SEQUENCING JOBS WITH UNCERTAIN PROCESSING TIMES AND MINIMIZING THE WEIGHTED TOTAL FLOW TIME

Yuri Sotskov, Natalja Egorova

Abstract: We consider an uncertain version of the scheduling problem to sequence set of jobs J on a single machine with minimizing the weighted total flow time, provided that processing time of a job can take on any real value from the given closed interval. It is assumed that job processing time is unknown random variable before the actual occurrence of this time, where probability distribution of such a variable between the given lower and upper bounds is unknown before scheduling. We develop the dominance relations on a set of jobs J. The necessary and sufficient conditions for a job domination may be tested in polynomial time of the number $n = |J|$ of jobs. If there is no a domination within some subset of set J, heuristic procedure to minimize the weighted total flow time is used for sequencing the jobs from such a subset. The computational experiments for randomly generated single-machine scheduling problems with $n \leq 700$ show that the developed dominance relations are quite helpful in minimizing the weighted total flow time of n jobs with uncertain processing times.

Keywords: Scheduling, robustness and sensitivity analysis.

ACM Classification Keywords: F.2.2 Nonnumerical algorithms and problems: Sequencing and scheduling.

Conference: The paper is selected from XIV[th] International Conference "Knowledge-Dialogue-Solution" KDS 2008, Varna, Bulgaria, June-July 2008

Introduction

There are scheduling problems in real life, where job processing times may be evaluated with high reliability before scheduling, and the vast majority of academic research assumes that job processing times are either deterministic (see book [Tanaev, Sotskov, Strusevich, 1994] and the first part of book [Pinedo, 1995]) or random variables with known probability distributions (the second part of [Pinedo, 1995]). However, it is not realistic to assume all the job processing times have known probability distribution for many other practical scheduling problems. For the most scheduling environments, job processing times are unknown variables and the only information that can be certainly obtained before scheduling is about lower and upper bounds for a job processing time. As such, a schedule obtained by assuming a certain probability distribution may not be close to the optimal schedule in practical realization of the process. Due to this reason, methods of construction of optimal and approximate schedules are practically important for scheduling problems with uncertain (interval) processing times [Kouvelis, Yu, 1997; Sotskov, Sotskova, 2004].

In this paper, we address a scheduling problem when it is impossible to obtain reliable probability distributions for the job processing times. Namely, it is assumed that the processing time of a job can take any real value from the given interval of uncertainty, regardless of the values taken by the processing times of other jobs. More precisely, we consider the non-preemptive single-machine sequencing problem with interval processing times to minimize the weighted sum of the job completion times.

The paper is organized as follows. In the second section, problem setting is given. The third section contains a literature review. The forth section reminds a known-result for a single-machine scheduling problem with the fixed processing times and the weighted total flow time criterion. The fifth section contains the necessary and sufficient condition over which a job dominates another one (i.e., for each set of possible processing times there exists an optimal permutation with the same order of these two jobs). An illustrative example is given in the sixth section. Computational results for randomly generated instances with interval processing times are given in the seventh section. The last section presents a brief conclusion.

Problem Setting

There are *n* ≥ 2 jobs $J = \{J_1, J_2, ..., J_n\}$ to be processed on a single machine. For each job $J_i \in J$, positive weight $w_i > 0$ is given. Processing time p_i of a job $J_i \in J$ may take any real value between given lower bound $a_i \ge 0$ and upper bound b_i, $b_i \ge a_i$, which are only known before scheduling. Real number C_i is equal to the completion time of the job $J_i \in J$ and criterion $\sum w_i C_i = \sum_{i=1}^{n} w_i C_i$ denotes minimization of sum of the weighted completion times of *n* jobs. Let $S = \{\pi_1, \pi_2, ..., \pi_{n!}\}$ denote a set of all permutations $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_n})$ of *n* jobs from the set $J = \{J_1, J_2, ..., J_n\}$. By adopting the three-field notation $\alpha | \beta | \gamma$ introduced in [Graham et al., 1976], we denote the scheduling problem of searching an optimal permutation within set *S* that minimizes the value $\sum_{i=1}^{n} w_i C_i$ as $1 | a_i \le p_i \le b_i | \sum w_i C_i$. A set $T = \{p : a_i \le p_i \le b_i, J_i \in J\}$ of vectors $p = (p_1, p_2, ..., p_n)$ of the processing times is a rectangular box in the space of non-negative *n*-dimensional real vectors. If a vector *p* of the processing times is known before scheduling (i.e., $a_i = b_i, J_i \in J$), then problem $1 | a_i \le p_i \le b_i | \sum w_i C_i$ becomes conventional problem $1 || \sum w_i C_i$ with the fixed job processing times. As it is proven in [Smith, 1956], the latter problem can be solved in $O(n \log_2 n)$ time. We call sequencing problem $1 | a_i \le p_i \le b_i | \sum w_i C_i$ an uncertain (sequencing) problem in contrast to problem $1 || \sum w_i C_i$, called a deterministic one.

Literature Review and Definition

In case of the uncertain problem $1 | a_i \le p_i \le b_i | \sum w_i C_i$, there may not exists a unique schedule that remains optimal for all possible realizations of the job processing times. Therefore, in [Daniels, Kouvelis, 1995], so-called robust schedule minimizing the worst-case absolute or relative deviation from optimality (called worst-case regret) was proposed to hedge against processing time uncertainty. In [Daniels, Kouvelis, 1995; Yang, Yu, 2002], uncertain problem $1 | a_i \le p_i \le b_i | \sum C_i$ with minimizing total flow time (i.e., it was assumed that $w_i = 1$ for each job $J_i \in J$) has been considered. In [Averbakh, 2000; Averbakh, 2001; Daniels, Kouvelis, 1995; Yang, Yu, 2002; Lebedev, Averbakh, 2006] along with *continuous* intervals of possible processing times defined by the above set *T*, the processing time uncertainty was described through a finite *discrete* set $T^*, |T^*| = h,$ of possible processing time vectors (called scenarios). Each scenario $p \in T^*$ represents fixed processing times for job set *J*, which can be realized with some positive (but unknown before scheduling) probability. For a specific scenario $p \in T^*$, deterministic problem $1 || \sum C_i$ arises which can be solved using optimal job permutation defined due to the following SPT rule: *Sort the jobs J according to non-decreasing order of their processing times.*

While deterministic problem $1 || \sum C_i$ is computationally simple, finding a permutation which minimizes the worst-case regret to the uncertain counterpart with discrete set of possible scenarios $T^*, |T^*| = h,$ is computationally hard problem. E.g., in [Daniels, Kouvelis, 1995], it was proven that to find a permutation $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_n}) \in S$ minimizing the worst-case absolute regret is binary NP-hard problem (see [Garey, Johnson, 1979] for definition) even for two possible scenarios: $h = 2$. In [Yang, Yu, 2002], it was proven that to find a permutation $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_n}) \in S$ minimizing the worst-case relative regret is binary NP-hard problem for two possible scenarios as well. In [Yang, Yu, 2002], it was proven that to find a permutation $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_n}) \in S$ minimizing the worst-case absolute (relative) regret is unary NP-hard problem [Garey, Johnson, 1979] for unbounded number h of possible scenarios.

Worst-case regret is also defined for the processing time uncertainty described through a rectangular box T of possible vectors p. In [Lebedev, Averbakh, 2006], it was proven that minimizing the worst-case absolute regret for problem $1|a_i \le p_i \le b_i|\sum C_i$ is binary NP-hard problem if intervals of the processing times have the same center for all jobs J. In [Averbakh, 2001], it was shown by an example that there is no direct relationship between the complexity of the uncertain problem with the given finite discrete set of possible scenarios $T^*, |T^*| = h,$ and the complexity of the uncertain problem with the given set T of n continuous intervals of possible scenarios.

Summarizing this overview, we can observe that for the most classical polynomially solvable deterministic scheduling problems, their uncertain counterparts with the worst-case regret criterion become binary or unary NP-hard problems. In fact, even for existence of only two scenarios of possible processing times (h=2), to minimize the absolute or relative regret implies a time-consuming search over set S of n! permutations of n jobs. In order to overcome this computational complexity in some special cases, we propose to use searching the minimal set of dominant schedules (permutations) introduced in [Lai, Sotskov, 1999] for solving the uncertain job-shop problem $J|a_i \le p_i \le b_i|C_{\max}$ with the makespan objective function: $C_{\max} = \max\{C_i : J_i \in J\}$.

Definition 1: *Set of permutations (schedules) S(T)* $\subseteq$ *S is a minimal dominant set for the uncertain problem* $\alpha|a_i \le p_i \le b_i|\gamma$ *, if for any vector* $p \in T$ *set S(T) contains at least one permutation (schedule), which is optimal for the deterministic problem* $\alpha||\gamma$ *with vector* p *of the job processing times provided that any proper subset of set S(T) loses such a property.*

A minimal dominant set *S(T)* was investigated in [Allahverdi, Sotskov, 2003; Allahverdi, Aldowaisan, Sotskov, 2003; Lai, Sotskov, 1999; Leshchenko, Sotskov, 2007] for the makespan criterion $C_{\max}$, and in [Allahverdi, Aldowaisan, Sotskov, 2003; Sotskov, Allahverdi, Lai, 2004] for the total flow time criterion $\sum C_i$. In particular, work of [Sotskov, Allahverdi, Lai, 2004] was addressed to the total flow time in a two-machine flow-shop with the interval processing times: $F2|a_i \le p_i \le b_i|C_{\max}$. A geometrical algorithm has been developed for solving the flow-shop problem $Fm|n=2, a_i \le p_i \le b_i|\sum C_i$ with m machines and two jobs. For uncertain flow-shop problems with two or three machines, sufficient conditions have been identified when the transposition of two jobs minimizes the total flow time. Work of [Allahverdi, Aldowaisan, Sotskov, 2003] was addressed to the case of separate setup times with the criterion $C_{\max}$ or $\sum C_i$. Namely, the processing times were fixed while each setup time was relaxed to be a distribution-free random variable within given lower and upper bounds. Dominance relations have been identified for an uncertain flow-shop problem with two machines. In [Allahverdi, Sotskov, 2003], for a two-machine flow-shop problem $F2|a_i \le p_i \le b_i|C_{\max}$, sufficient conditions have been identified when the transposition of two jobs minimizes the makespan $C_{\max}$. In [Leshchenko, Sotskov, 2007], the necessary and sufficient conditions were used for the case when a single schedule dominates all the others, and the necessary and sufficient conditions were used for the case when it is possible to fix the optimal order of two jobs for the makespan criterion $C_{\max}$ with interval job processing times.

The formula for calculating stability radius of an optimal schedule (i.e., the largest value of independent variations of the job processing times for the schedule to remain optimal) has been provided in [Sotskov, Sotskova, Werner, 1997] for a job-shop problem $Jm|a_i \le p_i \le b_i|C_{\max}$ with m machines. Stability radius of an optimal schedule was investigated for problem $Jm|a_i \le p_i \le b_i|C_{\max}$ in [Lai, Sotskov, 1999; Sotskov, Wagelmans, Werner, 1998], and for problem $Jm|a_i \le p_i \le b_i|\sum C_i$ in [Brasel, Sotskov, Werner, 1996]. In contrast to references [Brasel, Sotskov, Werner, 1996; Lai, Sotskov, 1999; Sotskov, Sotskova, Werner, 1997; Sotskov, Wagelmans, Werner, 1998], where exponential algorithms based on exhausting enumeration of the semi-active schedules (see p. 284 in [Tanaev, Sotskov, Strusevich, 1998]) were derived for constructing minimal dominant set $S(T)$ for uncertain job-shop problems, in this paper, we show how to find set $S(T)$ for the problem $1|a_i \le p_i \le b_i|\sum C_i$ in polynomial time. Next, we present an auxiliary result for the deterministic problem $1||\sum w_i C_i$.

Deterministic Sequencing Problem

In [Smith, 1956], it was proven that problem $1||\sum w_iC_i$ can be solved in $O(n\log_2 n)$ time due to the following sufficient condition for optimality of permutation $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_n}) \in S$:

$$\frac{w_{i_1}}{p_{i_1}} \geq \frac{w_{i_2}}{p_{i_2}} \geq ... \geq \frac{w_{i_n}}{p_{i_n}}. \quad (1)$$

It is easy to prove that inequalities (1) are also necessary conditions for optimality of permutation $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_n}) \in S$ for the problem $1||\sum w_iC_i$.

Theorem 1: *Permutation* $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_n}) \in S$ *is optimal for the problem* $1||\sum w_iC_i$ *if and only if inequalities (1) hold.*

Proof: *Sufficiency* of condition (1) for optimality of permutation π_i was proven in [Smith, 1956].

Next, we prove *necessity* of condition (1) for optimality of permutation π_i by contradiction method.

Let permutation $\pi_i = (J_{i_1}, J_{i_2}, ..., J_{i_{r-1}}, J_{i_r}, J_{i_{r+1}}, J_{i_{r+2}}, ..., J_{i_n}) \in S$ be optimal for the problem $1||\sum w_iC_i$. However, for the latter permutation at least one inequality from condition (1) is violated, e.g., we assume that the following opposite inequality holds:

$$\frac{w_{i_r}}{p_{i_r}} < \frac{w_{i_{r+1}}}{p_{i_{r+1}}}, \quad (2)$$

where $r \in \{1,2,...,n-1\}$. Let us consider permutation $\pi_i' = (J_{i_1}, J_{i_2}, ..., J_{i_{r-1}}, J_{i_{r+1}}, J_{i_r}, J_{i_{r+2}}, ..., J_{i_n}) \in S$, which defers from permutation π_i by transposition of jobs J_{i_r} and $J_{i_{r+1}}$. We obtain the following equalities provided that notation $\Phi(\pi_i) = \Phi(J_{i_1}, J_{i_2}, ..., J_{i_n}) = \sum_{k=1}^{n} w_{i_k} C_{i_k}$ is used:

$$\Phi(\pi_i) = \sum_{q=1}^{n} w_{i_q} \sum_{k=1}^{q} p_{i_k}, \quad \Phi(\pi_i') = \sum_{q=1}^{r-1} w_{i_q} \sum_{k=1}^{q} p_{i_k} + w_{i_{r+1}} (\sum_{k=1}^{r-1} p_{i_k} + p_{i_{r+1}}) + w_{i_r} \sum_{k=1}^{r+1} p_{i_k} + \sum_{q=r+2}^{n} w_{i_q} \sum_{k=1}^{q} p_{i_k}.$$

Let us calculate the difference of the objective function values defined for permutation π_i and permutation π_i':

$$\Phi(\pi_i) - \Phi(\pi_i') = \sum_{q=1}^{n} w_{i_q} \sum_{k=1}^{q} p_{i_k} - \left[\sum_{q=1}^{r-1} w_{i_q} \sum_{k=1}^{q} p_{i_k} + w_{i_{r+1}} \left(\sum_{k=1}^{r-1} p_{i_k} + p_{i_{r+1}} \right) + w_{i_r} \sum_{k=1}^{r+1} p_{i_k} + \sum_{k=r+2}^{n} w_{i_q} \sum_{k=1}^{q} p_{i_k} \right] =$$

$$= \sum_{q=1}^{r-1} w_{i_q} \sum_{k=1}^{q} p_{i_k} + w_{i_r} \sum_{k=1}^{r} p_{i_k} + w_{i_{r+1}} \sum_{k=1}^{r+1} p_{i_k} + \sum_{k=r+2}^{n} w_{i_q} \sum_{k=1}^{q} p_{i_k} -$$

$$- \left[\sum_{q=1}^{r-1} w_{i_q} \sum_{k=1}^{q} p_{i_k} + w_{i_{r+1}} \left(\sum_{k=1}^{r-1} p_{i_k} + p_{i_{r+1}} \right) + w_{i_r} \sum_{k=1}^{r+1} p_{i_k} + \sum_{k=r+2}^{n} w_{i_q} \sum_{k=1}^{q} p_{i_k} \right] =$$

$$= w_{i_r} \sum_{k=1}^{r} p_{i_k} + w_{i_{r+1}} \sum_{k=1}^{r+1} p_{i_k} - w_{i_{r+1}} \left(\sum_{k=1}^{r-1} p_{i_k} + p_{i_{r+1}} \right) - w_{i_r} \sum_{k=1}^{r+1} p_{i_k} =$$

$$= w_{i_r} \left(\sum_{k=1}^{r} p_{i_k} - \sum_{k=1}^{r+1} p_{i_k} \right) + w_{i_{r+1}} \left(\sum_{k=1}^{r+1} p_{i_k} - \sum_{k=1}^{r-1} p_{i_k} - p_{i_{r+1}} \right) = w_{i_r}(-p_{i_{r+1}}) + w_{i_{r+1}} p_{i_r} = w_{i_{r+1}} p_{i_r} - w_{i_r} p_{i_{r+1}}.$$

Thus, the following equality holds: $\Phi(\pi_i) - \Phi(\pi_i') = w_{i_{r+1}} p_{i_r} - w_{i_r} p_{i_{r+1}}$. If we multiply both left-hand side and right-hand side of the latter inequality by factor $p_{i_r} p_{i_{r+1}}$, we obtain inequalities $w_{i_r} p_{i_{r+1}} < w_{i_{r+1}} p_{i_r}$ and $w_{i_{r+1}} p_{i_r} - w_{i_r} p_{i_{r+1}} > 0$ which implies: $\Phi(\pi_i') < \Phi(\pi_i)$. The latter inequality contradicts to the above assumption that permutation π_i is optimal for the problem $1||\sum w_i C_i$. The contradiction obtained implies the necessity of condition (1) for optimality of permutation π_i for the problem $1||\sum w_i C_i$. Theorem 1 is proven. ■

Uncertain Sequencing Problem

Search of the minimal dominant set *S(T)* for an uncertain problem $1|a_i \le p_i \le b_i|\sum w_i C_i$ may be based on constructing a dominance relation on the set of jobs *J*. To this end, we define a dominance relation as follows.

Definition 2: Job J_u dominates job J_v with respect to T (i.e., $J_u \rightarrow J_v$), if there exists a minimal dominant set S(T) for the problem $1|a_i \le p_i \le b_i|\sum w_i C_i$ that each permutation from set S(T) has either the form (…, J_u, J_v, …) or the form (…, J_u, …, J_v, …) (i.e., in a permutation $\pi_k \in S(T)$, job J_u precedes job J_v).

From Definition 2 it follows that minimal dominant set *S(T)* for the deterministic problem $1||\sum w_i C_i$ is a singleton: $\{\pi_k\} = S(T)$. As a result the following dominance relations hold: $J_{k_1} \rightarrow J_{k_2} \rightarrow \ldots \rightarrow J_{k_n}$. For a general case of the problem $1|a_i \le p_i \le b_i|\sum w_i C_i$, the following claim may be proven using Theorem 1.

Theorem 2: *For the problem $1|a_i \le p_i \le b_i|\sum w_i C_i$, job J_u dominates job J_v with respect to T if and only if the following inequality holds:*

$$\frac{w_u}{b_u} \ge \frac{w_v}{a_v}. \quad (3)$$

Due to Theorem 2, if job J_u dominates job J_v and job J_v dominates job J_i, then job J_u dominates job J_i as well. Thus, dominance relation $J_u \rightarrow J_v$ is transitive. Theorem 2 allows us to find a minimal dominant set $S(T)$ for the uncertain problem $1|a_i \le p_i \le b_i|\sum w_i C_i$ and to present set *S(T)* in compact form. Indeed via checking condition (3) for each pair of jobs J_u and J_v from the set *J*, we construct digraph $G = (J, A)$ of dominance relation on the set *J*: *Arc* (J_u, J_v) *belongs to set A if and only if dominance relation $J_u \rightarrow J_v$ holds.* Obviously, it takes $O(n^2)$ time to construct digraph $G = (J, A)$. If due to Theorem 2, linearly ordered set of jobs *J*, $J_{k_1} \rightarrow J_{k_2} \rightarrow \ldots \rightarrow J_{k_n}$, will be constructed, then set $S(T)$ for the problem $1|a_i \le p_i \le b_i|\sum w_i C_i$ will be a singleton: $\{\pi_k\} = S(T)$. And permutation $\pi_k \in S$ will be optimal for any possible scenario $p \in T$. It is easy to convince that in the case of $\{\pi_k\} = S(T)$, inequality $|A| \ge \frac{n(n-1)}{2}$ must hold for the digraph $G = (J, A)$.

Illustrative Example

Let input data for the instance of the problem $1|a_i \le p_i \le b_i|\sum w_i C_i$ be given in columns 1–4 of Table 1.

Table 1. Input data for the problem $1|a_i \le p_i \le b_i|\sum w_i C_i$

i	a_i	b_i	w_i	w_i / a_i	w_i / b_i
1	1	3	18	18	6
2	5	6	30	6	5
3	4	10	20	5	2
4	3	4	12	4	3
5	4	4	8	2	2
6	5	10	10	2	1
7	2	2	3	1.5	1.5
8	7	10	14	2	1.4

Via testing condition (3) of Theorem 2 for each pair of jobs $J_u \in J$ and $J_v \in J$ we obtain the following relations:

$$\frac{w_1}{b_1} = 6 \ge 6 = \frac{w_2}{a_2};\ \frac{w_2}{b_2} = 5 \ge 5 = \frac{w_3}{a_3};\ \frac{w_2}{b_2} = 5 \ge 4 = \frac{w_4}{a_4};\ \frac{w_3}{b_3} = 2 \ge 2 = \frac{w_5}{a_5};\ \frac{w_4}{b_4} = 3 \ge 2 = \frac{w_5}{a_5};$$

$$\frac{w_5}{b_5} = 2 \ge 2 = \frac{w_6}{a_6};\ \frac{w_5}{b_5} = 2 \ge 1.5 = \frac{w_7}{a_7};\ \frac{w_5}{b_5} = 2 \ge 2 = \frac{w_8}{a_8}.$$

Thus, condition (3) holds for the following ordered pair of jobs: J_1 and J_2; J_2 and J_3; J_2 and J_4; J_3 and J_5; J_4 and J_5; J_5 and J_6; J_5 and J_7; J_5 and J_8. Due to Theorem 2, the following dominance relations hold: job J_1 dominates job J_2; job J_2 dominates jobs J_3 and J_4; job J_3 dominates job J_5; job J_4 dominates job J_5; job J_5 dominates jobs J_6, J_7, and J_8. It is easy to verify that there are no other dominance relations except those that are transitive to the above ones. Therefore, minimal dominant set $S(T)$ for this instance of the problem $1|a_i \le p_i \le b_i|\sum w_i C_i$ consists of 2!·3! = 12 permutations.

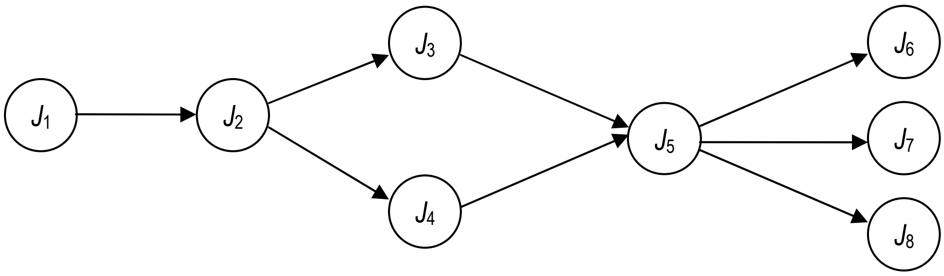

Fig. 1. Digraph $G = (J, A)$ without transitive arcs

Digraph $G = (J, A)$ defining set $S(T) = \{\pi_1, \pi_2, ..., \pi_{12}\}$ is represented in Fig. 1 (for simplicity, the transitive arcs are omitted). Thus, while searching optimal permutation for this instance of the uncertain problem $1|a_i \le p_i \le b_i|\sum w_i C_i$, it is sufficient to test only 12 permutations (instead of 8! = 256 feasible ones).

Computational Results

In this section, we describe the testing of randomly generated problems $1|a_i \le p_i \le b_i|\sum w_i C_i$ and answer (by experiments on PC) the question of how many pairs of jobs from set J satisfy condition (3) and how large errors of the optimal values of the criterion $\sum w_i C_i$ are for the schedules constructed using digraph $G = (J, A)$.

The computational algorithm was coded in C++. If relation $J_u \rightarrow J_v$ was fulfilled provided that $u < v$, our algorithm did not tested the validity of the opposite relation: $J_v \rightarrow J_u$. Therefore, an optimal permutation was obtained without fail, if equality $|A| = \frac{n(n-1)}{2}$ was fulfilled for the constructed digraph $G = (J, A)$.

For the experiments, we used an AMD 3000 MHz processor with 1024 MB main memory. We tested random instances of the uncertain problem $1|a_i \le p_i \le b_i|\sum w_i C_i$ with the following numbers of jobs: $n \in$ {10, 25, 50, 100, 150, …, 700}. The given integer lower and upper bounds of the possible integer processing times were uniformly distributed in the range [1, 100]. We tested the following errors L% of the uncertain job processing times: $L \in$ {0.1, 0.5, 1.0, 5.0, 10.0, 15.0, 20.0}. For each job $J_i \in J$, the given lower bound of a job processing time was randomly generated in the range [1, 100] and the upper bound was computed as follows: $b_i = a_i(1 + L\%/100\%)$. For each job $J_i \in J$, the weight $w_i > 0$ was randomly generated in the range [1, 50].

Table 2 represents the computational results for 80 series of the randomly generated instances. Each series included 10 instances with the same combination of the above n and L. The left-hand side of Table 2 (columns 1

– 6) represents the computational results for instances with the numbers of jobs from set {10, 25, 50, 100, 150,…, 300}. The right-hand side of Table 2 (columns 7 – 12) represents the computational results for instances with the number of jobs from set {350, 400,…, 700}. The number of series is given in column 1 (for series numbered from 1 to 40), and in column 7 (for series numbered from 41 to 80). The number of jobs in one instance is given in the corresponding column 2 or 8. The error L of the uncertain job processing times (in percentage) is given in the corresponding columns 3 or 9. The average error of the objective function value $\Phi^0 = \sum_{i=1}^{n} w_i C_i^0$ calculated for the heuristic schedules constructed due to Theorem 2 and digraph $G = (J, A)$, with respect to optimal objective function value $\Phi^* = \sum_{i=1}^{n} w_i C_i^*$, is given in the corresponding columns 4 or 10 (namely, values $(\Phi^0 - \Phi^*) : \Phi^*$ are given in columns 4 and 10). The average relative number of arcs $|A|$ (in percentage) constructed due to validity of condition (3), with respect to the number of arcs in the complete circuit-free digraph, is given in the corresponding columns 5 or 11 (namely, values $(|A| : \frac{n(n-1)}{2})100\%$ are given in columns 5 and 11). The average CPU-time (in seconds) used by the processor AMD 3000 MHz for solving one instance (approximately or exactly) is given in the corresponding columns 6 or 12.

From the experiments, it follows that dominance relation stated in Theorem 2 allow us to solve exactly all the instances from the series with numbers 1 – 10 and 14 (see column 5). The lowest relative number of arcs, i.e. 78.81069%, was constructed for the series with number 75. The lowest average quality of the schedules, i.e. $(\Phi^0 - \Phi^*) : \Phi^* = 0.759582$, was obtained for the series with the largest number 80. The largest CPU-time, 37.1 s, was obtained for the series with number 48. The average quality of the schedules obtained depends of the error L of the job processing times and remains almost the same for the instances with different number of jobs provided that they have the same error $L\%$ of the uncertain processing times. Increasing simultaneously both numbers *n* and L decreases the number of instances solved exactly due to Theorem 2.

Table 2. Computational results for randomly generated instances $1 | a_i \le p_i \le b_i | \sum w_i C_i$

	n	Error *L* %	Objective error	Number of arcs,%	CPU-time,s		*n*	Error *L* %	Objective error	Number of arcs,%	CPU-time,s
1	2	3	4	5	6	7	8	9	10	11	12
1	10	0.1	0.000000	100.00000	0	41	350	1.0	0.004562	99.181989	1.9
2	25	0.1	0.000000	100.00000	0	42	400	1.0	0.004622	99.160902	3.3
3	50	0.1	0.000000	100.00000	0	43	450	1.0	0.004386	99.175452	5.3
4	100	0.1	0.000000	100.00000	0	44	500	1.0	0.004430	99.179559	8.2
5	150	0.1	0.000000	100.00000	0.1	45	550	1.0	0.004539	99.179467	11.9
6	200	0.1	0.000000	100.00000	0.2	46	600	1.0	0.004494	99.202282	17.1
7	250	0.1	0.000000	100.00000	0.5	47	650	1.0	0.004504	99.197867	24.8
8	300	0.1	0.000000	100.00000	1.1	48	700	1.0	0.004738	99.174290	37.1
9	10	0.5	0.000000	100.00000	0	49	350	5.0	0.051612	94.942939	1.7
10	25	0.5	0.000000	100.00000	0	50	400	5.0	0.052478	94.784712	3.0
11	50	0.5	0.000058	99.991837	0	51	450	5.0	0.052236	94.871171	4.8
12	100	0.5	0.000000	99.997980	0	52	500	5.0	0.052704	94.917756	7.3
13	150	0.5	0.000000	99.996421	0	53	550	5.0	0.051536	95.060772	10.9
14	200	0.5	0.000000	100.00000	0.3	54	600	5.0	0.054459	94.834780	15.2
15	250	0.5	0.000006	99.998394	0.7	55	650	5.0	0.053217	94.908475	21.2
16	300	0.5	0.000033	99.996210	1.7	56	700	5.0	0.050923	94.883303	29.3

17	10	1.0	0.000374	98.888889	0	57	350	10.0	0.196527	89.868522	1.5
18	25	1.0	0.004613	99.333333	0	58	400	10.0	0.189964	89.885965	2.5
19	50	1.0	0.003874	99.412245	0	59	450	10.0	0.199014	89.876169	4.2
20	100	1.0	0.003450	99.236364	0	60	500	10.0	0.198004	89.393587	6.3
21	150	1.0	0.005091	99.314541	0.1	61	550	10.0	0.182142	89.829641	9.4
22	200	1.0	0.004504	99.190452	0.2	62	600	10.0	0.192427	89.771786	13.3
23	250	1.0	0.004385	99.178474	0.5	63	650	10.0	0.191777	89.593694	18.9
24	300	1.0	0.004397	99.196433	0.1	64	700	10.0	0.197116	89.578500	26.9
25	10	5.0	0.010130	92.666667	0	65	350	15.0	0.416224	84.606631	1.3
26	25	5.0	0.044132	95.033333	0	66	400	15.0	0.421506	84.389724	2.2
27	50	5.0	0.033901	95.346939	0	67	450	15.0	0.419705	84.949567	3.6
28	100	5.0	0.049146	95.032323	0	68	500	15.0	0.450631	84.497074	5.5
29	150	5.0	0.050174	94.741834	0	69	550	15.0	0.432102	84.505647	8.1
30	200	5.0	0.050892	95.029648	0.2	70	600	15.0	0.447372	84.360045	11.4
31	250	5.0	0.054236	94.993092	0.5	71	650	15.0	0.418753	84.208131	15.6
32	300	5.0	0.049330	94.978595	0.9	72	700	15.0	0.454469	84.130922	21.6
33	10	10.0	0.129524	89.111111	0	73	350	20.0	0.747840	79.328530	1.2
34	25	10.0	0.124472	89.400000	0	74	400	20.0	0.779839	79.001754	1.8
35	50	10.0	0.179310	90.889796	0	75	450	20.0	0.775389	78.810690	3.0
36	100	10.0	0.201273	89.715152	0	76	500	20.0	0.749320	79.445210	4.7
37	150	10.0	0.199625	89.681432	0.1	77	550	20.0	0.767790	79.044080	6.8
38	200	10.0	0.185334	89.533668	0.1	78	600	20.0	0.755315	79.437340	9.8
39	250	10.0	0.197567	89.108112	0.4	79	650	20.0	0.779930	79.073272	13.4
40	300	10.0	0.190829	89.575920	0.8	80	700	20.0	0.759582	79.465195	18.4

Conclusion

The main issue of this paper is to show how to construct a minimal dominant set *S*(*T*) in polynomial time via constructing digraph $G=(J, A)$ as a compact presentation of set *S*(*T*) of dominant permutations. We estimated a strength of using minimal dominant set *S*(*T*) by extensive computational experiments for randomly generated problems $1|a_i \le p_i \le b_i|\sum w_i C_i$ with number *n* of jobs from the range [10, 700].

Bibliography

[Allahverdi, Sotskov, 2003] A. Allahverdi, Yu.N. Sotskov. Two-machine flowshop minimum-length scheduling with random and bounded processing times, *International Transactions in Operational Research* 10 (2003) 65-76.

[Allahverdi, Aldowaisan, Sotskov, 2003] A. Allahverdi, T. Aldowaisan, Yu.N. Sotskov. Two-machine flowshop scheduling problem to minimize makespan or total completion time with random and bounded setup times, *International Transactions of Mathematical Sciences* 39 (2003) 2475-2486.

[Averbakh, 2000] I. Averbakh. Minmax regret solutions for minmax optimization problems with uncertainty, *Operations Research Letters* 27 (2000) 57-65.

[Averbakh, 2001] I. Averbakh. On the complexity of a class of combinatorial optimization problems with uncertainty, *Mathematical Programming,* Series A 90 (2001) 263-272.

[Brasel, Sotskov, Werner, 1996] H.-M. Brasel, Yu. N. Sotskov, F. Werner. Stability of a schedule minimizing mean flow time, *Mathematical and Computer Modelling* 24 (10) (1997) 39-53.

[Daniels, Kouvelis, 1995] R.L. Daniels, P. Kouvelis. Robust scheduling to hedge against processing time uncertainty in single-stage production, *Management Science* 41 (2) (1995) 363-376.

[Graham et al., 1976] R.L. Graham, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: A survey, *Annals of Discrete Mathematics* 5 (1976) 287-326.

[Garey, Johnson, 1979] M.R. Garey, D.S. Johnson. *Computers and Intractability. A Guide to the Theory of NP-Completeness.* San Francisko: Freeman, USA, 1979.

[Kouvelis, Yu, 1997] P. Kouvelis, G. Yu. *Robust Discrete Optimization and Its Applications*. Boston: Kluwer Academic Publishers, USA, 1997.

[Lebedev, Averbakh, 2006] V. Lebedev, I. Averbakh. Complexity of minimizing the total flow time with interval data and minmax regret criterion, *Discrete Applied Mathematics* 154 (2006) 2167-2177.

[Leshchenko, Sotskov, 2007] N. Leshchenko, Yu. N. Sotskov. Realization of an optimal schedule for the two-machine flow-shop with interval job processing times, *International Journal "Information, Theory & Applications"* 14 (2007) 182-189.

[Lai, Sotskov, 1999] T.-C. Lai, Yu. N. Sotskov. Sequencing with uncertain numerical data for makespan minimization, *Journal of the Operational Research Society* 50 (1999) 230-243.

[Pinedo, 1995] M. Pinedo. *Scheduling: Theory, Algorithms, and Systems.* Prentice-Hall: Enlewood Cliffs, USA, 1995.

[Smith, 1956] W.E. Smith, Various optimizers for single-stage production, *Naval Research and Logistics Quarterly* 3 (1) (1956) 59-66.

[Sotskov, Allahverdi, Lai, 2004] Yu.N. Sotskov, A. Allahverdi, T.-C. Lai. Flowshop scheduling problem to minimize total completion time with random and bounded processing times, *Journal of the Operational Research Society* 55 (2004) 277-286.

[Sotskov, Sotskova, 2004] Yu. N. Sotskov, N. Sotskova. *Scheduling Theory: Systems with Uncertain Numerical Parameters,* Minsk: United Institute of Informatics Problems, Belarus, 2004 (in Russian).

[Sotskov, Sotskova, Werner, 1997] Yu. N. Sotskov, N. Sotskova, F. Werner. Stability of an optimal schedule in a job shop, *Omega - Journal of Operational Research* 25 (4) (1997) 245-280.

[Sotskov, Wagelmans, Werner, 1998] Yu.N. Sotskov, A.P.M. Wagelmans, F. Werner. On the calculation of the stability radius of an optimal or an approximate schedule, *Annals of Operations Research* 83 (1998) 213-252.

[Tanaev, Sotskov, Strusevich, 1994] V.S. Tanaev, Yu.N. Sotskov, V.A. Strusevich. *Scheduling Theory: Multi-Stage Systems,* Dordrecht: Kluwer Academic Publishers, Netherlands, 1994.

[Yang, Yu, 2002] J. Yang, G. Yu. On the robust single machine scheduling problem, *Journal of Combinatorial Optimization* 6 (2002) 17-33.

Authors' Information

Yuri N. Sotskov – Professor, DSc, PhD, United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Surganova str., 6, 220012, Minsk, Belarus; e-mail: sotskov@newman.bas-net.by

Natalja G. Egorova – Junior researcher, United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Surganova str., 6, 220012, Minsk, Belarus; e-mail: egorova@newman.bas-net.by