On Computational Complexity of Scheduling Problems with Job Requisitions

Yu. Zakharova

Sobolev Institute of Mathematics SB RAS

Omsk–2022

The research was supported by RSF grant 22-71-10015

Report Structure

- Problem Statement
- Previous Research
- Single-stage statements
- Multi-stage systems
- Solving Approach
- Conclusion and Further Research

Problem Statement

- $\mathcal{J}$, $|\mathcal{J}| = n$, is the set of jobs.
- $\mathcal{M}$, $|\mathcal{M}| = m$, is the set of machines.
- Single-stage statements and multi-stage systems.
- p_{vj} the duration (processing time) of operation v of job j.
- $K_l = \{1, \ldots, k_l\}$ is the set of positions on machine l.
- Requisitions of jobs: $X^{i,l}$ is the subset of jobs (operations), which can be performed in position $i \in K_l$ of machine l.
- The goal is to assign jobs (or their operations) to positions of machines so that a polynomially computable regular criterion has the minimum value.

Motivation

Technological Constraints

Technological requisitions in production systems and multi-processor computer systems, where the order of job execution is influenced by setup times, fixed routes, working shifts, structural constraints and other factors.

Optimal Recombination

Given two parent permutations of jobs $\pi^1 = (\pi^1_1, \dots, \pi^1_n)$ and $\pi^2 = (\pi^2_1, \dots, \pi^2_n)$. It is required to find an offspring permutation $\pi' = (\pi'_1, \dots, \pi'_n)$ such that

(I) $\pi'_i = \pi^1_i$ or $\pi'_i = \pi^2_i$ for all $i = 1, \dots, n$;

(II) π' has the minimum objective value over all permutations satisfying condition (I).

Then jobs π^1_i and π^2_i compose requisition $X^{i,1}$ for position i of the machine.

Previous Research

- Serdyukov A.I.: On travelling salesman problem with prohibitions, Upravlaemye systemi (1978)
- Еремеев А.В., Коваленко Ю.В.: О сложности оптимальной рекомбинации для одной задачи составления расписаний с переналадками, Дискретн. анализ и исслед. опер. (2012)
- Eremeev A., Kovalenko Yu.: Optimal recombination in genetic algorithms for combinatorial optimization problems, Yugoslav Journal of Operations Research (2014)
- Eremeev A., Kovalenko Yu.: Experimental evaluation of two approaches to optimal recombination for permutation problems, LNCS 9595 (2016)
- Eremeev A., Kovalenko Yu.: On solving travelling salesman problem with vertex requisitions, Yugoslav Journal of Operations Research (2017)

Single Processor Problem with Job Requisitions

Input Data

- Jobs $j \in \mathcal{J}$: release date r_j, due date d_j, duration (processing time) p_j and weight w_j.
- Job requisitions: $X^i,\ i = 1, \ldots, n = |\mathcal{J}|$.

Criteria

$1|r_j = 0, d_j, w_j|\sum_j w_j U_j$ (the weighted number of tardy jobs);
$1|r_j, d_j|L_{\max} = \max_j L_j$ (the maximum lateness);
$1|r_j, d_j|\sum_j U_j$ (the number of tardy jobs);
$1|r_j, d_j|\sum_j T_j$ (the total tardiness);
$1|r_j = 0, C_j \leq d_j, w_j|\sum_j w_j C_j$ (the weighted total completion time);
$1|r_j|\sum_j C_j$ (the total completion time);
$1||C_{\max} = \max_j C_j$ (the makespan);
$2||C_{\max}$ and $2|r_j = r,\ d_j = d|L_{\max}$.

NP-hardness

Ordered 2-Partition Problem

Given ordered set $A = \{a_1, a_2, \dots, a_{2n_0}\}$ and weight e_i of each element $a_i \in A$ such that $\sum_{a_i \in A} e_i = 2E$ and $e_i < e_{i+1},\ i = 1, \dots, 2n_0 - 1.$

The question is to decide whether A can be partitioned into two subsets A_1 and A_2 so that

$$\sum_{a_i \in A_1} e_i = \sum_{a_i \in A_2} e_i = E,\ |A_1| = |A_2| = n_0,$$

and subset A_1 contains only one element from each pair $a_{2i-1},\ a_{2i},\ i = 1, \dots, n_0.$

Properties

A schedule is called non-idle if the processor is no idle during the interval $[r_{\min}, d_{\max}]$.
An instance of the problem has the non-idle property if there is no feasible schedule that is not non-idle.

Reduction

NP-hardness proofs are based on the polynomial reduction of the Ordered 2-Partition Problem to the decision version of the scheduling problem.
The obtained instance of the decision problem has the non-idle property.
Basic jobs j correspond to elements a_j and job requisitions contain pairs $a_{2i-1},\ a_{2i}$.
We have threshold values and additional critical jobs.
It is required to partition basic jobs into two parts.

NP-hardness

Properties

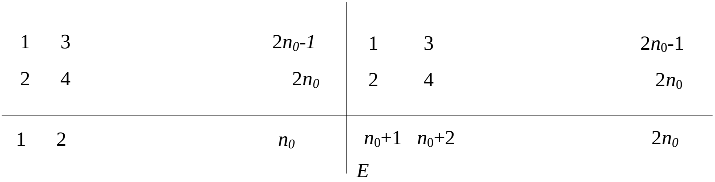
A schedule is called non-idle if the processor is no idle during the interval $[r_{\min}, d_{\max}]$.
An instance of the problem has the non-idle property if there is no feasible schedule that is not non-idle.

Reduction

NP-hardness proofs are based on the polynomial reduction of the Ordered 2-Partition Problem to the decision version of the scheduling problem.
The obtained instance of the decision problem has the non-idle property.
Basic jobs j correspond to elements a_j and job requisitions contain pairs a_{2i-1}, a_{2i}.
We have threshold values and additional critical jobs.
It is required to partition basic jobs into two parts.

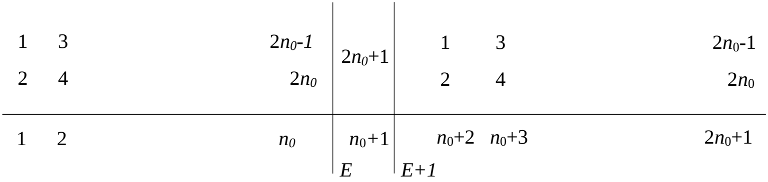
$1|r_j = 0, d_j = d, w_j, X^i|\sum_j w_j U_j$

The number of jobs $n = 2n_0$.
Job characteristics $p_j = w_j = e_j$, $d_j = E$, $j \in \mathcal{J}$.
Job requisitions $X^{i+n_0} = X^i = \{2i-1,\ 2i\}$, $i = 1, \ldots, n_0$.
Threshold value $\sum_j w_j U_j(\pi) \le E$.



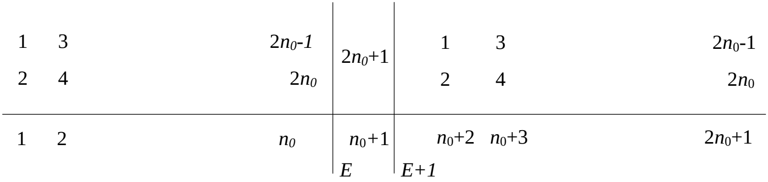
$1|r_j, d_j, X^i|\gamma,\ \gamma \in \{L_{\max}; \sum_j U_j; \sum_j T_j\}$

The number of jobs $n = 2n_0 + 1$.
Job characteristics:
$p_j = e_j,\ d_j = 2E + 1,\ r_j = 0$ for $j = 1, \dots, 2n_0$;
$p_{2n_0+1} = 1,\ r_{2n_0+1} = E,\ d_{2n_0+1} = E + 1$.
Job requisitions:
$X^{i+n_0+1} = X^i = \{2i-1,\ 2i\},\ i = 1, \dots, n_0$;
$X^{n_0+1} = \{2n_0 + 1\}$.
Threshold value $L_{\max}(\pi) \leq 0$ ($\sum_j U_j(\pi) \leq 0$ or $\sum_j T_j(\pi) \leq 0$).



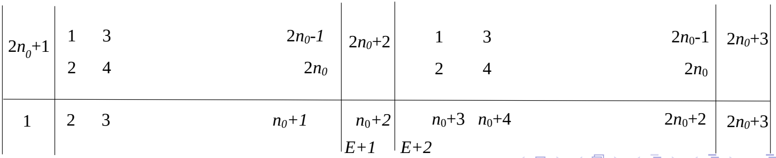
$1|r_j = 0, C_j \le d_j, w_j, X^i|\sum_j w_j C_j$

The number of jobs $n = 2n_0 + 1$.
Job characteristics:
$p_j = w_j = e_j$, $d_j = 2E + 1$ for $j = 1, \dots, 2n_0$;
$p_{2n_0+1} = 1$, $w_{2n_0+1} = 0$, $d_{2n_0+1} = E + 1$.
Job requisitions:
$X^{i+n_0+1} = X^i = \{2i - 1,\ 2i\}$, $i = 1, \dots, n_0$;
$X^{n_0+1} = \{2n_0 + 1\}$.
Threshold value $\sum_j w_j C_j(\pi) \le \sum_{1 \le i \le j \le 2n_0} e_i e_j + E$.



$1|r_j, X^i|\sum_j C_j$

The number of jobs $n = 2n_0 + 3$.
Job characteristics:
$p_j = e_j,\ r_j = 0$ for $j = 1, \ldots, 2n_0$;
$p_{2n_0+1} = p_{2n_0+2} = p_{2n_0+3} = 1$,
$r_{2n_0+1} = 0,\ r_{2n_0+2} = E + 1,\ r_{2n_0+3} = 2E + 2$.
Job requisitions:
$X^{i+n_0+2} = X^{i+1} = \{2i - 1,\ 2i\},\ i = 1, \ldots, n_0$;
$X^1 = \{2n_0 + 1\},\ X^{n_0+2} = \{2n_0 + 2\},\ X^{2n_0+3} = \{2n_0 + 3\}$.
Threshold value $\sum_j C_j(\pi') \le L := (1) + (E + 2) + (2E + 3) + \left(\sum_{j=1}^{n_0}(n_0 - j + 1)(e_{2j-1} + e_{2j}) + 1n_0 + (E + 2)n_0\right)$.



Job-shop Scheduling with Job Requisitions

Input Data

$\mathcal{J} = \{1, \ldots, n\}$ is the set of jobs.
$\mathcal{M} = \{1, \ldots, m\}$ is the set of machines.
n_j is the number of sequential operations for job j.
Operation O_v^j has duration p_{vj} and uses machine $L_{vj} \in \mathcal{M}$.
$L_j = (L_{1j}, L_{2j}, \ldots, L_{n_j,j})$ may be different for different jobs.

Requisitions

$O_l = \{O_v^j : \ L_{vj} = l\}$ is the set of operations for machine l, $|O_l| = k_l$.
$X^{i,l}$ is the requisition for position $i = 1, \ldots, k_l$ of machine $l \in \mathcal{M}$.

Criteria

makespan $C_{\max} = \{C_j : j \in \mathcal{J}\}$
maximum lateness $L_{\max} = \{C_j - d_j : j \in \mathcal{J}\}$

$J2|X^{i,l}|C_{\max}$

Input Data

$m = 2,\ n = 2n_0 + 2.$

Job characteristics:

$p_{1,j} = 0,\ \ p_{2,j} = e_j, j = 1, \ldots, 2n_0;$

$p_{1,2n_0+1} = 2E,\ \ p_{2,2n_0+1} = E;$

$p_{2,2n_0+2} = 2E,\ \ p_{1,2n_0+2} = E.$

Job Requisitions

$$X^{11} = \{2n_0 + 1\},\ X^{21} = \{2n_0 + 2\},$$

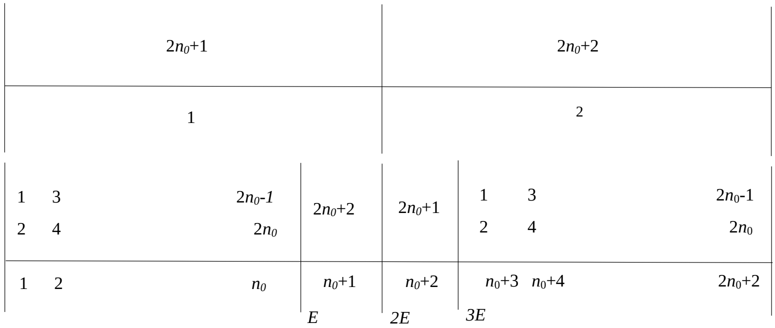
$$X^{i+n_0+2,2} = X^{i,2} = \{2i - 1,\ 2i\},\ i = 1, \ldots, n_0,$$

$$X^{n_0+1,2} = \{2n_0 + 2\},\ X^{n_0+2,2} = \{2n_0 + 1\}.$$

Threshold

$C_{\max}(\{\pi^l\}_{l \in \mathcal{M}}) \leqslant 4E.$

$J2|X^{i,l}|C_{\max}$



2n0+1
2n0+2
1
2
1 3 2n0-1
2 4 2n0
2n0+2
2n0+1
1 3 2n0-1
2 4 2n0
1 2 n0
n0+1
n0+2
n0+3 n0+4 2n0+2
E
2E
3E

Flow-shop Scheduling with Job Requisitions

Input Data

$\mathcal{J} = \{1, \dots, n\}$ is the set of jobs.
$\mathcal{M} = \{1, \dots, m\}$ is the set of machines.
Job $j \in \mathcal{J}$ is processed on $1 \to 2 \to \dots \to m$.
Operation O_l^j has duration p_{lj} and uses machine $l \in \mathcal{M}$.
Flow-shop with missing operations: $p_{lj} = 0$ means that job j on machine l is not executed. [a]

[a] Eremeev A., Kovalenko Yu.: On solving travelling salesman problem with vertex requisitions (2017)

Requisitions

$O_l = \{O_l^j : \; p_{lj} \neq 0\}$ is the set of operations for $l, \; |O_l| = k_l$.
$X^{i,l}$ is the requisition for position $i = 1, \dots, k_l$ of machine $l \in \mathcal{M}$.

Criteria

$C_{\max} = \{C_j : j \in \mathcal{J}\}$, $L_{\max} = \{C_j - d_j : j \in \mathcal{J}\}$

$F3|Miss-Oper, X^{i,l}|C_{\max}$

Input Data

$m = 3$, $n = 2n_0 + 2$.

Job characteristics:

$p_{1,i} = p_{3,i} = e_i,\ p_{2,i} = 0, i = 1, \ldots, 2n_0;$

$p_{1,2n_0+1} = p_{3,2n_0+2} = E,\ p_{2,2n_0+1} = p_{2,2n_0+2} = 2E;$

$p_{3,2n_0+1} = p_{1,2n_0+2} = 0.$

Job Requisitions

$X^{i,1} = X^{n_0+1+i,1} = \{2i-1, 2i\},\ i = 1, \ldots, n_0,$

$X^{n_0+1,1} = \{2n_0+1\},$

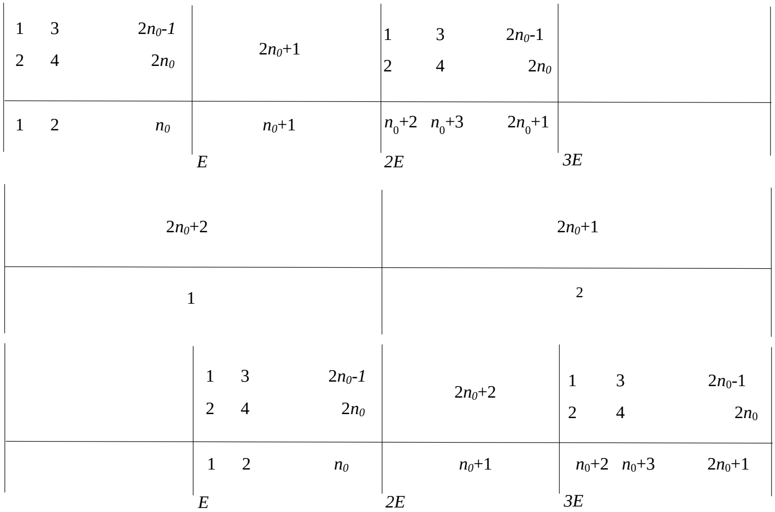
$X^{1,2} = \{2n_0+2\},\ = X^{2,2} = \{2n_0+1\},$

$X^{i,3} = X^{n_0+1+i,3} = \{2i-1, 2i\},\ i = 1, \ldots, n_0,$

$X^{n_0+1,3} = \{2n_0+2\}.$

Threshold

$C_{\max}(\{\pi^l\}_{l\in\mathcal{M}}) \leqslant 4E.$


1 3 2n0-1
2 4 2n0
2n0+1
1 3 2n0-1
2 4 2n0
1 2 n0
n0+1
n0+2 n0+3 2n0+1
E
2E
3E
2n0+2
2n0+1
1
2
1 3 2n0-1
2 4 2n0
2n0+2
1 3 2n0-1
2 4 2n0
1 2 n0
n0+1
n0+2 n0+3 2n0+1
E
2E
3E

Solving Approach, $|X^i| \le 2$

Main Idea

- $\bar{G} = (X_n, X, \bar{U})$ is the bipartite graph.
- $\bar{U} = \{\{i, x\} : i \in X_n,\ x \in X^i\}$ is the set of edges.
- Vertices of the left part $\leftrightarrow$ positions.
- Vertices of the right part $\leftrightarrow$ jobs.
- There is a one-to-one correspondence between the set of perfect matchings $\mathcal{W}$ in the graph $\bar{G}$ and the set Π of feasible permutations to a problem instance $I(\gamma,\ X^i)$[a].

[a] Serdyukov A.I. (1978); Eremeev A., Kovalenko Yu. (2017)

Types of Edges

- An edge $\{i, x\} \in \bar{U}$ is called *special* if $\{i, x\}$ belongs to all perfect matchings in the graph $\bar{G}$.
- All edges, except for the special edges and those adjacent to them, are slit into cycles.

Finding special edges and cycles in graph $\bar{G}$, $O(n)$

Step 1 *(Initialization)*. Assign $\bar{G}' := \bar{G}$.
Step 2. Repeat Steps 2.1-2.2 while it is possible:
Step 2.1 *(Solvability test)*. If the graph $\bar{G}'$ contains a vertex of degree 0 then, problem $I(\gamma,\ X^i)$ is infeasible, terminate.
Step 2.2 *(Finding a special edge)*. If the graph $\bar{G}'$ contains a vertex z of degree 1, then store the corresponding edge $\{z, y\}$ as a special edge and remove its endpoints y and z from $\bar{G}'$.

- The cycles of the graph $\bar{G}$ can be computed in $O(n)$ time using the Depth-First Search algorithm.
- $q(\bar{G}) = q(I)$ is the number of cycles in the graph $\bar{G}$ for instance $I(\gamma,\ X^i)$.
- Each cycle j, $j = 1, \ldots, q(\bar{G})$, contains exactly two maximal perfect matchings.

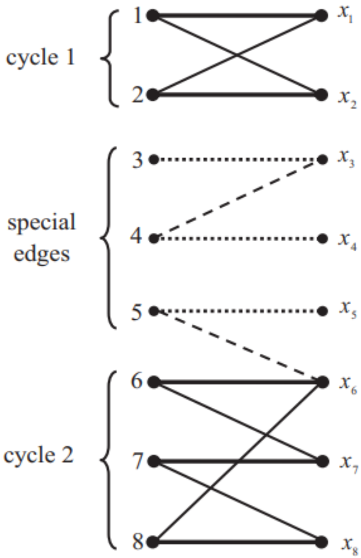
Solving $I(\gamma,\ X^i)$

Step 1. Build the bipartite graph $\bar{G}$, identify the set of special edges and cycles and find all maximal matchings in cycles.
Step 2. Enumerate all perfect matchings $W \in \mathcal{W}$ of $\bar{G}$ by combining the maximal matchings of cycles and joining them with special edges.
Step 3. Assign the corresponding solution $\pi \in \Pi$ to each $W \in \mathcal{W}$ and compute $\gamma(\pi)$.
Step 4. Output the result $\pi^* \in \Pi$, such that $\gamma(\pi^*) = \min_{\pi \in \Pi} \gamma(\pi)$.

Time Complexity

$O(T(\gamma)2^{q(I)})$, where $q(I) = q(\bar{G}) \leq \lfloor \frac{n}{2} \rfloor$ and the last inequality is tight, $T(\gamma)$ is the time for computing γ.

Example



Almost all feasible instances

A graph $\bar{G} = (X_n, X, \bar{U})$ is called "good" if it satisfies the inequality $q(\bar{G}) \leq 1.1 \ln n$.

$\bar{\chi}_n$ is the set of "good" bipartite graphs $\bar{G} = (X_n, X, \bar{U})$.

χ_n is the set of all bipartite graphs $\bar{G} = (X_n, X, \bar{U})$.

$\frac{|\bar{\chi}_n|}{|\chi_n|} \to 1$ as $n \to \infty$ (Serdyukov A.I., 1978).

Theorem

Almost all feasible instances $I(\gamma,\ X^i)$ with $|X^i| \leq 2$ have at most n feasible solutions and thus, they are solvable in $O(T(\gamma)n)$ time.

Almost all feasible instances

Job-shop

Job-shop scheduling problem $J|X^{i,l}|\gamma$, $\gamma \in \{C_{\max}, L_{\max}\}$, with job requisitions such that $|X^{i,l}| \leq 2$ can be solved in $O\left(1.42^{k_{\max}m}(n^2 n_{\max}^2 + nmn_{\max}k_{\max})\right)$ time.

Almost all instances of $Jm|X^{i,l}|\gamma$, $\gamma \in \{C_{\max}, L_{\max}\}$, with job requisitions such that $|X^{i,l}| \leq 2$ are polynomially solvable.

Flow-shop

Flow-shop scheduling problems $F|X^{i,l}|\gamma$, $F|Miss - Oper, X^{i,l}|\gamma$, $\gamma \in \{C_{\max}, L_{\max}\}$, with job requisitions such that $|X^{i,l}| \leq 2$ can be solved in $O\left(1.42^{nm}(nm)\right)$ time.

Almost all instances of $Fm|X^{i,l}|\gamma$, $Fm|Miss - Oper, X^{i,l}|\gamma$, $\gamma \in \{C_{\max}, L_{\max}\}$, with job requisitions such that $|X^{i,l}| \leq 2$ are polynomially solvable.

MIP-model (variables)

Boolean variables

$$x_{ij} = \begin{cases} 1, & \text{if job } j \text{ is performed in position } i, \\ 0 & \text{otherwise,} \end{cases}$$

$i = 1, \ldots, n, \; j \in X^i.$

Continuous variables

$y_i \geq 0$ is the duration of the job in position i (auxiliary variable);

$z_i \geq 0$ is the release date of the job in position i (auxiliary variable);

$v_i \geq 0$ is the due date of the job in position i (auxiliary variable);

C_i is the completion time of a job in position i.

MIP-model (constraints)

$$\sum_{j \in X^i} x_{ij} = 1,\ i = 1, \dots, n, \tag{1}$$

$$\sum_{i \in Y^j} x_{ij} = 1,\ j \in \mathcal{J}, \tag{2}$$

$$C_i \geq C_{i-1} + y_i,\ i = 2, \dots, n, \tag{3}$$

$$C_i \geq z_i + y_i,\ i = 1, \dots, n, \tag{4}$$

$$y_i = \sum_{j \in X^i} x_{ij} p_j,\ i = 1, \dots, n, \tag{5}$$

$$z_i = \sum_{j \in X^i} x_{ij} r_j,\ i = 1, \dots, n, \tag{6}$$

$$v_i = \sum_{j \in X^i} x_{ij} d_j,\ i = 1, \dots, n, \tag{7}$$

$$C_i \geq 0,\ x_{ij} \in \{0, 1\},\ i = 1, \dots, n,\ j \in X^i. \tag{8}$$

MIP-model (criteria)

the maximum lateness

$$L_{\max} \geq C_i - v_i,\ i = 1, \dots, n,$$

the total tardiness

$$T_{\sum} = \sum_{i=1}^{n} T_i,$$

$$T_i \geq 0,\ T_i \geq C_i - v_i,\ i = 1, \dots, n,$$

the total completion time

$$C_{\sum} = \sum_{i=1}^{n} C_i,$$

the number of tardy jobs

$$U_{\sum} = \sum_{i=1}^{n} U_i,$$

$$C_i \leq v_i + U_i \cdot BigM,\ U_i \in \{0, 1\},\ i = 1, \dots, n.$$

New MIP-model (variables)

Boolean variables

$$x_l = \begin{cases} 0, & \text{if the first matching is selected in cycle } l, \\ 1, & \text{if the second matching is selected in cycle } l, \end{cases}$$

$l = 1, \dots, q(\bar{G})$.

Continuous variables

$y_i \geq 0$ is the duration of the job in position i (auxiliary variable);

$z_i \geq 0$ is the release date of the job in position i (auxiliary variable);

$v_i \geq 0$ is the due date of the job in position i (auxiliary variable);

C_i is the completion time of a job in position i.

New MIP-model (constraints)

$$C_i \geq C_{i-1} + y_i,\ \ i = 2, \dots, n, \tag{9}$$

$$C_i \geq z_i + y_i,\ \ i = 1, \dots, n, \tag{10}$$

$$y_i = p_i^0(1 - x_l) + p_i^1 x_l,\ \ l = 1, \dots, q(\bar{G}),\ \ i \in N_l, \tag{11}$$

$$y_i = p_i^0,\ \ i = 1, \dots, n : |X^i| = 1,$$

$$z_i = r_i^0(1 - x_l) + r_i^1 x_l,\ \ l = 1, \dots, q(\bar{G}),\ \ i \in N_l, \tag{12}$$

$$z_i = r_i^0,\ \ i = 1, \dots, n : |X^i| = 1,$$

$$v_i = d_i^0(1 - x_l) + d_i^1 x_l,\ \ l = 1, \dots, q(\bar{G}),\ \ i \in N_l, \tag{13}$$

$$v_i = d_i^0,\ \ i = 1, \dots, n : |X^i| = 1,$$

$$C_i \geq 0,\ \ i = 1, \dots, n, \tag{14}$$

$$x_l \in \{0, 1\},\ \ l = 1, \dots, q(\bar{G}). \tag{15}$$

MIP-model (variables)

Boolean variables

$$x_{oik} = \begin{cases} 1, & \text{if operation } o \text{ is performed in position } k \text{ on machine } i, \\ 0 & \text{otherwise,} \end{cases}$$

$i = 1, \dots, m,\ \ k = 1, \dots, n_i,\ \ o \in X^{i,k}.$

Continuous variables

$C_{ik} \geq 0$ is the completion time of a job operation in position k of machine i.

MIP-model (constraints)

$$\sum_{o \in X^{i,k}} x_{oik} = 1,\ i = 1, \dots, m,\ k = 1, \dots, n_i, \tag{16}$$

$$\sum_{k \in Y^{o,i}} x_{oik} = 1,\ i = 1, \dots, m,\ o \in O_i, \tag{17}$$

$$C_{ik} + \sum_{o \in X^{i,k+1}} p_o x_{o,i,k+1} \le C_{i,k+1}, \tag{18}$$

$$i = 1, \dots, m,\ k = 1, \dots, n_i - 1,$$

$$C_{i_1,k_1} + p_{o_2} \le C_{i_2,k_2} + \tag{19}$$

$$BigM(1 - x_{o_1,i_1,k_1}) + BigM(1 - x_{o_2,i_2,k_2}) +$$

$$j \in \mathcal{J}, v = 1, \dots, k_j - 1,\ o_1 = O^j_v,\ o_2 = O^j_{v+1},$$

$$i_1 = L_{v,j},\ i_2 = L_{v+1,j},\ k_1 \in Y^{o_1,i_1},\ k_2 \in Y^{o_2,i_2},$$

$$C_{ik} \ge 0,\ x_{oik} \in \{0,1\},\ o \in X^{i,k},\ i \in \mathcal{M},\ k = 1, \dots, n_i. \tag{20}$$

New MIP-model (variables)

Boolean variables

$$x_{il} = \begin{cases} 0, & \text{if the first matching is selected in cycle } l \text{ of machine } i, \\ 1, & \text{if the second matching is selected in cycle } l \text{ of machine } i; \end{cases}$$

$i = 1, \ldots, m,\ l = 1, \ldots, q(i).$

Continuous variables

$C_{ik} \geq 0$ is the completion time of a job operation in position k of machine i.

$C(O_v^j) \geq 0$ is the completion time of operation O_v^j.

MIP-model (constraints)

$$C_{ik} + y_{i,k+1} \leq C_{i,k+1}, \ i \in \mathcal{M}, \ k = \overline{1, n_i}, \quad (21)$$

$$C(O_v^j) + p_{v+1,j} \leq C(O_{v+1}^j), \ j \in \mathcal{J}, \ v = \overline{1, k_j - 1}, \quad (22)$$

$$C(O_v^j) \geq C_{L_{vj}, k_{vj}^0} - BigM(x_{L_{vj}, l_{vj}}), \ j \in \mathcal{J}, \ v = \overline{1, k_j}, \quad (23)$$

$$C(O_v^j) \leq C_{L_{vj}, k_{vj}^0} + BigM(x_{L_{vj}, l_{vj}}), \ j \in \mathcal{J}, \ v = \overline{1, k_j}, \quad (24)$$

$$C(O_v^j) \geq C_{L_{vj}, k_{vj}^1} - BigM(1 - x_{L_{vj}, l_{vj}}), \ j \in \mathcal{J}, \ v = \overline{1, k_j}, \quad (25)$$

$$C(O_v^j) \leq C_{L_{vj}, k_{vj}^1} + BigM(1 - x_{L_{vj}, l_{vj}}), \ j \in \mathcal{J}, \ v = \overline{1, k_j}, \quad (26)$$

$$y_{ik} = p_{ik}^0 (1 - x_{i, l_{ik}}) + p_{ik}^1 x_{i, l_{ik}}, \ i \in \mathcal{M}, \ k = \overline{1, n_i}, \quad (27)$$

$$C(O_v^j) \geq 0, \ j \in \mathcal{J}, \ v = \overline{1, k_j}, \quad (28)$$

$$C_{ik} \geq 0, \ i \in \mathcal{M}, \ k = \overline{1, n_i}, \quad (29)$$

$$x_{il} \in \{0, 1\}, \ i \in \mathcal{M}, \ l = 1, \dots, q(i). \quad (30)$$

Multi-machine Scheduling with Job Requisitions

Algorithm

The total number of positions over all machines is equal to the number of jobs.
The multi-machine problem can be solved using the same methods as the single-machine one.

Example

M_1	$\{j_1, j_5\}$	$\{j_9, j_{10}\}$		
M_2	$\{j_3, j_5\}$	$\{j_4, j_6\}$	$\{j_6, j_7\}$	$\{j_2, j_8\}$
M_3	$\{j_2, j_9\}$	$\{j_7, j_8\}$	$\{j_1, j_{10}\}$	
M_4	$\{j_{11}\}$	$\{j_3, j_4\}$		

Open-shop with Job Requisitions

Statement

The order in which the jobs are processed on the machine, and the order in which the job is processed by the machines can be chosen arbitrarily. Position i of machine l may contain only jobs from the given subset $X^{i,l}$.

NP-hardness and Algorithm

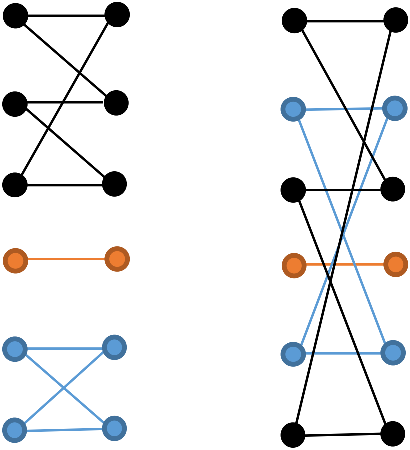
When $X^{i,l} = \{i\}$ for all $l \in \mathcal{M}$, we have the classic flow-shop scheduling problem by reversing the sense of jobs and machines (NP-hard even in the case of three jobs[a]).

Using the same arguments as in the NP-hardness proof for job-shop with job requisitions we obtain
Theorem. Open-shop scheduling problem $O2|X^{i,l}|\gamma,\ \gamma \in \{C_{\max}, L_{\max}\}$, is NP-hard.

An objective can not be computed in polynomial time for the fixed sequences of jobs on machines.

[a] Sotskov Yu., Shakhlevich N. (1993)

Dependent and Independent Cycles



Conclusion and Further Research

Conclusion

1. NP-hardness proofs of scheduling problems with job requisitions.
2. Enumeration approach and MIP methods for solving single stage and multi stage instances.

Further Research

1. Computational complexity of shop scheduling problems with the total completion time criterion.
2. Experimental evaluation of various solving approaches.
3. New properties of cycles in bipartite graphs and parallelization.
4. Importance of solution representation.

Thank you for your attention!