Optimal Scheduling Works for Two Employees with Ordered Criteria

NATALJA M. MATSVEICHUK[1,*], YURI N. SOTSKOV[2], LIANGLIANG SUN[3]
[1]Department of Automated Systems of Production Control,
Belarusian State Agrarian Technical University,
99 Nezavisimosti av., Minsk 220012,
BELARUS

[2]United Institute of Informatics Problems, National Academy of Sciences of Belarus,
6 Surganov Street, Minsk 220012,
BELARUS

[3]School of Control Engineering, Northeastern University,
143 Taishan Road, Qinhuangdao, 066004,
CHINA

[*]*Corresponding Author*

Abstract: - The number of users of time management in the world is increasing due to the need for remote work (at home), study, teaching, and service. This requires the improvement of time-management models, methods, and the development of new optimization algorithms and software tools that will take into account the characteristics and needs of new time-management users. This article presents the necessary and sufficient conditions for schedule optimality, algorithms, and results of computational experiments conducted on the optimal selection and planning of interrelated jobs for two performers (a supervisor and a subordinate). Such an optimization problem arises in time management for scheduling the performance of the selected jobs in conditions of the uncertainty of the selected jobs to be scheduled and then executed. We develop a scheduling algorithm and present computational results for selecting and scheduling interrelated jobs for two employees for minimizing the schedule length as the main criterion and minimizing the sum of the weighted completion times of the jobs as the second criterion.

Key-Words: - time-management; schedule; interval uncertainty; makespan; total completion time; computational experiment.

Received: May 14, 2024. Revised: September 29, 2024. Accepted: November 2, 2024. Published: November 22, 2024.

1 Introduction

Optimal planning of work performance during work and personal time is a complex process that requires time resources and human intellectual abilities, [1], [2], [3]. Time-management skills can compensate for work role overload [4], and effective planning and scheduling can reduce wasted time [5], [6]. As a result, the use of time management allows an employee to save up to 50% of the working time on the completion of planned tasks, spending no more than 10% of the time on analysis and planning, [7].

The importance of time management is increasing in the modern world when more and more people are switching to remote work.

Organizing work time requires certain skills and personal experience, so it is preferable to automate the process of selecting and planning jobs using a computer, laptop, or smartphone, [1]. Since modern people (especially young people) interact a lot with smartphones, they will be able to easily control their planned work. On the other hand, there is a negative relationship between mobile phone addiction and time management, [8]. Automatic work scheduling can help to overcome this gap.

The time-management literature focuses on the problems of managing the working and personal time of an individual employee. Various techniques and procedures have been developed to determine the optimal sequence of the jobs planned by an employee, [1].

This paper examines the problems of constructing optimal schedules for two employees with close and related works and positions, e.g., for a supervisor and a subordinate.

The main objectives of time management are to select the most important jobs to construct optimal schedules for both employees during their working hours. We show how scheduling theory can be used for optimal time management of two employees.

2 Problem Setting and Preliminaries

The considered scheduling problem for time management has the following characteristics.

A key feature is the uncertainty of the duration of jobs performed by a human. Indeed, for a real duration of the selected job, only lower and upper bounds can be determined before scheduling, which will contain a factual duration of the job. The exact (factual) duration of each job is not known until the job has been completed.

Job interruptions should be avoided, if possible, as they result in a direct loss of time and the additional time needed to prepare the interrupted job, [9].

We assume that the planned jobs have different weights that determine different levels of importance of the jobs to be fulfilled in the planning horizon.

The selected jobs for two employees can be performed by one of them or by both of them in a fixed order (fixed sequence). In the latter case, it is possible that the superior starts the common job and then the subordinate completes it (e.g., the superior formalizes a problem, outlines possible ways of its solution, and delegates the started job to the subordinate).

An opposite sequence of the common job is also allowed in time management, e.g., the supervisor checks the results of the common job performed by the subordinate.

The optimality of a schedule for human jobs is quite subjective, and employees may consider various quantitative and qualitative optimality criteria for the desired schedules.

The following optimality criteria can be considered in time management:

Maximization of the number of completed jobs;

Minimization of the sum of the weighted job completion times;

Minimization of the weighted number of late jobs relative to the given due dates;

Minimization of the weighted total delay of the jobs;

Minimization of the maximum delay;

Minimization of the number of scheduled jobs that are not completed by the end of the working day;

Ensuring the alternation of easy and difficult jobs in the schedule;

Preference for hard work in the morning or, conversely, at the end of the work day;

Preference for stable schedules (repeatability of similar schedules from day to day).

For example, a scheduling problem for a single employee with three criteria (total completion time, total lateness, and total earlyness) was considered in [10].

In this paper, we consider the makespan criterion (i.e., minimizing a schedule length) as the main criterion. Indeed, a role overload (i.e., insufficient time to complete work tasks) leads to a decrease in an employee's productivity [4], which affects future work. The second criterion considered is a minimization of the sum of the weighted job completion times.

Planning and scheduling problems are treated in scheduling theory, [11], [12], where effective models and algorithms for constructing optimal schedules have been developed for different numbers of machines and different classical criteria.

In the following, we use the terminology of scheduling theory [11] and the $\alpha|\beta|\gamma$ classification from [12] to denote the scheduling problems, where α specifies machine environments, β job characteristics, and γ objective functions.

Let $\Im = \{J_1, J_2,\dots, J_n\}$ denote a set of jobs that have to be processed by two workers $M = \{M_1, M_2\}$. A weight (importance) w_i of the job $J_i \in \Im$ is known before scheduling. The job J_i in the set $\Im$ has a fixed route (sequence) with one or two stages (operations) m_i.

Let the following equality holds: $\Im = \Im_1 \cup \Im_2 \cup \Im_{1,2} \cup \Im_{2,1}$, where the subset $\Im_{1,2}$ includes all jobs with the route (M_1, M_2), where $|\Im_{1,2}| = n_{1,2}$. The subset $\Im_{2,1}$ includes jobs with the opposite route (M_2, M_1), where $|\Im_{2,1}| = n_{2,1}$.

The subset $\Im_1$ (or subset $\Im_2$) includes jobs that must be processed by employee M_1 (by employee M_2).

Let the following equalities hold: $|\Im_1| = n_1$, $|\Im_2| = n_2$, $n = n_{1,2} + n_{2,1} + n_1 + n_2$.

All jobs are available for processing from the same release time $t = 0$. Preemption of any operation O_{ij} of the job $J_i \in \Im$ processed by the employee $M_j \in M$ is not allowed.

The factual duration of the operation O_{ij} is denoted by p_{ij}, where $J_i \in \Im$ and $M_j \in M$. The lower and upper bounds of the possible duration p_{ij} are denoted by a_{ij} and b_{ij}, respectively. The probability distributions of the random durations are unknown before scheduling. In the realization of a schedule, the factual value of the processing time p_{ij} can be any real number not less than the lower bound a_{ij} and not greater than the upper bound b_{ij}.

In [11], such a processing system was called a two-machine job-shop scheduling with uncertain (interval) processing times, where the possible duration p_{ij} of the operation O_{ij} belongs to the closed interval (segment) $[a_{ij}, b_{ij}]$.

Let C_i denote a moment of the completion time of the job $J_i \in \mathfrak{J}$. We will consider the following two ordered criteria: the minimization of the makespan $C_{\max} = \max\{C_i : J_i \in \mathfrak{J}\}$ and the minimization of the sum $\sum w_i C_i$ of the weighted completion times of the jobs $J_i \in \mathfrak{J}$. Using the three-field notation α|β|γ, this scheduling problem with interval operation durations is denoted as follows:

$$J2|a_{ij} \le p_{ij} \le b_{ij}, m_i \le 2|C_{\max}, \sum w_i C_i.$$

Two considered criteria are strictly ordered and the main criterion is to minimize the schedule length $C_{\max}$.

A set of all possible vectors $p = (p_{1,1}, p_{1,2}, ..., p_{n1}, p_{n2})$ of the operation durations is denoted by $T = \{p : a_{ij} \le p_{ij} \le b_{ij}, J_i \in \mathfrak{J}, M_j \in \mathrm{M}\}$. Each vector $p \in T$ of possible durations is called a scenario.

For a fixed scenario $p \in T$, the uncertain scheduling problem $J2|a_{ij} \le p_{ij} \le b_{ij}, m_i \le 2|C_{\max}$ turns out to be the deterministic scheduling problem $J2|p, m_i \le 2|C_{\max}$, which is an individual problem associated with the fixed scenario p.

As it is noted in [11], it is sufficient to search for the optimal schedule for the deterministic problem $J2|a_{ij} \le p_{ij} \le b_{ij}, m_i \le 2|C_{\max}$ among the finite set of semi-active schedules.

The schedule for the scheduling problem $\alpha|\beta|\gamma$ is called semi-active if the execution of each operation cannot be processed earlier without violating the order of operations in this schedule or another operation is processed later than in this schedule, [11].

In [13], it is proved that for any fixed scenario $p \in T$ there exists a Jackson's pair of job permutations of the form (π', π"). The permutation $\pi' = (\pi_{1,2}, \pi_1, \pi_{2,1})$ determines an optimal schedule (an optimal sequence) for processing jobs by employee M_1 and the permutation $\pi'' = (\pi_{2,1}, \pi_2, \pi_{1,2})$ determines an optimal schedule (an optimal sequence) for processing jobs by employee M_2.

Job J_i belongs to the permutation π_h, if the inclusion $J_i \in \mathfrak{J}_h$ holds. The permutation $\pi_{1,2}$ (the permutation $\pi_{2,1}$) is the same in the permutations π' and π".

The optimal order of the jobs from the set $\mathfrak{J}_1$ and the jobs from the set $\mathfrak{J}_2$ can be arbitrary [13], [14]. In the Johnson's permutation $\pi_{1,2} = (\ldots, J_k, \ldots, J_m, \ldots)$ (in the permutation $\pi_{2,1} = (\ldots, J_k, \ldots, J_m, \ldots)$, respectively), the Johnson's inequalities (see [14]) hold for all indices k and m, where $1 \le k \le n_{1,2}$, $1 \le m \le n_{1,2}$:

$$\min\{p_{k,1}, p_{m,2}\} \le \min\{p_{m,1}, p_{k,2}\},$$
$$(\min\{p_{k,2}, p_{m,1}\} \le \min\{p_{m,2}, p_{k,1}\}).$$

More general sufficient conditions for the permutation optimality have been proved in [15], [16].

3 Properties of the Uncertain Scheduling Problem

In most cases, there is no single schedule that is optimal for all possible scenarios $p \in T$. Due to this fact, the uncertain (interval) scheduling problem $J2|a_{ij} \le p_{ij} \le b_{ij}, m_i \le 2|C_{\max}$ is mathematically incorrect. In the worst case, any pair of permutations (π', π") may be the only optimal one for some scenarios $p \in T$.

Properties of the makespan-optimal schedule with interval processing times have been investigated in papers [17], [18].

Paper [17] provides sufficient conditions for a pair of job permutations (π', π") to be optimal for the deterministic problem $J2|p, m_i \le 2|C_{\max}$ with any fixed scenario $p \in T$. Such a pair of job permutations (π', π") is optimal for the uncertain scheduling problem $J2|a_{ij} \le p_{ij} \le b_{ij}, m_i \le 2|C_{\max}$.

Theorem 7 and Corollaries 3 and 4 proved in [17] determine sufficient conditions (1) – (4) for the optimality of the pair of job permutations (π', π") with an arbitrary order of jobs of the set $\mathfrak{J}_{1,2}$ in the permutation $\pi_{1,2}$ and an arbitrary order of jobs of the set $\mathfrak{J}_{2,1}$ in the permutation $\pi_{2,1}$:

$$\sum_{J_i \in \mathfrak{J}_{1,2}} b_{i1} \le \sum_{J_j \in \mathfrak{J}_2 \cup \mathfrak{J}_{2,1}} a_{j2}, \quad (1)$$

$$\sum_{J_i \in \mathfrak{J}_{1,2}} a_{i2} \ge \sum_{J_j \in \mathfrak{J}_1 \cup \mathfrak{J}_{2,1}} b_{j1}, \quad (2)$$

$$\sum_{J_i \in \mathfrak{J}_{2,1}} b_{i2} \le \sum_{J_j \in \mathfrak{J}_1 \cup \mathfrak{J}_{1,2}} a_{j1}, \quad (3)$$

$$\sum_{J_i \in \mathfrak{J}_{2,1}} a_{i1} \ge \sum_{J_j \in \mathfrak{J}_2 \cup \mathfrak{J}_{1,2}} b_{j2}. \quad (4)$$

If the conditions (1) and (2) (the condition (1)) hold, then the orders of the jobs in the sets $\mathfrak{J}_{1,2}$ and $\mathfrak{J}_{2,1}$ (the jobs in the set $\mathfrak{J}_{1,2}$) in the optimal pair of job permutations (π', π") can be arbitrary.

If the conditions (3) and (4) (the condition (3)) hold, then the orders of the jobs in the sets $\mathfrak{J}_{2,1}$ and $\mathfrak{J}_{1,2}$ (the jobs in the set $\mathfrak{J}_{2,1}$) in the optimal pair of job permutations (π', π") can be arbitrary.

Theorem 2 given in [17] determines the following necessary and sufficient conditions (a) and (b) for the existence of the permutation $\pi_{1,2}$ that is a Johnson's one for the jobs from the set $\Im_{1,2}$ for any fixed scenario $p \in T$:

a) for each pair of jobs $J_i \in \Im^1_{1,2}$ and $J_j \in \Im^1_{1,2}$ (jobs $J_i \in \Im^2_{1,2}$ and $J_j \in \Im^2_{1,2}$, respectively), either the inequality $b_{i1} \le a_{j1}$ or $b_{j1} \le a_{i1}$ holds (either inequality $b_{i2} \le a_{j2}$ or $b_{j2} \le a_{i2}$ holds, respectively);

b) the inequality $|\Im^*_{1,2}| \le 1$ holds and for the job $J_{i^*} \in \Im^*_{1,2}$, the inequalities $a_{i^*1} \ge \max\{b_{i1} \mid J_i \in \Im^1_{1,2}\}$, $a_{i^*2} \ge \max\{b_{i2} \mid J_i \in \Im^2_{1,2}\}$ hold.

The equality $\Im_{1,2} = \Im^1_{1,2} \cup \Im^2_{1,2} \cup \Im^*_{1,2}$ holds, where

$$\Im^1_{1,2} = \{J_i \in \Im_{1,2} \mid b_{i1} \le a_{i2}\},$$
$$\Im^2_{1,2} = \{J_i \in \Im_{1,2} \mid b_{i2} \le a_{i1}\},$$
$$\Im^*_{1,2} = \{J_i \in \Im_{1,2} \mid b_{i1} > a_{i2}, b_{i2} > a_{i1}\}.$$

The conditions of Theorem 2 given in [17] can be similarly reformulated for the jobs from the set $\Im_{2,1}$.

If the above sufficient conditions do not hold, one can construct the dominant set of the pairs of job permutations for the uncertain (interval) scheduling problem $J2|a_{ij} \le p_{ij} \le b_{ij}, m_i \le 2|C_{\max}$, which contains at least one optimal pair (π', π'') of job permutations for the deterministic problem $J2|p, m_i \le 2|C_{\max}$ with any fixed scenario $p \in T$.

In [18], to reduce the number of such permutations, the binary relation $A^{1,2}_{\prec}$ on the set $\Im_{1,2}$ (the binary relation $A^{2,1}_{\prec}$ on the set $\Im_{2,1}$, respectively) was constructed and the conflict sets of the jobs were identified.

Two jobs J_u and J_v are in binary relation $A^{1,2}_{\prec}$, i.e., the inclusion $(J_u, J_v) \in A^{1,2}_{\prec}$ holds, if one of the following conditions (5) and (6) holds:

$$b_{u1} \le a_{v2} \text{ and } b_{u1} \le a_{v1}, \quad (5)$$

$$b_{v2} \le a_{v1} \text{ and } b_{v2} \le a_{u2}. \quad (6)$$

The subset $\Im_x \subseteq \Im_{1,2}$ is called a conflict set of jobs if, for any job $J_y \in \Im_{1,2} \backslash \Im_x$, either the relation $(J_x, J_y) \in A^{1,2}_{\prec}$ or the relation $(J_y, J_x) \in A^{1,2}_{\prec}$ holds for each job $J_x \in \Im_x$; provided that any proper subset of the set $\Im_{1,2}$ does not have such a property.

The construction of the above binary relation requires no more than n^2 elementary operations. The set of permutations determined by this binary relation contains at least one Johnson's permutation of the jobs from the set $\Im_{1,2}$ (from the set $\Im_{2,1}$, respectively) for each fixed scenario $p \in T$, i.e., it is a dominant set of the permutations.

Let the binary relation $A^{1,2}_{\prec}$ on the set $\Im_{1,2}$ have the following form:

$$J_1 \prec \ldots \prec J_k \prec \{J_{k+1}, \ldots, J_{k+r}\} \prec J_{k+r+1} \prec \ldots \prec J_{n_{1,2}},$$

where the jobs $J_{k+1}, \ldots, J_{k+r}$ constitute a conflict set, while the remaining jobs are strictly ordered based on the dominant relations.

The sufficient conditions for verifying an optimal order for processing jobs in the conflict set were proved in [18]; see Theorems 10, 11 and 12.

If the following conditions (7) of Theorem 10 hold, then the order of the jobs in the conflict set can be arbitrary:

$$\sum_{i=1}^{k+r} b_{i,1} \le \sum_{J_i \in \Im_2 \cup \Im_{2,1}} a_{i,2} + \sum_{j=1}^{k} a_{j,2} \quad (7)$$

To check the conditions of Theorems 11 and 12, it is required to construct a permutation of the jobs of the conflict set using the algorithms developed in [18]. For the permutation of the form $(J_{k+1}, \ldots, J_{k+r})$ the sufficient conditions of the optimality in the Theorems 11 and 12 have the following form (8) and (9), respectively:

$$b_{k+s,1} \le \sum_{J_i \in \Im_2 \cup \Im_{2,1}} a_{i,2} + \sum_{j=1}^{k+s-1} \left(a_{j,2} - b_{j,1}\right), \quad (8)$$

$$s \in \{1, 2, \ldots, r\};$$

$$\sum_{i=r-s+2}^{r+1} a_{k+i,1} \ge \sum_{j=r-s+1}^{r} b_{k+j,2}, \; s \in \{1, 2, \ldots, r\}. \quad (9)$$

If there are multiple conflict sets in the job set $\Im_{1,2}$, the conditions for each such conflict set can be checked sequentially.

Various methods have been developed to select the next job when there are no sufficient conditions for an optimal job order (see survey papers [19], [20]). In our paper, we use ordering based on the second criterion, i.e., in the non-increasing order of the job weights.

4 Illustrative Example

We assume that eight jobs must be planned for two employees for a day. The upper and lower bounds of possible job durations and job weights are given in Table 1.

Table 1. Input data for the Example.

J_i	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8
a_{i1}	3	3	3	2	-	6	5	5
b_{i1}	4	4	4	3	-	7	6	6
a_{i2}	1	1	1	-	2	6	8	7
b_{i2}	3	3	3	-	3	7	9	9
w_i	3	2	4	1	3	4	3	4

Let the following equalities hold: $\Im_{1,2} = \{J_1, J_2, J_3\}$, $\Im_1 = \{J_4\}$, $\Im_2 = \{J_5\}$, $\Im_{2,1} = \{J_6, J_7, J_8\}$.

We need to determine the optimal order of the jobs for the makespan as the main criterion and the sum of the weighted job completion times as the second criterion.

Thus, we need to solve the uncertain job-shop scheduling problem $J2|a_{ij} \le p_{ij} \le b_{ij},\ m_i \le 2|C_{\max}, \sum w_iC_i$.

First, we check the conditions (1)–(4) for a single pair of job permutations that is optimal for all possible scenarios $p \in T$.

The condition (1) holds:

$$4 + 4 + 4 < 2 + 6 + 8 + 7,$$

while the condition (2) does not hold:

$$1 + 1 + 1 < 3 + 7 + 6 + 6.$$

The condition (3) does not hold:

$$7 + 9 + 9 > 2 + 3 + 3 + 3.$$

The condition (4) can be left unchecked.

Thus, the order of the jobs in the set $\Im_{1,2}$ in the optimal pair of the job permutations (π', π'') can be arbitrary. We fix an order of the jobs in the permutation $\pi_{1,2}$ with the decreasing order of their weights. We obtain that $\pi_{1,2} = (J_3, J_1, J_2)$.

Since $\Im_1 = \{J_4\}$, $\Im_2 = \{J_5\}$, we also obtain the equalities $\pi_1 = (J_4)$ and $\pi_2 = (J_5)$.

Consider the set $\Im_{2,1} = \{J_6, J_7, J_8\}$. We obtain $\Im^1_{2,1} = \{J_7, J_8\}$, $\Im^2_{2,1} = \varnothing$, $\Im^*_{2,1} = \{J_6\}$. The following inequalities $b_{7,1} = 6 > a_{8,1} = 5$ and $b_{8,1} = 6 > a_{7,1} = 5$ hold. Therefore, the condition (*a*) does not hold, and the permutation $\pi_{2,1}$, which is a Johnson's one for the jobs from the set $\Im_{2,1}$ for any fixed scenario $p \in T$, does not exist.

We determine the binary relation $A^{2,1}_{\prec}$ on the set $\Im_{2,1}$ using the conditions (5) and (6). We conclude that the conditions (6) hold for the pair of jobs J_6 and J_7, and for the pair of jobs J_6 and J_8.

This gives us the relations $(J_6, J_7) \in A^{2,1}_{\prec}$ and $(J_6, J_8) \in A^{2,1}_{\prec}$. For the pair of jobs J_7 and J_8, neither the condition (5) nor the condition (6) holds. Therefore, the binary relation $A^{2,1}_{\prec}$ on the set $\Im_{2,1}$ has the following form: $J_6 \prec \{J_7, J_8\}$. The job set $\{J_7, J_8\}$ is a conflict set.

Next, we check the conditions (7)–(9) for the conflict set of jobs. We obtain

$$7 + 9 + 9 > 3 + 3 + 3 + 2 + 6.$$

Thus, the condition (7) does not hold.

To check the condition (8), we must check both orders of the conflicting jobs.

Consider the order (J_7, J_8). For $s = 1$, we obtain

$$9 < 3 + 3 + 3 + 2 + (6 - 7),$$

and for $s = 2$, we obtain

$$9 > 3 + 3 + 3 + 2 + (6 - 7) + (5 - 9).$$

Therefore, the condition (8) does not hold for the order (J_7, J_8) of the conflicting jobs.

Analogously, the condition (8) does not hold for the order (J_8, J_9) of the conflicting jobs.

Note that we do not check the condition (9) because the conflict set of jobs $\{J_7, J_8\}$ is at the end of the partial strict order $A^{2,1}_{\prec}$.

Thus, there is no a pair of job permutations (π', π'') of the jobs from the set $\Im$, which is optimal for the makespan criterion for all scenarios $p \in T$.

We use the second criterion to order the jobs in the conflict set, and obtain the following order: (J_8, J_7). Therefore, $\pi_{2,1} = (J_6, J_8, J_7)$.

Thus, to solve Example, one must consider the following pair of job permutations:

$$(\pi', \pi'') = ((\pi_{1,2}, \pi_1, \pi_{2,1}), (\pi_{2,1}, \pi_2, \pi_{1,2})) =$$
$$((J_3, J_1, J_2, J_4, J_6, J_8, J_7), (J_6, J_8, J_7, J_5, J_3, J_1, J_2)).$$

5 An Algorithm for Constructing a Daily Schedule for Two Employees

In [18], the algorithms were developed to test a set of schedule dominance conditions for the uncertain job-shop scheduling problem. When sufficient conditions hold for all conflict sets, the algorithms construct a pair of job permutations (π′, π″) that is optimal for all possible scenarios $p \in T$ of the uncertain problem $J2|a_{ij} \le p_{ij} \le b_{ij},\ m_i \le 2|C_{\max}$. Otherwise, the developed algorithms stop at the construction of the binary relations $A^{1,2}_{\prec}$ and $A^{2,1}_{\prec}$. The jobs from the sets $\Im_1$ and $\Im_2$ were arranged in increasing order of their indices.

Next, we propose a modification of the algorithms developed in [18] for the case of the following two-criteria uncertain job-shop scheduling problem: $J2|a_{ij} \le p_{ij} \le b_{ij},\ m_i \le 2\ |C_{\max}, \sum w_iC_i$.

Algorithm 1

Input: Set $\Im = \Im_1 \cup \Im_2 \cup \Im_{1,2} \cup \Im_{2,1}$ of the selected jobs; lower bounds a_{ij} and upper bounds b_{ij}, $0 < a_{ij} < b_{ij}$, job weights w_i, where $J_i \in \Im$, $M_j \in$ M.

Output: A pair of job permutations (π', π''), with the optimality proof or without an optimality proof.

Step. 1. Arrange the jobs of the sets $\Im_1$ and $\Im_2$ in the permutations π_1 and π_2 in the non-increasing order of their weights w_i.

Step 2. **IF** the condition $\sum_{J_i \in \Im_{1,2}} b_{i1} \le \sum_{J_j \in \Im_2 \cup \Im_{2,1}} a_{j2}$ holds, arrange the jobs in the permutation $\pi_{1,2}$ in the non-increasing order of their weights w_i. The permutation $\pi_{1,2}$ is optimal. **IF** the condition $\sum_{J_i \in \Im_{1,2}} a_{i2} \ge \sum_{J_j \in \Im_1 \cup \Im_{2,1}} b_{j1}$ holds, arrange the jobs in the permutation $\pi_{2,1}$ in the non-increasing order of the weights w_i. The permutation $\pi_{2,1}$ is optimal **ENDIF ENDIF.**

Step 3. **IF** both permutations $\pi_{1,2}$ and $\pi_{2,1}$ are optimal, **GOTO** *Step* 18 **ENDIF.**

Step 4. **IF** the condition $\sum_{J_i \in \Im_{2,1}} b_{i2} \le \sum_{J_j \in \Im_1 \cup \Im_{1,2}} a_{j1}$ holds, arrange the jobs in the permutation $\pi_{2,1}$ in the non-increasing order of the weights w_i. The permutation $\pi_{2,1}$ is optimal.

IF the condition $\sum_{J_i \in \Im_{2,1}} a_{i1} \ge \sum_{J_j \in \Im_2 \cup \Im_{1,2}} b_{j2}$ holds, arrange the jobs in the pemutation $\pi_{1,2}$ in the non-increasing order of the weights w_i. The permutation $\pi_{1,2}$ is optimal **ENDIF ENDIF.**

Step 5. **IF** both permutations $\pi_{1,2}$ and $\pi_{2,1}$ are optimal **GOTO** *Step* 18 **ENDIF.**

Step 6. **IF** the permutation $\pi_{1,2}$ is not optimal, divide the set $\Im_{1,2} = \Im^1_{1,2} \cup \Im^2_{1,2} \cup \Im^*_{1,2}$ as follows:

$$\Im^1_{1,2} = \{J_i \in \Im_{1,2} \mid b_{i1} \le a_{i2}\},$$
$$\Im^2_{1,2} = \{J_i \in \Im_{1,2} \mid b_{i2} \le a_{i1}\},$$
$$\Im^*_{1,2} = \{J_i \in \Im_{1,2} \mid b_{i1} > a_{i2}, b_{i2} > a_{i1}\}.$$

Step 7. Test the following conditions:

a) for each pair of jobs $J_i \in \Im^1_{1,2}$ and $J_j \in \Im^1_{1,2}$ (jobs $J_i \in \Im^2_{1,2}$ and $J_j \in \Im^2_{1,2}$, respectively), either $b_{i1} \le a_{j1}$ or $b_{j1} \le a_{i1}$ (either $b_{i2} \le a_{j2}$ or $b_{j2} \le a_{i2}$, respectively);

b) the inequality $|\Im^*_{1,2}| \le 1$ holds and for the job $J_{i^*} \in \Im^*_{1,2}$ (if any) both inequalities $a_{i^*1} \ge \max\{b_{i1} \mid J_i \in \Im^1_{1,2}\}$, $a_{i^*2} \ge \max\{b_{i2} \mid J_i \in \Im^2_{1,2}\}$ hold;

IF both conditions *a*) and *b*) hold, construct the permutation $\pi_{1,2} = (\pi^1, J_{i^*}, \pi^2)$ such that in the permutation π^1, jobs from the set $\Im^1_{1,2}$ are located in the increasing order of the values b_{i1}, while in the permutation π^2, jobs from the set $\Im^2_{1,2}$ are located in the decreasing order of the values b_{i2}. The resulting permutation $\pi_{1,2}$ is optimal.

Step 8. **ELSE** construct a binary relation $A^{1,2}_{\prec}$ on the set $\Im_{1,2}$ by pairwise comparing all jobs in this set as follows: $(J_u, J_v) \in A^{1,2}_{\prec}$, if $b_{u1} \le a_{v2}$ and $b_{u1} \le a_{v1}$, or $b_{v2} \le a_{v1}$ and $b_{v2} \le a_{u2}$; identifying all conflict sets of the jobs.

Step 9. **FOR** each conflict set **DO**:

Assume that k is the number of the last job before the conflict set in the binary relation $A^{1,2}_{\prec}$ and r is the cardinality of this conflict set.

Step 10. **IF** the condition

$$\sum_{i=1}^{k+r} b_{i,1} \le \sum_{J_i \in \Im_2 \cup \Im_{2,1}} a_{i,2} + \sum_{j=1}^{k} a_{j,2}$$

holds, arrange the jobs of this conflict set in the non-increasing order of the weights w_i; now the conflicts are resolved.

Step 11. **ELSE FOR** each job J_i in the conflict set

IF $a_{i,2} - b_{i,1} \ge 0$ **THEN** $J_i \in \pi^1$ **ELSE** $J_i \in \pi^2$.

ENDIF ENDFOR.

Construct the permutation (π^1, π^2), arrange the jobs in the permutation π^1 in the non-decreasing order of the upper bounds b_{i1} and arrange the jobs in the permutation π^2 in the non-increasing order of the lower bounds a_{i2}.

Step 12. **IF** the condition

$$b_{k+s,1} \le \sum_{J_i \in \Im_2 \cup \Im_{2,1}} a_{i,2} + \sum_{j=1}^{k+s-1} \left(a_{j,2} - b_{j,1}\right)$$

holds for $s \in \{1, 2, ..., r\}$, the conflicts are resolved.

Step 13. **ELSE**

FOR each job J_i in the conflict set

IF $a_{i,1} - b_{i,2} \ge 0$ **THEN** $J_i \in \pi^1$ **ELSE** $J_i \in \pi^2$.

ENDIF ENDFOR.

Construct the permutation (π^2, π^1), arrange the jobs in the permutation π^1 in the non-increasing order of the upper bounds b_{i2}, and arrange the jobs in the permutation π^2 in the non-decreasing order of the lower bounds a_{i1}.

Step 14. **IF** the condition $\sum_{i=r-s+2}^{r+1} a_{k+i,1} \geq \sum_{j=r-s+1}^{r} b_{k+j,2}$ holds for $s \in \{1,2,...,r\}$, the conflicts are resolved.

Step 15. **ELSE** arrange the jobs of the conflict set in the non-increasing order of the weights w_i, the conflict is not resolved **ENDIF ENDIF ENDIF ENDFOR ENDDO.**

IF all conflicts are resolved, the permutation $\pi_{1,2}$ is optimal. **ENDIF ENDIF ENDIF.**

Step 16. **IF** the permutation $\pi_{2,1}$ is optimal, **GOTO** *Step* 18 **ENDIF.**

Step 17. Repeat *steps* 6–15 by replacing the set $\Im_{1,2}$ with the set $\Im_{2,1}$, employee M_1 with employee M_2, the binary relation $A_{\prec}^{1,2}$ with the binary relation $A_{\prec}^{2,1}$, and the permutation $\pi_{1,2}$ with the permutation $\pi_{2,1}$.

Step 18. Construct the desired pair of job permutations $(\pi', \pi'') = ((\pi_{1,2}, \pi_1, \pi_{2,1}), (\pi_{2,1}, \pi_2, \pi_{1,2}))$.

It is easy to see that the asymptotic complexity of Algorithm 1 is $O(n^2)$ elementary operations.

As a result of executing the developed Algorithm 1, a pair of job permutations (π', π'') is constructed, which can be either optimal for all possible scenarios (with the proof of the optimality, if the sufficient conditions hold), or optimal for the factual scenario without proof of the optimality, or non-optimal for the makespan criterion of the uncertain (interval) job-shop scheduling problem $J2|a_{ij} \leq p_{ij} \leq b_{ij}, m_i \leq 2|C_{\max}$.

Note that arranging the jobs from the set $\Im_1$ and the set $\Im_2$, and in some cases arranging the jobs from the set $\Im_{1,2}$ and the set $\Im_2$,, in some conflict sets, were located in the non-decreasing order of their weights (see steps 1, 2, 4, 10 in Algorithm 1).

If the sufficient conditions do not hold, Algorithm 1 does not construct a permutation with a proof of its optimality. In such a case, we will arrange the jobs in the non-decreasing order of their weights in *step* 15 of Algorithm 1 in order to improve the achieved value of the second criterion.

6 Computational Experiments and Results

The developed Algorithm 1 was coded in MATLAB and tested on a large number of randomly generated instances of the two-criterion uncertain job-shop scheduling problem $J2|a_{ij} \leq p_{ij} \leq b_{ij}, m_i \leq 2|C_{\max}, \sum w_iC_i$.

The total number of jobs (the total number of jobs that employees chose to complete during the day) in the set $\Im$ was equal to 20. The number of jobs in each of the four subsets $\Im_1$, $\Im_2$, $\Im_{1,2}$, $\Im_{2,1}$ can be different in different series of the conducted computational experiment. Each series of the tests is characterized by the ratio: $n_{1,2} : n_1 : n_2 : n_{2,1}$. There were 100 trials (instances) in each series.

The experiments consisted of two parts. In the first part of the experiments, the equality $n_1 = n_2$ holds for each value from 1 to 5, and the values $n_{1,2}$ and $n_{2,1}$ take all possible combinations keeping the total number of jobs equal to 20.

In the second part of the computational experiment, either the equality $n_1 = 0$ holds or the equality $n_2 = 0$ holds, and the other three values were equal to all possible numbers with keeping the total number of jobs equal to 20.

The generation of the lower bounds a_{ij} and the upper bounds b_{ij} for possible values of the durations p_{ij} of the operations O_{ij}, $p_{ij} \in [a_{ij}, b_{ij}]$, was organized as follows. A value of the upper bound b_{ij} was randomly chosen from the segment [1, 100] based on the uniform distribution. With the given value of the relative length δ of the segment $[a_{ij}, b_{ij}]$, the lower bound a_{ij} was calculated using the following equality: $a_{ij} = b_{ij}(1 - \delta)$.

The maximum relative length δ of the segment of possible durations of the operations O_{ij} was equal to the following values: 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5. The bounds a_{ij} and b_{ij} were decimal fractions with the maximum possible number of digits after the decimal point.

The strict inequality $a_{ij} < b_{ij}$ was guaranteed for each job $J_i \in \Im$ and each employee $M_j \in$ M.

The weights w_i were randomly generated from the integers from 1 to 5.

For each tested instance, we used Algorithm 1 to construct a pair of job permutations (π', π''). The jobs were processed with respect to the constructed pair of the permutations (π', π'').

After all jobs were completed, the factual durations of all operations were known. In the experiment, the components of the random scenario p^* of the factual durations were generated using the uniform distribution from the segments $[a_{ij}, b_{ij}]$. Therefore, it was possible to calculate the factual value of both tested criteria.

For the constructed pair of permutations, we found the values of the criteria $C_{\max}(\pi', \pi'')$ and $\sum w_iC_i(\pi', \pi'')$ based on the factual durations of the operations. On the other hand, one can construct a Jackson's pair of the job permutations (π'_J, π''_J) for the factual scenario p^* and calculate the optimal values of the makespan $C_{\max}(\pi'_J, \pi''_J)$ and the weighted sum of the job completion times $\sum w_i C_i(\pi'_J, \pi''_J)$.

We say that the instance is solved optimally, if Algorithm 1 ends with the statement that the permutations $\pi_{1,2}$ and $\pi_{2,1}$ are optimal. This means that the above sufficient conditions are satisfied for all conflict sets (if any).

We say that the instance is solved optimally without a proof of optimality if the value of the criterion turns out to be optimal for the factual scenario of job durations, while not all conflict sets had sufficient conditions met.

In both cases, the relative makespan error was equal to zero, i.e., the following equality holds: $\Delta C_{\max} = 0$.

In other cases, the tested instances were not solved optimally. Therefore, we compared the values of both criteria for the computed job permutation pairs and factually optimal Jackson's job permutation pairs calculated after all jobs were completed.

We calculated the relative errors of the job permutations constructed by Algorithm 1 using the following formulas:

$$\Delta C_{\max} = \frac{C_{\max}(\pi', \pi'') - C_{\max}(\pi'_J, \pi''_J)}{C_{\max}(\pi'_J, \pi''_J)},$$

$$\Delta \sum w_i C_i = \frac{\sum w_i C_i(\pi', \pi'') - \sum w_i C_i(\pi'_J, \pi''_J)}{\sum w_i C_i(\pi'_J, \pi''_J)}.$$

Since we consider minimizing the schedule length as the main criterion, the criterion value for the constructed pair of job permutations $\sum w_i C_i(\pi', \pi'')$ is compared with the criterion value $\sum w_i C_i(\pi'_J, \pi''_J)$ calculated for the pair of Jackson's permutations that is optimal for the scenario with the factual operation durations according to the main criterion. For the tested instances with equality $\Delta C_{\max} = 0$, we assume that $\Delta \sum w_i C_i = 0$.

In the experiments, we estimated the average improvement and the maximal improvement of the weighted sum of job completion times.

We also found the number of jobs that were completed after the end of the workday in the constructed schedule.

We limited the size of an 8-hour workday to 800.

Table 2 and Table 3 present the computational results obtained for the first part of the experiments and for the second part of the experiments, respectively, for all tested instances.

The tables are organized as follows. In the first column, the value of the relative length δ of the segment of possible durations of the operations O_{ij} is presented, columns from 2 to 9 represent some indicator values.

We consider the following indicators:

Opt_Pr (in percentages, %) is the average value of tested instances solved optimally for the makespan criterion with the proof of the optimality;

Opt_Not_Pr (in %) is the average value of tested instances solved optimally for the makespan criterion without proof of the optimality;

Not_Opt (in %) means the average value of tested instances solved non-optimally for the makespan criterion,

Late_Job is the average number of jobs completed after the end of the working day;

Av_Cmax (in %) is the average relative makespan error;

Max_Cmax (in %) is the maximum makespan error;

Av_impr (in %) is the average improvement (for all instances tested) of the weighted sum of the weighted total completion time criterion for the non-optimally solved instances for the makespan criterion;

Max_impr (in %) is the maximum improvement of the weighted sum of the job completion times for the non-optimally solved instances for the makespan criterion.

The percentage of instances solved optimally for the makespan criterion with the proof of optimality decreases with increasing the value of δ, and for the experiment with $n_1 = n_2$ this value is larger than for the second part of the experiments.

In the individual cases, the value of the improvement of the second criterion for the factual scenario could be up to 48%.

Table 2. Average indicators for the experiments, where $n_1 = n_2$

δ	Opt_Pr, %	Opt_Not_Pr, %	Not_Opt, %	Late_Job	Av_Cmax, %	Max_Cmax, %	Av_impr, %	Max_impr, %
0.05	98.5	1.5	0.025	4.5	0.000	0.14	0.02	0.62
0.10	96.2	3.8	0	3.9	0	0	0	0
0.15	92.5	7.5	0.03	3.27	0.000	0.76	0.46	18.22
0.20	85.8	14.1	0.15	2.71	0.002	3.67	1.60	25.4
0.25	78.3	21.6	0.125	2.2	0.003	5.56	2.04	32.03
0.30	65.5	34.3	0.15	1.73	0.006	9.43	1.53	32.5
0.35	54.6	45.1	0.35	1.31	0.007	8.23	1.9	23.9
0.40	41.7	57.9	0.4	0.97	0.012	12.06	2.27	37.7
0.45	28.6	71.0	0.425	0.64	0.011	7.94	2.45	32.5
0.50	16.8	82.6	0.7	0.39	0.023	12.21	3.01	37.81

Table 3. Average indicators for the experiments, where either $n_1 = 0$ or $n_2 = 0$

δ	Opt_Pr, %	Opt_Not_Pr, %	Not_Opt, %	Late_Job	Av_Cmax, %	Max_Cmax, %	Av_impr, %	Max_impr, %
0.05	97.02	2.93	0.06	5.40	0.00	8.22	0.29	37.24
0.10	92.94	6.89	0.16	4.91	0.00	5.77	0.83	34.12
0.15	87.93	11.82	0.25	4.40	0.00	8.31	1.23	35.44
0.20	81.00	18.64	0.36	3.87	0.00	8.14	1.54	40.04
0.25	72.81	26.72	0.47	3.32	0.01	10.15	1.77	40.28
0.30	63.55	35.83	0.63	2.77	0.01	11.98	2.08	41.71
0.35	53.66	45.49	0.85	2.24	0.01	8.76	2.91	39.82
0.40	43.11	55.94	0.95	1.78	0.02	9.69	3.37	48.33
0.45	33.41	65.42	1.17	1.32	0.03	13.34	3.59	43.07
0.50	24.58	74.18	1.23	0.93	0.03	18.07	3.85	42.59

The constructed schedule is optimal for the makespan criterion with proof of the optimality if sufficient conditions are met, in particular, if all conflicts are resolved. The percentage of resolved conflicts relative to the total number of conflicts is shown in Figure 1 and Figure 2 as a function of the δ value. Figure 1 and Figure 2 present separate graphs for different values of $n_{1,2}$ and $n_{2,1}$ for the first and second part of the experiments, respectively. The percentage of resolved conflicts can be close to 100%. There is a clear tendency for the percentage of resolved conflicts to decrease with increasing the value of δ. However, even with the maximum δ tested of 0.5, on average at least 25% of the conflicts were resolved in the experiments with $n_1 = n_2$.

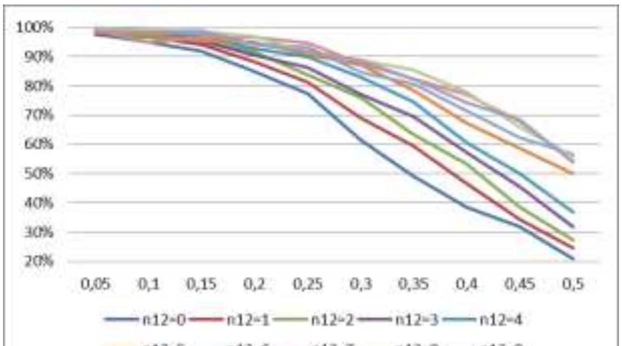

Fig. 1: Average percentage of the resolved conflicts for the experiments with $n_1 = n_2$

The total number of the tested instances for which the constructed pair of job permutations provides the optimal value of the makespan criterion consists of the optimally solved instances with or without the proof of the optimality. Table 4 and Table 5 show the number of optimally solved instances in the first and in second part of the experiments, respectively.

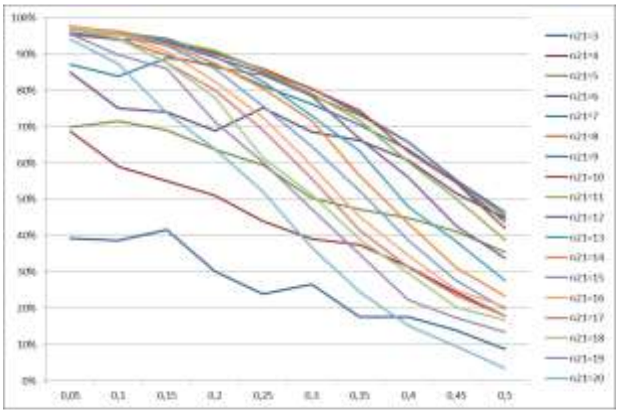

Fig. 2: Average percentage of the resolved conflicts for the experiments, where either $n_1 = 0$ or $n_2 = 0$

It can be seen that despite of the uncertainty of the operation durations, the proposed Algorithm 1 allows to find the optimal value of the makespan criterion for more than 93–95% of the tested instances, even with a large relative error δ = 0.5 of the input data, i.e., the relative error was 50%.

Table 4. The average percentage of the tested instances solved optimally for the makespan criterion, with or without the proof of the optimality, for the instances with $n_1 = n_2$

δ / n_{12}	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
0	99.8	100	99.8	99	99.4	98.8	97.6	97.6	97.8	95.8
1	100	100	100	99.8	99.6	100	99.6	99.2	98.8	98.6
2	100	100	100	100	100	100	100	100	100	100
3	100	100	100	100	100	100	100	100	100	100
4	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100
7	100	100	100	100	100	100	100	100	100	100
8	100	100	100	100	100	100	100	100	100	100
9	100	100	100	100	100	100	100	100	100	100

Table 5. An average percentage of tested instances solved optimally for the makespan criterion, with or without the proof of the optimality for the experiments, where either $n_1 = 0$ or $n_2 = 0$

δ / n_{12}	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
0	99.8	100	100	100	100	100	99.4	100	99.8	99.9
1	100	99.2	99.9	100	99.6	99.8	99.9	100	98.6	100
2	100	99.1	100	100	98.5	99.8	100	100	98.7	99.8
3	100	100	96.9	100	100	99.3	98.9	99.9	100	100
4	96	99.4	100	100	98	98.8	99.9	100	100	96.2
5	99.1	99.8	100	100	97.3	98.3	99.7	100	100	100
6	93.6	98.2	99.8	99.9	99.9	100	97.1	97.2	98.8	99.5
7	99.9	100	100	100	99.7	91.7	97.8	99.9	99.2	99.8
8	100	100	100	100	94.1	95.9	98.4	98.4	99.3	99.7
9	99.8	99.8	99.8	100	100	100	100	100	100	100
10	100	100	100	100	100	100	100	100	100	100

For the same series of experiments as in Figure 1 and Figure 2, Figure 3 and Figure 4 show graphs of the average relative improvements in the values of the weighted sum of job completion times for the instances that were not optimal due to the main makespan criterion.

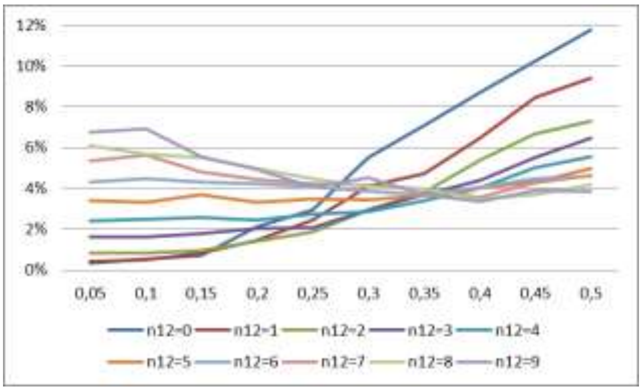

Fig. 3: Average improvements (in %) of the weighted sum of the job completion times for the experiments with $n_1 = n_2$

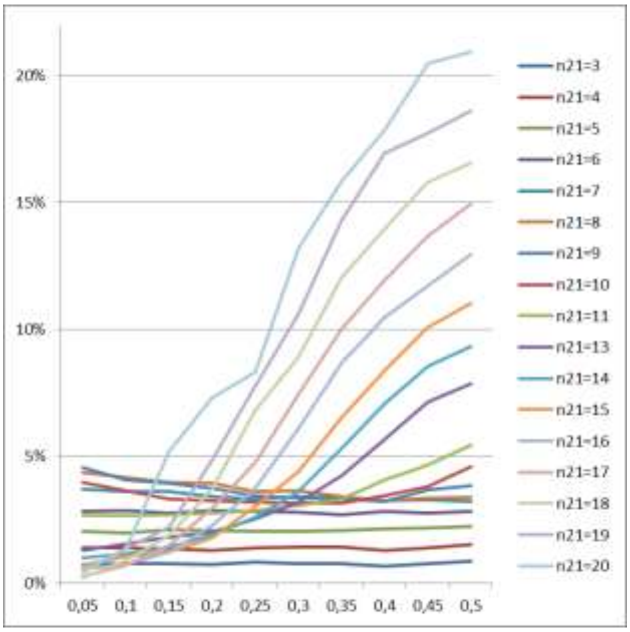

Fig. 4: Average improvements (in %) of the weighted sum of the job completion times for the experiments, where either $n_1 = 0$ or $n_2 = 0$

Figure 3 and Figure 4 show that on average the achieved improvements were up to 12–22%, and in general this value increases with increasing relative length δ of the segment $[a_{ij}, b_{ij}]$ of possible operation durations.

7 Concluding Remarks

We investigated the uncertain problems of scheduling the selected jobs for two employees. Only the lower and upper bounds for the possible duration of each selected job were assumed to be known before scheduling. For the dominant set of active schedules with the fixed orders in the pair of job permutations, the binary relation was constructed.

Based on the presented results, an efficient (polynomial) algorithm was developed to solve the uncertain job-shop scheduling problems either exactly or heuristically.

In order to test the effectiveness of the developed algorithm used for time management, the computational experiments were conducted on a personal computer for the evaluation of a day period for drawing up daily schedules for two employees.

Every day, 20 jobs were received for the execution. To schedule the jobs, the uncertain job-shop scheduling problem was solved. In the considered scheduling problem, two criteria were optimized in the fixed priority order. The minimization of the schedule length was the main criterion, while the minimization of the sum of the weighted completion times of the jobs was a second criterion.

A personal computer was used to select the most important jobs for two employees and to generate optimal schedules for their execution.

The computational experiments were conducted on randomly generated uncertain (interval) scheduling problems showed that the use of the job permutations constructed by the developed algorithm provides optimal schedules in more than 93–95% of the cases when the maximum relative length of the job duration interval $[a_{ij}, b_{ij}]$ does not exceed 50%. At the same time, the improvement of the received values of the second criterion amounted to 12–22%.

In future research, it will be promising to consider more than two ordered criteria for the uncertain (interval) job-shop scheduling problem arising in time management for two employees.

References:

[1] Claessens, B.J.C., Van Eerde, W., Rutte, C.G., Roe, R.A. A review of the time management literature, *Personnel Review*, Vol. 36, 2007, pp. 255–276, https://doi.org/10.1108/00483480710726136.

[2] Pakpoom, P., Charnsethikul, P. A stochastic programming approach for cyclic personnel scheduling with double shift requirement, *WSEAS Transactions on Systems and Control*, Vol. 13, 2018, pp. 275-284.

[3] Waldeyer, J., Dicke, T., Fleischer, J., Guo, J., Trentepohl, S., Wirth, J., Leutner, D. A moderated mediation analysis of conscientiousness, time management strategies, effort regulation strategies, and university students' performance, *Learning and Individual Differences*, Vol. 100, 2022, p. 102228. DOI: 10.1016/j.lindif.2022.102228.

[4] Bachrach, D.G., Rapp, T.L., Ogilvie, J., Rapp, A.A. It's about time (management)!: Role overload as a bridge explaining relationships between helping, voice, and objective sales performance, *Journal of Business Research*, Vol. 172, 2024, pp. 114295. DOI: 10.1016/j.jbusres.2023.114295.

[5] Macan, T.H. Time-management training: effects on time behaviors, attitudes, and job performance. *The Journal of Psychology*. Vol. 130(3), 1996, p. 229–236. DOI: 10.1080/00223980.1996.9915004.

[6] Häfner, A., Stock, A. Time Management Training and Perceived Control of Time at Work. *The Journal of Psychology*, Vol. 144(5), 2010, p. 429–447. DOI: 10.1080/00223980.2010.496647.

[7] Eilon S. Time-management, *OMEGA International Journal of Management Sciences*, Vol. 21(3), 1993, pp. 255–259, DOI: 10.1016/0305-0483(93)90084-X.

[8] Wang, Y., Lu, Y., Tian, X., Liu, Y., Ma, W. The relationship between mobile phone addiction and time management disposition among Chinese college students: a cross-lagged panel model, *Heliyon*, Vol. 10, 2024, e25060. DOI: 10.1016/j.heliyon.2024.e25060.

[9] Ahmad, N.L., Yusuf, A.N.M., Shobri, N.D.M., Wahab, S. The relationship between time management and job performance in event management, *Procedia – Social and Behavioral Sciences*, Vol. 65, 1993, pp. 937–941, DOI: 10.1016/j.sbspro.2012.11.223.

[10] Motair, H.M. Hybridization simulated annealing algorithm in a single machine scheduling problem, *WSEAS Transactions on Mathematics*, Vol. 20, 2021, pp. 598-606, DOI: 10.37394/23206.2021.20.63.

[11] Brucker, P. *Scheduling Algorithms*; Springer: Berlin, Germany, 1995, 326 p. DOI: 10.1004/978-3-662-03088-2.

[12] Graham, R.E., Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G. Optimization and approximation in deterministic sequencing and scheduling: a survey, *Annals of Discrete Mathematics*, Vol. 5, 1979, pp. 287–326, https://doi.org/10.1016/S0167-5060(08)70356-X.

[13] Jackson, J.R. An extension of Johnson's results on job lot scheduling, *Naval Research Logistics Quarterly*, Vol. 3, 1956, pp. 201–203, DOI: 10.1002/nav.3800030307.

[14] Johnson, S.M. Optimal two and three stage production schedules with set up times included. *Naval Research Logistics Quarterly*, Vol. 1, 1954, pp. 61–68, DOI: 10.1002/nav.3800010110.

[15] Briand, C., Trung La, H., Erschler J. A new sufficient condition of optimality for the two-machine owshop problem, *European Journal of Operational Research*, Vol. 169, 2006, pp. 712-722, DOI: 10.1016/j.ejor.2004.10.027.

[16] Lin, Y., Wang, X. Necessary and sufficient conditions of optimality for some classical scheduling problems, *European Journal of Operational Research*, Vol. 176, 2007, pp. 809–818, DOI: 10.1016/j.ejor.2005.09.017.

[17] Sotskov, Y.N., Matsveichuk, N.M., Hatsura, V.D., Two-machine job-shop scheduling problem to minimize the makespan with uncertain job durations, *Algorithms*, Vol. 13(5). No. 4, 2020, pp. 1–45, DOI: 10.3390/a13010004.

[18] Matsveichuk, N.M., Sotskov, Y.N., Werner, F. The dominance digraph as a solution to the two-machine flow-shop problem with interval processing times, *Optimization*, Vol. 60, 2011, pp. 1493–1517, DOI: 10.1080/02331931003657691.

[19] Aytug, H., Lawley, M.A., McKay, K., Mohan S., Uzsoy, R. Executing production schedules in the face of uncertainties: a review and some future directions, *European Journal of Operational Research*, Vol. 161, 2005, pp. 86–110, DOI: https://doi.org/10.1016/j.ejor.2003.08.027.

[20] Gupta, J.N.D., Stafford Jr. E.F. Flowshop scheduling research after five decades, *European Journal of Operational Research*, Vol. 169, 2006, pp. 699–711, DOI: 10.1016/j.ejor.2005.02.001.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed to the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

The research of the first and the second authors was partially supported by the Belarusian Republican Foundation for Fundamental Research, grant number Ф23РНФ-017. The research of the third author was partially supported by the National Natural Science Foundation of China (92367106), Natural Science Foundation of Hebei Province of China (F2023501006).

Conflict of Interest

The authors have no conflicts of interest interest to declare.