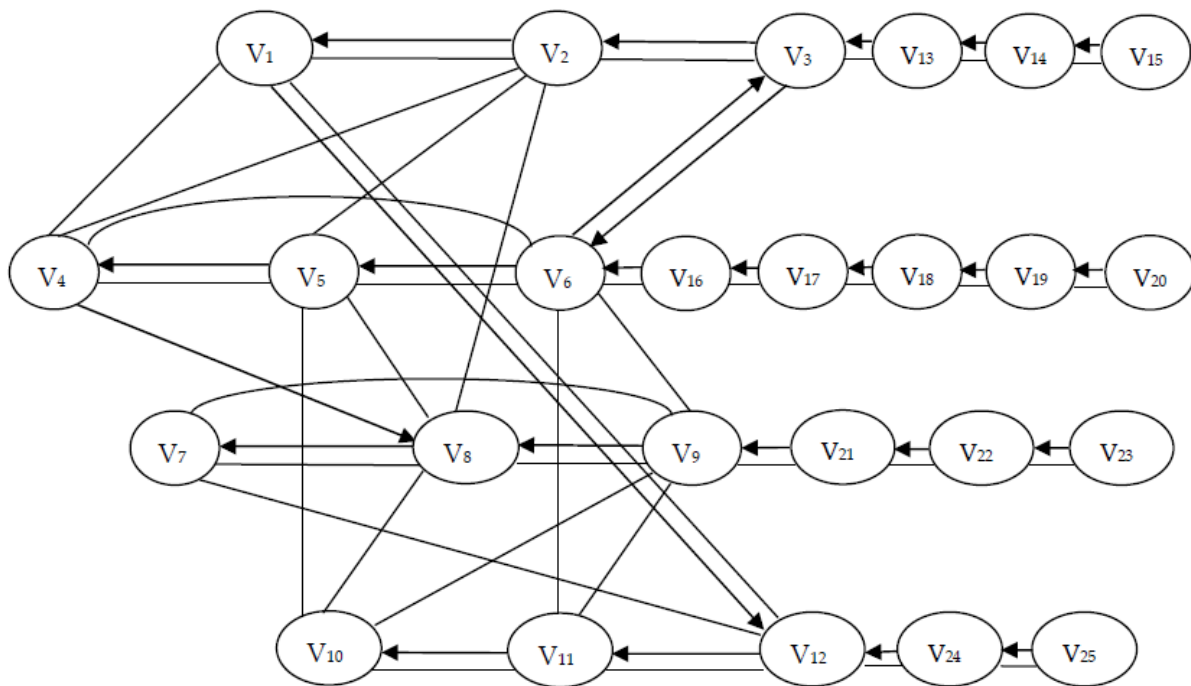


Figure 3. Mixed graph $G = (V, A, E)$ determining Example 4 of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$, optimal mixed graph coloring $c(G)$ being equivalent to Example 4.

Example 4. Let ten dedicated machines $\{M_1, M_2, \dots, M_{10}\} = M$ have to process four jobs $\{J_1, J_2, J_3, J_4\} = J$ with release times given as follows: $r_1 = 3, r_2 = 5, r_3 = 3, r_4 = 2$.

The job $J_1 \in J$ includes the set $V^{(1)} = \{v_1, v_2, v_3\}$ of the linearly ordered operations: (v_3, v_2, v_1) . In Figure 3, the job $J_1 \in J$ is represented as a union of the path (v_3, v_2, v_1) in the digraph (V, A, \emptyset) and the chain (v_1, v_2, v_3) in the graph (V, \emptyset, E) . The job $J_2 \in J$ includes the set $V^{(2)} = \{v_4, v_5, v_6\}$ of the linearly ordered operations: (v_6, v_5, v_4) . The job $J_2 \in J$ is represented as a union of the path (v_6, v_5, v_4) in the digraph (V, A, \emptyset) and the chain (v_4, v_5, v_6) in the graph (V, \emptyset, E) . The job $J_3 \in J$ includes the set $V^{(3)} = \{v_7, v_8, v_9\}$ of the linearly ordered operations: (v_9, v_8, v_7) . The job $J_3 \in J$ is represented as a union of the path (v_9, v_8, v_7) in the digraph (V, A, \emptyset) and the chain (v_7, v_8, v_9) in the graph (V, \emptyset, E) . The job $J_4 \in J$ includes the set $V^{(4)} = \{v_{10}, v_{11}, v_{12}\}$ of the linearly ordered operations: (v_{12}, v_{11}, v_{10}) . The job 4 is represented as a union of the path (v_{12}, v_{11}, v_{10}) and the chain (v_{10}, v_{11}, v_{12}) in the graph (V, \emptyset, E) .

The machine M_1 processes the operations $V_1 = \{v_1, v_4\}$. The forbiddance to process operations from the set V_1 simultaneously is determined by the edge $[v_1, v_4]$ in the graph (V, \emptyset, E) . The machine M_2 processes the operations $V_2 = \{v_2, v_5\}$. The forbiddance to process operations from the set V_2 simultaneously is determined by the edge $[v_2, v_5]$. The machine M_3 processes the operations $V_3 = \{v_2, v_4\}$. The forbiddance to process a pair of operations from the set V_3 simultaneously is determined by the edge $[v_2, v_4]$. The machine M_4 processes the operations $V_4 = \{v_5, v_8, v_{10}\}$. The forbiddance to process operations from the set V_4 simultaneously is determined by the clique $\{v_5, v_8, v_{10}\}$ in the graph (V, \emptyset, E) . The machine M_5 processes the operations $V_5 = \{v_6, v_9, v_{11}\}$. The forbiddance to process a pair of operations from the set V_5 simultaneously is determined by the clique $\{v_6, v_9, v_{11}\}$. The machine M_6 processes the operations $V_6 = \{v_7, v_9\}$. The forbiddance to process operations from the set V_6 simultaneously is determined by the edge $[v_7, v_9]$. The machine M_7 processes the operations $V_7 = \{v_2, v_8\}$. The forbiddance to process operations from the set V_7 simultaneously is determined by the edge $[v_2, v_8]$. Machine M_8 processes the operations $V_8 = \{v_9, v_{10}\}$. The forbiddance to process operations from the set V_8 simultaneously is determined by the edge $[v_9, v_{10}]$. The machine M_9 processes the operations $V_9 = \{v_7, v_{12}\}$.

The forbiddance to process operations from the set V_9 simultaneously is determined by the edge $[v_7, v_{12}]$. The machine M_{10} processes one operation $V_{10} = \{v_3\}$.

The precedence constraints between operations belonging to different jobs are given as follows: $v_3 \mapsto v_6; v_6 \mapsto v_3; v_4 \mapsto v_8; v_1 \mapsto v_{12}$. In Figure 3, these precedence constraints are presented by four arcs $(v_3, v_6), (v_6, v_3), (v_4, v_8), (v_1, v_{12})$ and one edge $[v_1, v_{12}]$ in the mixed graph $G = (V, A, E)$. The release time $r_1 = 3$ of the job $J_1 \in J$ is presented as a union of the path $(v_{15}, v_{14}, v_{13}, v_3)$ in the digraph (V, A, \emptyset) and the chain $(v_{15}, v_{14}, v_{13}, v_3)$ in the graph (V, \emptyset, E) . The release time $r_2 = 5$ of the job $J_2 \in J$ is presented as a union of the path $(v_{20}, v_{19}, v_{18}, v_{17}, v_{16}, v_6)$ in the digraph (V, A, \emptyset) and the chain $(v_{20}, v_{19}, v_{18}, v_{17}, v_{16}, v_6)$ in the graph (V, \emptyset, E) . The release time $r_3 = 3$ of the job $J_3 \in J$ is presented as a union of the path $(v_{23}, v_{22}, v_{21}, v_9)$ in the digraph (V, A, \emptyset) and the chain $(v_{23}, v_{22}, v_{21}, v_9)$ in the graph (V, \emptyset, E) . The release time $r_4 = 2$ of the job $J_4 \in J$ is presented as a union of the path (v_{25}, v_{24}, v_{12}) in the digraph (V, A, \emptyset) and the chain (v_{25}, v_{24}, v_{12}) in the graph (V, \emptyset, E) . The digraph (V, A, \emptyset) has one circuit (v_3, v_6, v_3) , which has no adjacent vertices in the graph (V, \emptyset, E) . Since the digraph (V, A, \emptyset) has no circuit containing adjacent vertices in the graph (V, \emptyset, E) , there exists a feasible schedule for Example 4 of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ (due to Theorem 4).

Due to Theorem 6, an optimal schedule for Example 4 is determined by the following optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ depicted in Figure 3: $c(v_1) = 8, c(v_2) = 7, c(v_3) = 6, c(v_4) = 9, c(v_5) = 8, c(v_6) = 6, c(v_7) = 11, c(v_8) = 10, c(v_9) = 4, c(v_{10}) = 11, c(v_{11}) = 10, c(v_{12}) = 9, c(v_{13}) = 3, c(v_{14}) = 2, c(v_{15}) = 1, c(v_{16}) = 5, c(v_{17}) = 4, c(v_{18}) = 3, c(v_{19}) = 2, c(v_{20}) = 1, c(v_{21}) = 3, c(v_{22}) = 2, c(v_{23}) = 1, c(v_{24}) = 2, c(v_{25}) = 1$. The coloring $c(G)$ with $\chi(G) = 11$ is optimal since the mixed graph $G = (V, A, E)$ includes the path $(v_{20}, v_{19}, v_{18}, v_{17}, v_{16}, v_6, v_3, v_2, v_1, v_{12}, v_{11}, v_{10})$ of the length 11. Thus, the non-strict inequality $\chi(G) \geq 11$ must hold.

3.2. Finding a Makespan Optimal Schedule with Integer Release Times Reduces to Finding a Schedule with a Smallest Maximal Lateness for Integer Due Dates

We next prove the equivalence of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ with integer release times $r_i \geq 0, J_i \in J$, and the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ to minimize the maximal lateness $L_{max} = \max\{C_i - d_i : J_i \in J\}$ with integer due dates $d_i \geq 0, J_i \in J$. To illustrate the proof of Theorem 7, we use Examples 4 of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ depicted in Figure 3.

Theorem 7. Any solvable general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ with integer release times $r_i \geq 0$ is equivalent to a general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ with integer due dates $d_i \geq 0$, and vice versa.

Proof. The general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ under consideration is solvable. Thus, due to Theorem 6, the mixed graph $G = (V, A, E)$ is colorable. Hence, due to Theorem 1, the directed subgraph (V, A, \emptyset) of the mixed graph $G = (V, A, E)$ has no circuit containing adjacent vertices in the graph (V, \emptyset, E) . Given the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$, we next construct the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ with integer due dates. In the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$, the same set of the jobs $J = \{J_1, J_2, \dots, J_{|J|}\}$ enter the processing system simultaneously at time $r = 0$. The set $J = \{J_1, J_2, \dots, J_{|J|}\}$ of the jobs must be processed on the set $M = \{M_1, M_2, \dots, M_{|M|}\}$ of the dedicated machines. We assume that in the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$, every machine $M_i \in M$ has to process the same subset $V_i = \{v_{i_1}, v_{i_2}, \dots, v_{i_{|V_i|}}\} \subseteq V$ of the jobs similarly as in the considered general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$.

For every job $J_i \in J$, which must be processed in the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$, the integer due date $d_i = \max\{r_j : J_j \in J\} - r_i$ is given along with the linear order $(v_{k_{|V^{(k)}|}}, v_{k_{|V^{(k)}|-1}}, \dots, v_{k_2}, v_{k_1})$ for processing all operations $V^{(k)} = \left\{ v_{k_1}, v_{k_2}, \dots, v_{k_{|V^{(k)}|}} \right\}$. The order $(v_{k_{|V^{(k)}|}}, v_{k_{|V^{(k)}|-1}}, \dots, v_{k_2}, v_{k_1})$ is opposite to the linearly order $(v_{k_1}, v_{k_2}, \dots, v_{k_{|V^{(k)}|}})$ determined in the considered general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ for processing the same set $V^{(k)}$ of the operations.

Further, if the precedence constraints $v_i \mapsto v_j$ and $v_p \rightarrow v_q$ are given between operations belonging to different jobs in the considered general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$, then the inverse precedence constraints $v_j \mapsto v_i$ and $v_q \rightarrow v_p$ are determined in the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$.

We demonstrate the construction of the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ for Example 4 of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ depicted in Figure 3. Given Examples 4 of the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$, we construct the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ on the mixed graph $G' = (V', A', E')$ depicted in Figure 2, provided that due dates $d_i \geq 0$ for the jobs $\{J_1, J_2, J_3, J_4\} = J$ are determined as follows: $d_1 = \max\{r_j : J_j \in J\} - r_1 = 5 - 3 = 2$, $d_2 = 5 - r_2 = 5 - 5 = 0$, $d_3 = 5 - r_3 = 5 - 3 = 2$, $d_4 = 5 - r_4 = 5 - 2 = 3$.

Due to Theorem 6, an optimal schedule for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ is determined by the optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. It is easy to convince yourself that the part of the coloring $c(G)$ of the mixed graph $G' = (V', A', E')$, where $V = V' \cup \left\{ \bigcup_{i=1}^n \left\{ v_{r_i^0}, \dots, v_{r_{i-1}^0} \right\} \right\} =: V' \cup V^0$, determine an optimal schedule for the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$, where the optimal maximal lateness is calculated as follows: $L_{max} = \chi(G) - \max\{r_i : J_i \in J\}$. Thus, the following claim is proved: *any solvable general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$ with integer release times $r_i \geq 0$ is equivalent to a general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ with integer due dates $d_i = \max\{r_j : J_j \in J\} - r_i$.*

For the constructed example of the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ on the mixed graph $G' = (V', A', E')$ depicted in Figure 2, the following optimal maximal lateness is obtained: $L_{max} = \chi(G) - \max\{r_i : J_i \in J\} = 11 - 5 = 6$.

Since the arguments presented above are reversible, one can similarly prove the following opposite claim: *any solvable general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ with integer due dates $d_i \geq 0$ is equivalent to the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the specified mixed graph $G = (V, A, E)$ with integer release times $r_i \geq 0$. Thus, Theorem 7 is proved. \square*

Theorems 6 and 7 directly imply the following corollary.

Corollary 1. *Any solvable general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ may be represented as finding an optimal coloring $c(G^*)$ of the specified mixed graph $G^* = (V^*, A^*, E^*)$.*

In Figure 4, the mixed graph $G^* = (V^*, A^*, E^*)$ is presented for the example of the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ constructed in the proof of Theorem 7. It is easy to convince yourself that an optimal schedule for the constructed general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ is determined as an optimal coloring $c(G^*)$ of the mixed graph $G^* = (V^*, A^*, E^*)$ depicted in Figure 4.

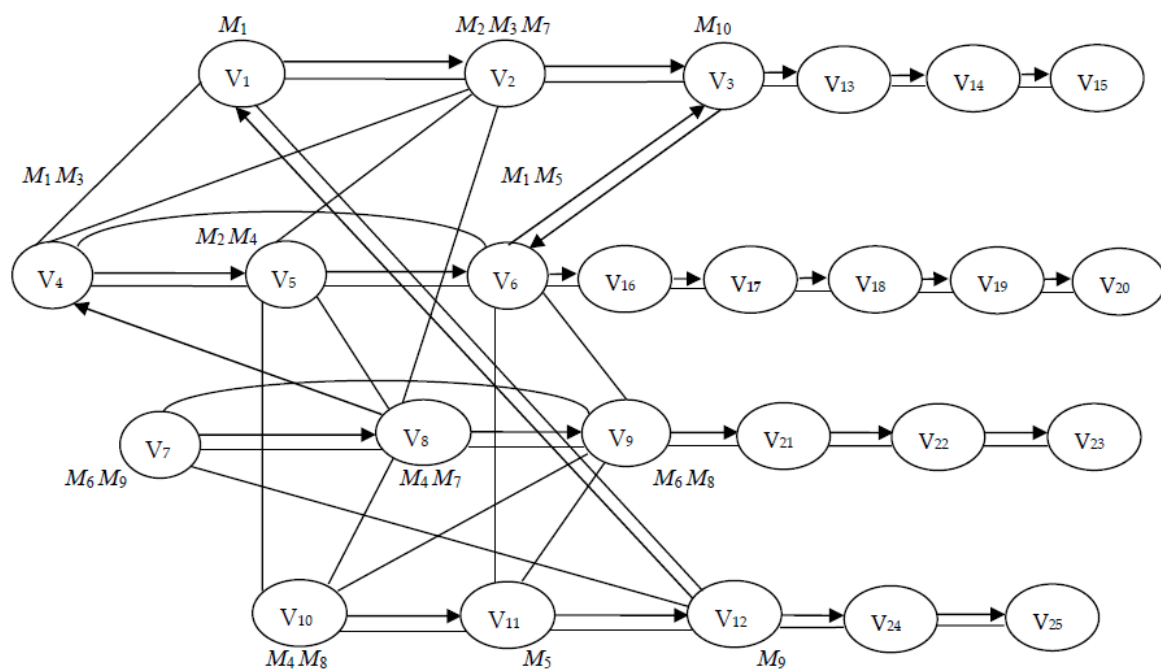


Figure 4. Mixed graph $G^* = (V^*, A^*, E^*)$, whose optimal coloring $c(G)$ determines an optimal schedule for the problem $GcMPT|t_i = 1, [d_i]|L_{max}$ constructed in the proof of Theorem 7.

Compare the mixed graph $G^* = (V^*, A^*, E^*)$ depicted in Figure 4 for the general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{max}$ and mixed graph $G = (V, A, E)$ depicted in Figure 3 for the equivalent general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$.

3.3. Optimal Mixed Graph Colorings and Equivalent Shop-Scheduling Problems

The results surveyed in Section 2 and proved in Sections 3.1 and 3.2 are summarized in Figure 5, where the relation $A \leftrightarrow B$ indicates the equivalence of the scheduling problem A presented in the left-hand part of the table in the oval and the problem B of optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ presented in the right-hand part in the corresponding oval.

Further, the arrow \rightarrow indicates that the problem A may be represented as the problem B , provided that the relation $A \rightarrow B$ holds. The mandatory properties (restrictions) of the mixed graph $G = (V, A, E)$, whose colorings are equivalent to the scheduling problem A , are described (if any) under the corresponding mixed graph $G = (V, A, E)$ in the oval.

Theorems and Corollary 1 on the equivalence of the problems A and B are indicated under the correspondent arrows in Figure 5.

In Figure 5, a pair of arrows \leftrightarrow with common right directions shows that there is a unique problem of optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$, which is equivalent to the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$. On the other hand, there are many general shop scheduling problems $GcMPT|t_i = 1, [r_i]|C_{max}$ on the mixed graph $G = (V, A, E)$, which are equivalent to the same problem of optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.

Further, there are many general shop scheduling problems $GcMPT|t_i = 1|C_{max}$ on the mixed graph $G = (V, A, E)$, which are equivalent to the same problem of optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$.

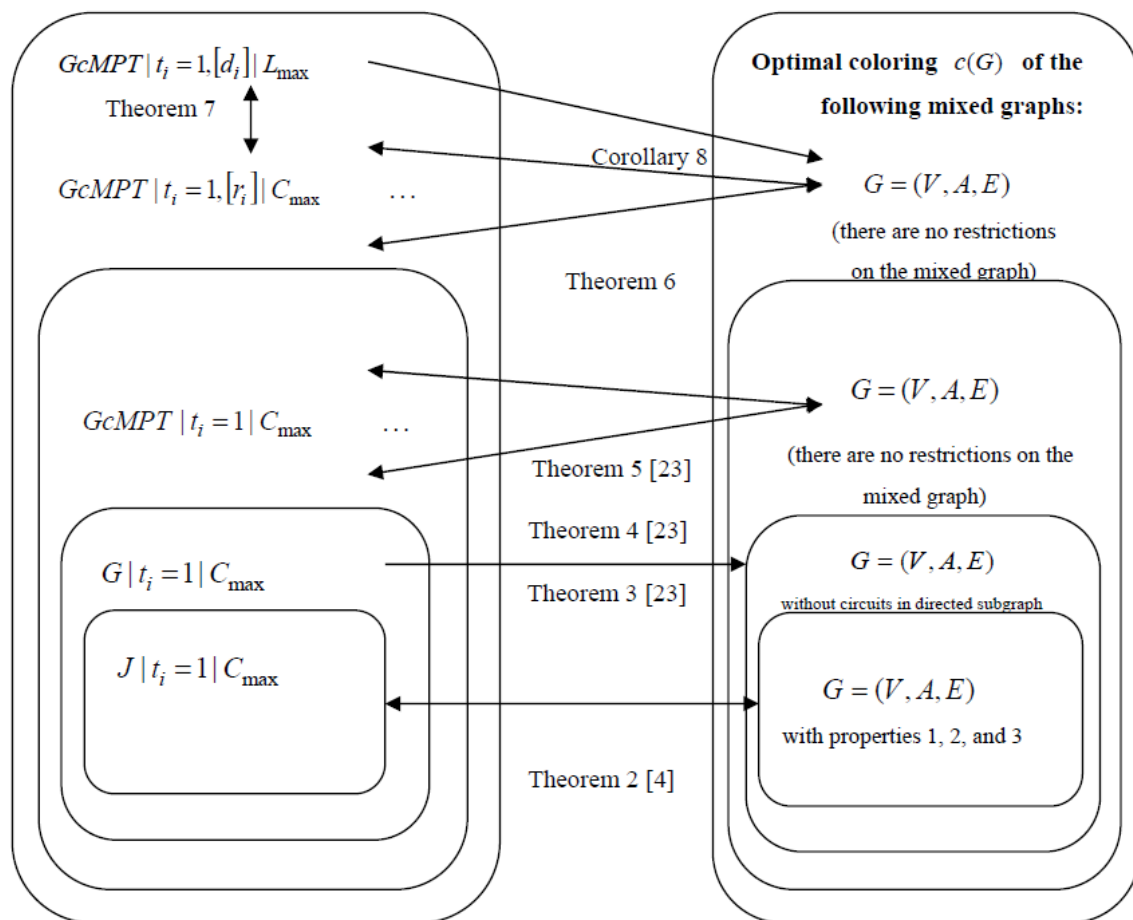


Figure 5. Optimal colorings $c(G)$ of the mixed graph $G = (V, A, E)$ and equivalent scheduling problems $\alpha|t_i = 1|\gamma$ on the mixed graph $G = (V, A, E)$, where $\alpha \in \{J, G, GcMPT\}$ and $\gamma \in \{C_{max}, L_{max}\}$ [4,23].

4. Semi-Active Schedules and Minimal Colorings of the Mixed Graphs

Obviously, there exist infinitely many feasible colorings $c(G)$ (and infinitely many feasible schedules, respectively) for any colorable mixed graph $G = (V, A, E)$ (for any solvable shop-scheduling problem $\alpha|t_i = 1|\gamma$).

In order to restrict a set of feasible schedules for a shop-scheduling problem $\alpha|\beta|\gamma$, which have to be tested in order to find an optimal schedule for a fixed regular objective function γ [16], a finite set of the semi-active schedules are sufficient to test, since there exists an optimal semi-active schedule for each concrete shop-scheduling problem $\alpha|\beta|\gamma$ with any fixed regular objective function γ [27,28].

Definition 2 ([27,28]). A schedule is called semi-active if no task (operation) can be processed earlier without violating a given constraint or changing the task (operation) processing order in the newly constructed schedule.

Remark that any coloring $c(G)$ of the mixed graph $G = (V, A, E)$ uniquely determines a strict order on the colors $c(v_j)$ used for all vertices of the set V . Due to this, one can define a minimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ as follows.

Definition 3. A coloring $c(G)$ of the mixed graph $G = (V, A, E)$ is called minimal if no color $c(v_i)$ can be decreased without changing the increasing order of colors $c(v_j)$ of the vertices in the set $V \setminus \{v_i\}$ in the newly constructed coloring $c'(G)$ of the mixed graph $G = (V, A, E)$.

Based on the proof of Theorem 6 and using Definitions 2 and 3, it is easy to convince yourself that each semi-active schedule existing for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$ uniquely determines a minimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$. On the other hand, each minimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$ uniquely determines a semi-active schedule existing for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$. Thus, the following theorem is proved.

Theorem 8. *There exists a one-to-one correspondence between all minimal colorings $c(G)$ of the mixed graph $G = (V, A, E)$ and all semi-active schedules existing for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$.*

Due to Theorems 6 and 8, in order to find an optimal schedule for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$, it is sufficient to test minimal colorings $c(G)$ of the mixed graph $G = (V, A, E)$. On the other hand, in order to find an optimal coloring $c(G)$ of the mixed graph $G = (V, A, E)$, it is sufficient to test semi-active schedules existing for the general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$.

5. Discussion

In Section 3.1, we defined a general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ of finding a schedule with the minimum length for processing multiprocessor tasks with unit-time durations and integer release times r_i provided that two types of the precedence constraints may be given on the set of the multiprocessor tasks. Contrary to a classical shop-scheduling problem, several machines are required to process a multiprocessor task in the problem $GcMPT|t_i = 1, [r_i]|C_{\max}$. Furthermore, it is required that specified subsets of the multiprocessor tasks must be processed simultaneously.

We proved Theorem 6 showing that an optimal coloring $c(G)$ of any colorable mixed graph $G = (V, A, E)$ is equivalent to the general shop scheduling problem $GcMPT|p_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$. Theorem 6 implies that most analytical and algorithmic results proven for optimal colorings of the mixed graph, have analogous results, which are valid for the newly defined scheduling problem $GcMPT|p_i = 1, [r_i]|C_{\max}$, and vice versa. In particular, some results that have been proven in [10,11,27–34] for the scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ and for its special cases may be interpreted as analogous results for optimal colorings $c(G)$ of the corresponding mixed graphs $G = (V, A, E)$. Similarly, some results that have been proven in [3–5,8,9,17,18,22–24] for optimal mixed graph colorings $c(G)$ may be interpreted as analogous results for general shop scheduling problems $GcMPT|t_i = 1, [r_i]|C_{\max}$. In other words, there are articles [3–5,8–11,17,18,22–24,27–34] studied both of these problems without indicating that they are actually the same problems.

Theorem 6 may be considered as a generalization of Theorem 2 proven in [4] and Theorems 3 and 4 proven in [23].

In Section 3.2, we defined a new general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{\max}$ to minimize the maximal lateness $L_{\max} = \max\{C_i - d_i : J_i \in J\}$ with integer due dates $d_i \geq 0$, two types of the precedence constraints, specified subset of tasks, which must be processed simultaneously, and unit-time durations of the multiprocessor tasks. We proved Theorem 7 showing that any solvable general shop scheduling problem $GcMPT|t_i = 1, [r_i]|C_{\max}$ on the mixed graph $G = (V, A, E)$ with integer release times is equivalent to a general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{\max}$ with integer due dates, and vice versa. Corollary 1, Theorem 6, and Theorem 7 imply that any solvable general shop scheduling problem $GcMPT|t_i = 1, [d_i]|L_{\max}$ reduces to finding an optimal coloring $c(G)$ of the specified mixed graph $G = (V, A, E)$.

5.1. New Approaches to Shop-Scheduling Problems and Mixed Graph Colorings

The results presented in Sections 2–4 may be interpreted as a new scheduling approach to the mixed graph coloring problems and as a new mixed graph coloring approach to the specified scheduling problems. For applying the mixed graph coloring approach, a scheduling problem must have equal durations of all tasks, integer release times for the makespan criterion (and integer due dates for the maximal lateness criterion). Furthermore, preemptions of the given task are forbidden. There is no restriction on mixed graphs for applying the scheduling approach to the mixed graph coloring problems.

It should be noted that many terms used in scheduling theory (such as schedule, job, machine, processor, operation, task, processing time, release times, due dates, maximal lateness, makespan, etc.) may be considered as usual terms used in mixed graph colorings. The terminology used in modern scheduling theory is more complicated since there are many scheduling applications in the different fields of real world. Based on the constructive proofs of Theorems 6 and 7, it is possible to describe many results either using only graph terminology or using only scheduling terminology.

The mixed graph coloring approach may be promising in the mass production, which presupposes scheduling problems with equal processing times of the jobs to minimize either makespan C_{\max} or maximal lateness L_{\max} .

5.2. Future Research Directions

A future research direction may be connected with the usage of the mixed graph coloring approach to develop efficient mechanisms for scheduling cloud computations, where there is not enough information on the durations of the tasks to be processed on virtual machines. Both considered objective functions (the makespan and maximal lateness) are important for the optimization of the cloud computations.

Another promising research direction is connected with the application of the mixed graph coloring approach to scheduling personal jobs in the time-management framework, where a user needs to have a break (preemption) in her (his) activity after equal time intervals. In other words, one can assume that the scheduling problems arising in the time-management have unit-time durations of the tasks to be processed within a day. In future research, we plan to investigate other classes of shop-scheduling problems, which are equivalent to optimal colorings of the specified mixed graphs.

Author Contributions: Y.N.S. proposed the methodology and defined the new scheduling problems; Y.N.S. and E.I.M. jointly proved theoretical results; E.I.M. designed the algorithm; Y.N.S. and E.I.M. jointly prepared examples and analyzed the obtained data; E.I.M. wrote the review; Y.N.S. wrote the manuscript; Y.N.S. and E.I.M. have read and approved the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: The research by the first author was funded by Belarusian Republican Foundation for Fundamental Research BRFFR-NSFC, grant number F22KI-005.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Duguay, C.R.; Landry, S.; Pasin, F. From mass production to flexible/agile production. *Int. J. Oper. Prod. Manag.* **1997**, *17*, 1183–1195. [[CrossRef](#)]
2. Wan, L.; Mei, J.; Du, J. Two-agent scheduling of unit processing time jobs to minimize total weighted completion time and total weighted number of tardy jobs. *Eur. J. Oper. Res.* **2021**, *290*, 26–35. [[CrossRef](#)]
3. Sotskov, Y.N.; Tanaev, V.S. A chromatic polynomial of a mixed graph. *Vestsi Akademii Navuk BSSR Seryya Fizika-Matematychnykh Navuk.* **1976**, *6*, 20–23. (In Russian)

4. Sotskov, Y.N.; Dolgui, A.; Werner, F. Mixed graph coloring for unit-time job-shop scheduling. *Int. J. Math. Algorithms* **2001**, *2*, 289–323.
5. Sotskov, Y.N.; Tanaev, V.S.; Werner, F. Scheduling problems and mixed graph colorings. *Optimization* **2002**, *51*, 597–624. [[CrossRef](#)]
6. Karp, R.M. Reducibility among combinatorial problems. In *Complexity of Computer Computations*; Miller, R.E., Thatcher, J.W., Eds.; Plenum Press: New York, NY, USA, 1972; pp. 85–103.
7. Al-Anzi, F.S.; Sotskov, Y.N.; Allahverdi, A.; Andreev, G.V. Using mixed graph coloring to minimize total completion time in job shop scheduling. *Appl. Math. Comput.* **2006**, *182*, 1137–1148. [[CrossRef](#)]
8. Kouider, A.; Ait Haddadne, H.; Ourari, S.; Oulamara, A. Mixed graph coloring for unit-time scheduling. *Int. J. Prod. Res.* **2017**, *55*, 1720–1729. [[CrossRef](#)]
9. Kouider, A.; Ait Haddadne, H.; 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**, *4967*, 1001–1008.
10. Lenstra, J.K.; Rinnooy Kan, A.H.G. Computational complexity of discrete optimization problems. *Ann. Discret. Math.* **1979**, *4*, 121–140.
11. Gonzalez, T. Unit execution time shop problems. *Math. Oper. Res.* **1982**, *7*, 57–66. [[CrossRef](#)]
12. Sotskov, Y.N. Complexity of optimal scheduling problems with three jobs. *Cybernetics* **1990**, *26*, 686–692. [[CrossRef](#)]
13. Sotskov, Y.N. The complexity of shop-scheduling problems with two or three jobs. *Eur. J. Oper. Res.* **1991**, *53*, 326–336. [[CrossRef](#)]
14. Sotskov, Y.N.; Shakhlevich, N.V. NP-hardness of shop-scheduling problems with three jobs. *Discret. Appl. Math.* **1995**, *59*, 237–266. [[CrossRef](#)]
15. Kravchenko, S.A.; Sotskov, Y.N. Optimal makespan schedule for three jobs on two machines. *ZOR Z. Oper. Res.* **1996**, *43*, 233–238. [[CrossRef](#)]
16. Brucker, P.; Kravchenko, S.A.; Sotskov, Y.N. On the complexity of two machine job-shop scheduling with regular objective functions. *Oper.-Res.-Spektrum* **1997**, *19*, 5–10. [[CrossRef](#)]
17. Damaschke, P. Parameterized mixed graph coloring. *J. Comb. Optim.* **2019**, *38*, 326–374. [[CrossRef](#)]
18. Hansen, P.; Kuplinsky, J.; De Werra, D. Mixed graph colorings. *Math. Meth. Oper. Res.* **1997**, *45*, 145–160. [[CrossRef](#)]
19. Kruger, K.; Sotskov, Y.N.; Werner, F. Heuristic for generalized shop scheduling problems based on decomposition. *Int. J. Prod. Res.* **1998**, *36*, 3013–3033. [[CrossRef](#)]
20. Sotskov, Y.N. Software for production scheduling based on the mixed [multi]graph approach. *Comput. Contr. Eng. J.* **1996**, *7*, 240–246. [[CrossRef](#)]
21. Sotskov, Y.N. Mixed multigraph approach to scheduling jobs on machines of different types. *Optimization* **1997**, *42*, 245–280. [[CrossRef](#)]
22. De Werra, D. On a multiconstrained model for chromatic scheduling. *Discret. Appl. Math.* **1999**, *94*, 171–180. [[CrossRef](#)]
23. Sotskov, Y.N. Mixed graph coloring as scheduling multi-processor tasks with equal processing times. *J. Belarusian State Univ. Math. Inform.* **2021**, *2*, 67–81.
24. Sotskov, Y.N. Mixed graph colorings: A historical review. *Mathematics* **2020**, *8*, 385. [[CrossRef](#)]
25. Harary, F. *Graph Theory*; Addison-Wesley: Reading, MA, USA, 1969.
26. Thulasiraman, K.; Swamy, M.N.S. *Graphs: Theory and Algorithms*; John Wiley & Sons, Inc.: Toronto, ON, Canada, 1992.
27. Tanaev, V.S.; Sotskov, Y.N.; Strusevich, V.A. *Scheduling Theory: Multi-Stage Systems*; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1994.
28. Brucker, P. *Scheduling Algorithms*; Springer: Berlin, Germany, 1995.
29. Graham, R.L.; Lawler, E.L.; Lenstra, J.K.; Rinnooy-Kan, A.H.G. Optimization and approximation in deterministic sequencing and scheduling. *Ann. Discret. Appl. Math.* **1979**, *5*, 287–326.
30. Baptiste, P. A note on scheduling multiprocessor tasks with identical processing times. *Comput. Oper. Res.* **2003**, *30*, 2071–2078. [[CrossRef](#)]
31. Zinder, Y.; Dob, V.H.; Oğuz, C. Computational complexity of some scheduling problems with multiprocessor tasks. *Discret. Optimization* **2005**, *2*, 391–408. [[CrossRef](#)]
32. Kis, T. Scheduling multiprocessor UET tasks of two sizes. *Theor. Comput. Sci.* **2009**, *410*, 4864–4873. [[CrossRef](#)]
33. Giaro, K.; Kubale, M.; Obszarski, P. A graph coloring approach to scheduling of multiprocessor tasks on dedicated machines with availability constraints. *Discret. Appl. Math.* **2009**, *157*, 3625–3630. [[CrossRef](#)]
34. Drozdowski, M. Scheduling multiprocessor—an overview. *Eur. J. Oper. Res.* **1996**, *94*, 215–230. [[CrossRef](#)]
35. Błazewicz, J.; Olmo, P.D.; Drozdowski, M.; Mazczka, P. Scheduling multiprocessor tasks on parallel processors with limited availability. *Eur. J. Oper. Res.* **2003**, *149*, 377–389. [[CrossRef](#)]
36. Chou, F.D. Particle swarm optimization with cocktail decoding method for hybrid flow shop scheduling problems with multiprocessor tasks. *Int. J. Prod. Econ.* **2013**, *141*, 137–145. [[CrossRef](#)]
37. 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**, *44*, 987–1002. [[CrossRef](#)]
38. Brucker, P.; Kramer, A. Shop scheduling problems with multiprocessor tasks on dedicated processors. *Ann. Oper. Res.* **1995**, *57*, 13–27. [[CrossRef](#)]
39. Brucker, P.; Kramer, A. Polynomial algorithms for resource-constrained and multiprocessor task scheduling problems. *Eur. J. Oper. Res.* **1996**, *90*, 214–226. [[CrossRef](#)]

-
40. Hoogeveen, J.A.; van de Velde, S.L.; Veltman, B. Complexity of scheduling multiprocessor tasks with prespecified processor allocations. *Discret. Appl. Math.* **1994**, *55*, 259–272. [[CrossRef](#)]
 41. Hoogeveen, J.A.; Lenstra, J.K.; Veltman, B. Preemptive scheduling in a two-stage multiprocessor flow shop is NP-hard. *Eur. J. Oper. Res.* **1996**, *89*, 172–175. [[CrossRef](#)]