Single-machine scheduling with an external resource

Speaker:
Dirk Briskorn
Department of Production and Logistics, University of Wuppertal, Germany

Co-authors:
Morteza Davari
SKEMA Business School, Lille, France
Jannik Matuschke
Research Centre for Operations Management, KU Leuven, Belgium

Oct 2023 @ schedulingseminar.com

External resources

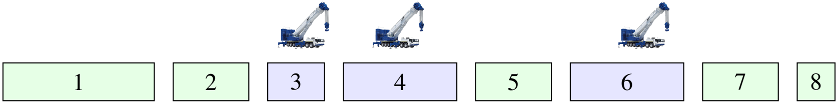
Resources that are rented

External resources

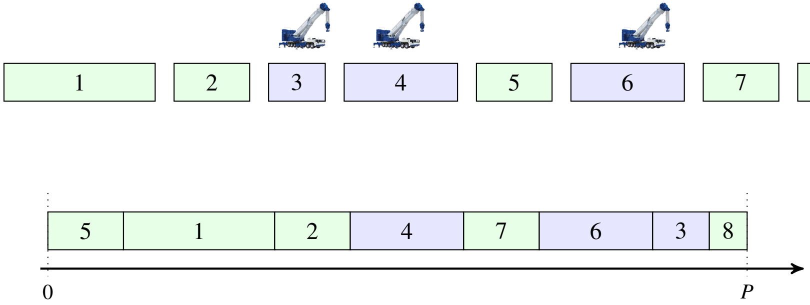
Resources that are rented

Human experts invited in projects

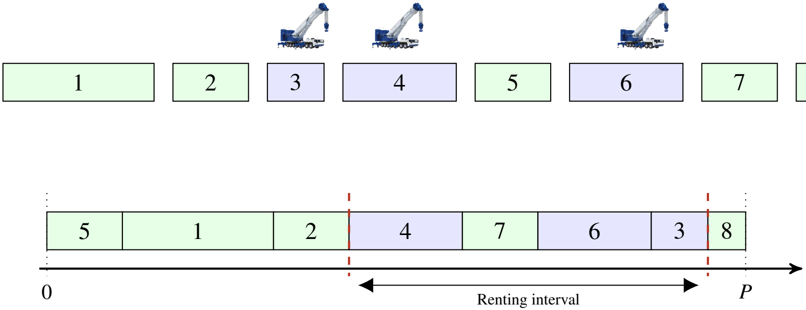
Problem definition



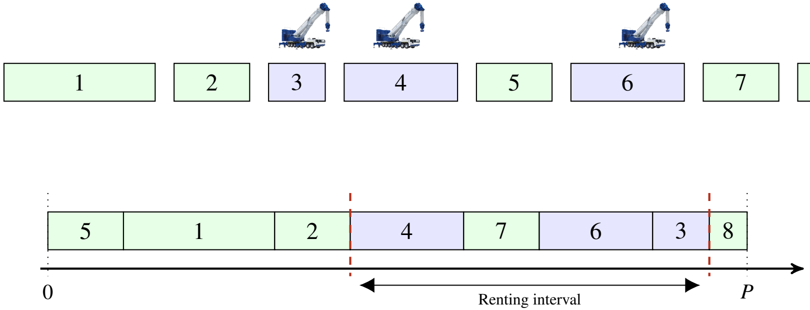
Problem definition



Problem definition



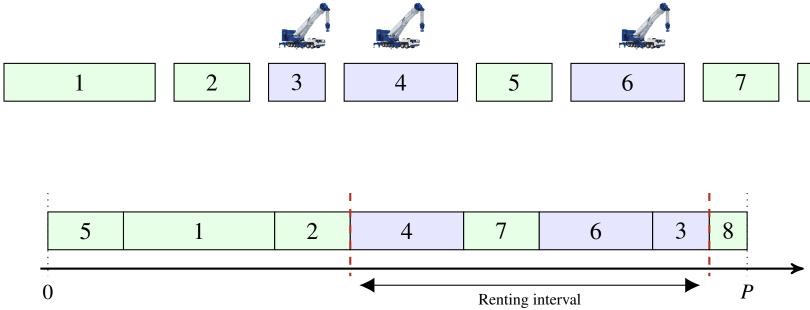
Problem definition



We have two type of costs

1. The scheduling cost (γ)
2. The renting cost (er)

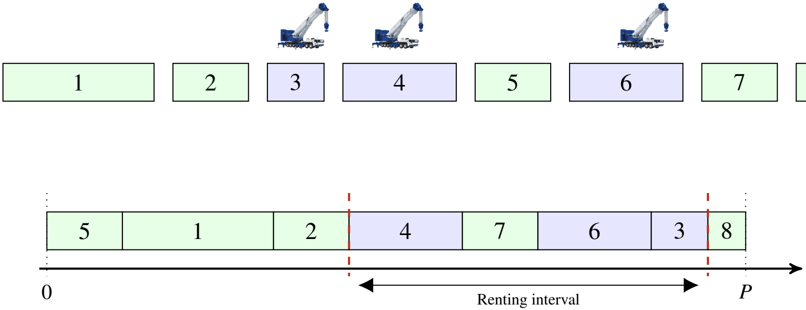
Problem definition



We have two type of costs

1. The scheduling cost (γ) $\gamma = \sum C_j, \sum w_jC_j, \max L_j, \sum w_jU_j$
2. The renting cost (er)

Problem definition



We have two type of costs

1. The scheduling cost (γ) — $\gamma = \sum C_j, \sum w_jC_j, \max L_j, \sum w_jU_j$
2. The renting cost (er) — Linear function of length of the renting interval

Problem definition

Four variants corresponding to $1||\gamma$

Problem	To optimize	To respect
$1\|er\|\gamma$	the scheduling cost (γ)	renting interval length of at most K^r

Problem definition

Four variants corresponding to $1||\gamma$

Problem	To optimize	To respect
$1\|er\|\gamma$	the scheduling cost (γ)	renting interval length of at most K^r
$1\|\gamma\|er$	renting interval length (er)	scheduling cost at most K^γ

Problem definition

Four variants corresponding to $1||\gamma$

Problem	To optimize	To respect
$1\|er\|\gamma$	the scheduling cost (γ)	renting interval length of at most K^r
$1\|\gamma\|er$	renting interval length (er)	scheduling cost at most K^γ
$1\|\|(\gamma, er)$	Pareto-front of γ and er	-

Problem definition

Four variants corresponding to $1||\gamma$

Problem	To optimize	To respect
$1\|er\|\gamma$	the scheduling cost (γ)	renting interval length of at most K^r
$1\|\gamma\|er$	renting interval length (er)	scheduling cost at most K^γ
$1\|\|(\gamma, er)$	Pareto-front of γ and er	-
$1\|\|\gamma + \lambda \cdot er$	the sum of γ and $\lambda \cdot er$	-

*: $\lambda \geq 0$ denotes the renting cost per time unit.

Problem	γ	Complexity
$1\|\|\gamma$	$\sum C_j$	$O(n \log n)$ (Smith [4])
	$\sum w_jC_j$	$O(n \log n)$ (Smith [4])
	$\max L_j$	$O(n \log n)$ (Jackson [2])
	$\sum w_jU_j$	$O(nP)$ (Lawler and Moore [3])
$1\|er\|\gamma$	$\sum C_j$	Open
	$\sum w_jC_j$	Open
	$\max L_j$	Open
	$\sum w_jU_j$	Open
$1\|\gamma\|er$	$\sum C_j$	Open
	$\sum w_jC_j$	Open
	$\max L_j$	Open
	$\sum w_jU_j$	Open
$1\|\|(\gamma, er)$	$\sum C_j$	Open
	$\sum w_jC_j$	Open
	$\max L_j$	Open
	$\sum w_jU_j$	Open
$1\|\|\gamma + er$	$\sum C_j$	Open
	$\sum w_jC_j$	Open
	$\max L_j$	Open
	$\sum w_jU_j$	Open

$1|er|\sum w_jC_j$

$1|er|\sum C_j$ is NP-hard

EVEN-ODD-PARTITION: Given integers $a_1, \ldots, a_{2m}$ with $a_{k-1} < a_k$ for $k = 2, \ldots, 2m$ and with total value $2B$, is there a subset of m of these numbers with total value of B such that for each $k = 1, ..., m$ exactly one of the pair $\{a_{2k-1}, a_{2k}\}$ is in the subset?
Given such an instance of EVEN-ODD-PARTITION, we construct an instance of $1|er|\sum C_j$ with $2m + 2$ jobs as follows:

- $J = \{1, \ldots, 2m + 2\}$ and $J^r = \{2m + 1, 2m + 2\}$,
- $p_j = B^2 + a_j$ for each $j = 1, \ldots, 2m$,
- $p_{2m+1} = 0$ and $p_{2m+2} = C + D + 1$, and
- $K^r = p_{2m+2} + mB^2 + B$

where

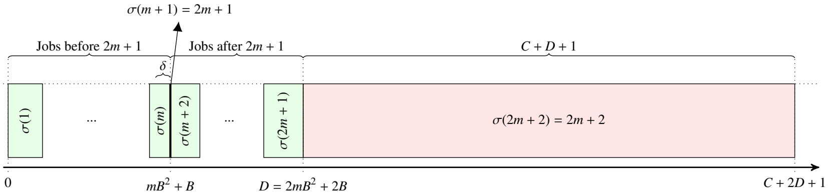
$$C = \sum_{k=1}^{m} (m + 1 - k)(p_{2k-1} + p_{2k}) + (mB^2 + B)(m + 1)$$

and

$$D = \sum_{j=1}^{2m} p_j = 2mB^2 + 2B.$$

Claim: There is a feasible schedule with total completion time of no more than $2(C + D) + 1$ if and only if the answer to the instance of EVEN-ODD-PARTITION is yes.

$1|er|\sum C_j$ is NP-hard



Claim

Job $2m+2$ is the last job in σ and job $2m+1$ is not started before $mB^2 + B$.

Claim

Exactly m jobs are scheduled between $2m+1$ and $2m+2$ in σ.

An example of 8 jobs

$p_1 = 8$	$p_2 = 4$	$p_3 = 3$	$p_4 = 6$	$p_5 = 4$	$p_6 = 6$	$p_7 = 4$	$p_8 = 2$
$w_1 = 16$	$w_2 = 8$	$w_3 = 3$	$w_4 = 24$	$w_5 = 28$	$w_6 = 12$	$w_7 = 12$	$w_8 = 20$

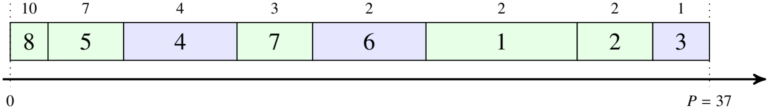
An example of 8 jobs

	1	2	3	4	5	6	7	8
	$p_1 = 8$	$p_2 = 4$	$p_3 = 3$	$p_4 = 6$	$p_5 = 4$	$p_6 = 6$	$p_7 = 4$	$p_8 = 2$
	$w_1 = 16$	$w_2 = 8$	$w_3 = 3$	$w_4 = 24$	$w_5 = 28$	$w_6 = 12$	$w_7 = 12$	$w_8 = 20$
$w_i/p_i =$	2	2	1	4	7	2	3	10

An example of 8 jobs

Job	1	2	3	4	5	6	7	8
p_j	$p_1 = 8$	$p_2 = 4$	$p_3 = 3$	$p_4 = 6$	$p_5 = 4$	$p_6 = 6$	$p_7 = 4$	$p_8 = 2$
w_j	$w_1 = 16$	$w_2 = 8$	$w_3 = 3$	$w_4 = 24$	$w_5 = 28$	$w_6 = 12$	$w_7 = 12$	$w_8 = 20$
$w_i/p_i =$	2	2	1	4	7	2	3	10

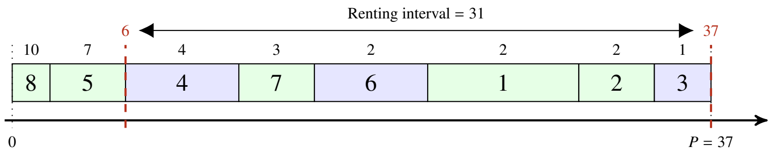
Step 1: Create a sequence according to WSPT↓



An example of 8 jobs

	1	2	3	4	5	6	7	8
p_i	$p_1 = 8$	$p_2 = 4$	$p_3 = 3$	$p_4 = 6$	$p_5 = 4$	$p_6 = 6$	$p_7 = 4$	$p_8 = 2$
w_i	$w_1 = 16$	$w_2 = 8$	$w_3 = 3$	$w_4 = 24$	$w_5 = 28$	$w_6 = 12$	$w_7 = 12$	$w_8 = 20$
$w_i/p_i =$	2	2	1	4	7	2	3	10

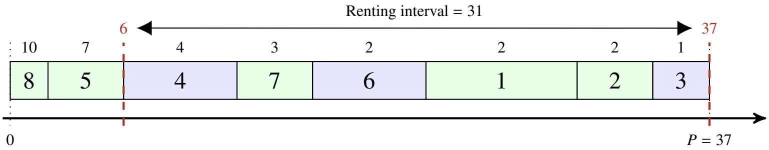
Step 1: Create a sequence according to WSPT↓



An example of 8 jobs

Job	1	2	3	4	5	6	7	8
p_i	$p_1 = 8$	$p_2 = 4$	$p_3 = 3$	$p_4 = 6$	$p_5 = 4$	$p_6 = 6$	$p_7 = 4$	$p_8 = 2$
w_i	$w_1 = 16$	$w_2 = 8$	$w_3 = 3$	$w_4 = 24$	$w_5 = 28$	$w_6 = 12$	$w_7 = 12$	$w_8 = 20$
$w_i/p_i =$	2	2	1	4	7	2	3	10

Step 1: Create a sequence according to WSPT↓



If $K^r \geq 31$, this sequence is optimal.

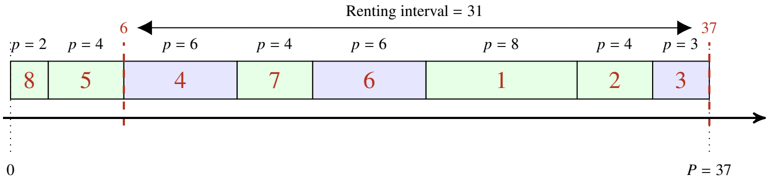
But, what if $K^r < 31$?

What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.

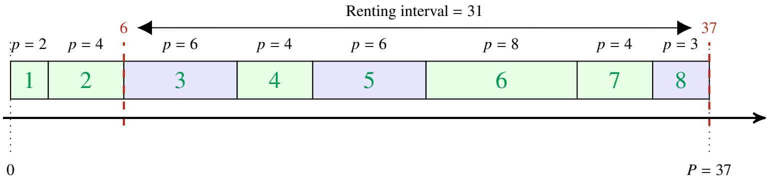
What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.



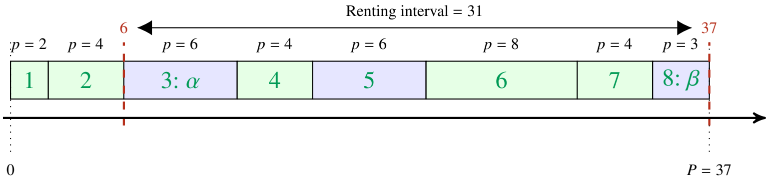
What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.



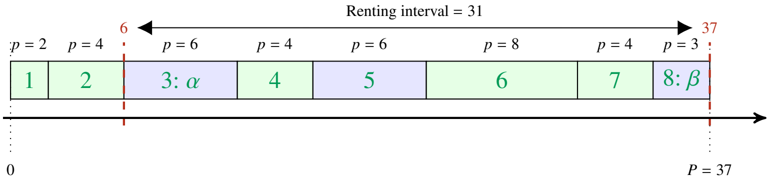
What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.



What if $K^r = 23$?

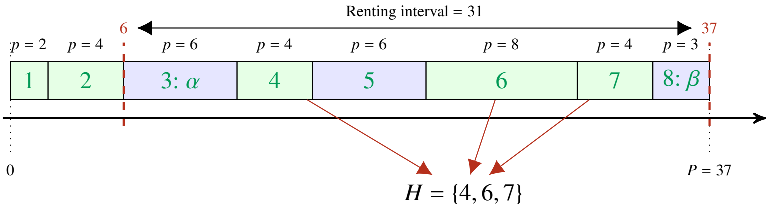
Step 2: Re-number jobs according to WSPT.



Step 3: Form set H of *ordinary jobs* between α and β.

What if $K^r = 23$?

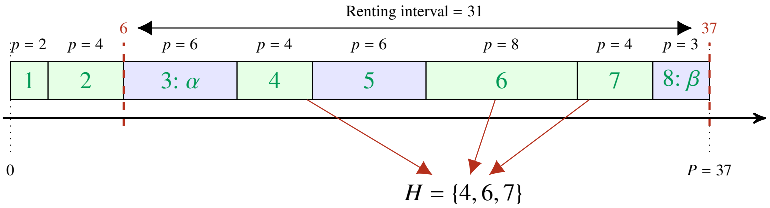
Step 2: Re-number jobs according to WSPT.



Step 3: Form set H of *ordinary jobs* between α and β.

What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.

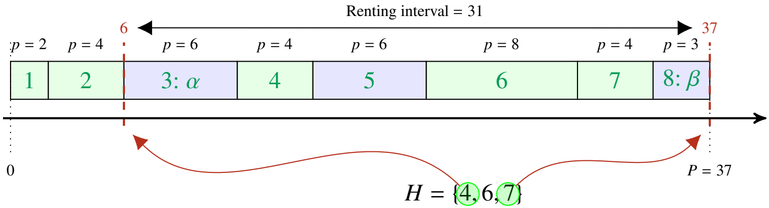


Step 3: Form set H of *ordinary jobs* between α and β.

Step 4: Find a subset $S \subseteq H$ to be moved around such that $er \leq K^r$.

What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.

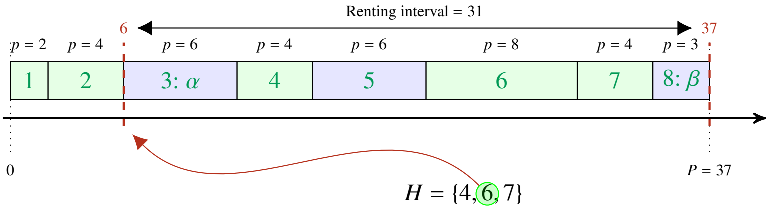


Step 3: Form set H of *ordinary jobs* between α and β.

Step 4: Find a subset $S \subseteq H$ to be moved around such that $er \leq K^r$.

What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.

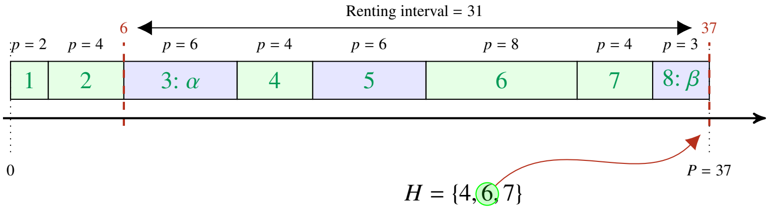


Step 3: Form set H of *ordinary jobs* between α and β.

Step 4: Find a subset $S \subseteq H$ to be moved around such that $er \leq K^r$.

What if $K^r = 23$?

Step 2: Re-number jobs according to WSPT.

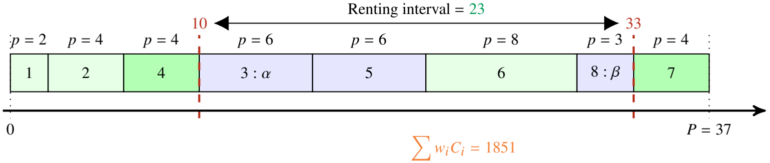


Step 3: Form set H of *ordinary jobs* between α and β.

Step 4: Find a subset $S \subseteq H$ to be moved around such that $er \leq K^r$.

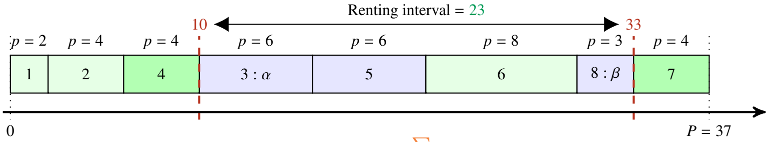
Three different scenarios

Scenario 1:



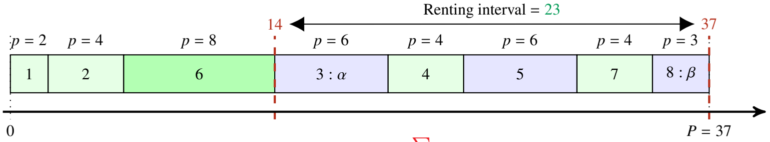
Three different scenarios

Scenario 1:



$\sum w_i C_i = 1851$

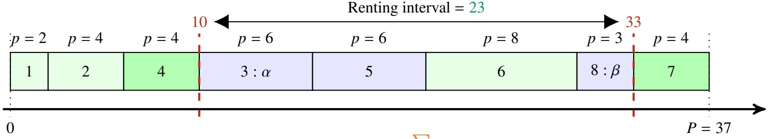
Scenario 2:



$\sum w_i C_i = 1943$

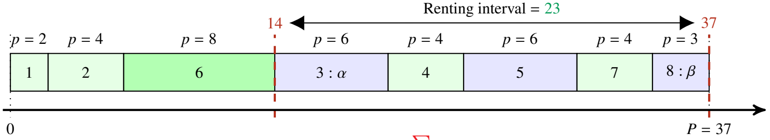
Three different scenarios

Scenario 1:



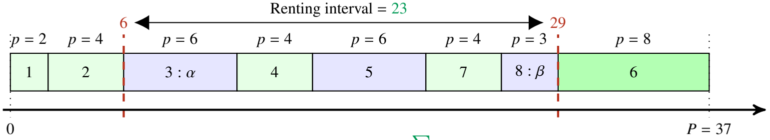
$\sum w_iC_i = 1851$

Scenario 2:



$\sum w_iC_i = 1943$

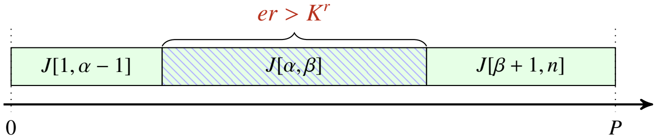
Scenario 3:



$\sum w_iC_i = 1775$

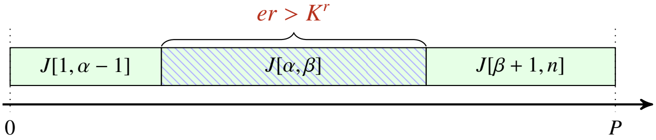
$\sigma \longrightarrow \sigma_{X,Y}$

Given sequence σ (the WSPT sequence)

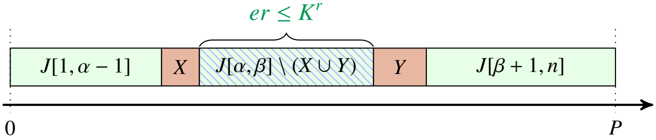


**Remember: Jobs are re-numbered according to WSPT.
**Note: $S = X \cup Y \subseteq H$.

$\sigma \rightarrow \sigma_{X,Y}$



Given sequence σ (the WSPT sequence)



I try to find sequence $\sigma_{X,Y}$

that minimizes $\sum w_iC_i$.

**Remember: Jobs are re-numbered according to WSPT.
**Note: $S = X \cup Y \subseteq H$.

Splitting the Renting interval

Lemma

For each instance of $1|er|\sum w_jC_j$ *there exists* $X^*, Y^* \subseteq H$ *with* $\max X^* < \min Y^*$ *such that* σ_{X^*,Y^*} *is optimal.*

**Remember: Jobs are re-numbered according to WSPT.

Splitting the Renting interval

Lemma

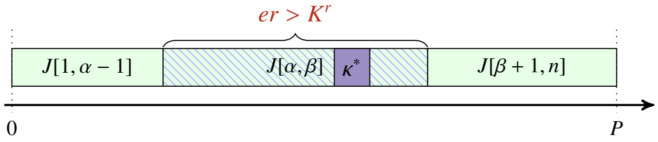
For each instance of $1|er|\sum w_jC_j$ there exists $X^*, Y^* \subseteq H$ with $\max X^* < \min Y^*$ such that σ_{X^*,Y^*} is optimal.

The benefit is that it shows: there is κ^* such that

$$X^* \subseteq H \cap J[\alpha+1, \kappa^*-1]$$

and

$$Y^* \subseteq H \cap J[\kappa^*, \beta-1].$$



**Remember: Jobs are re-numbered according to WSPT.

Splitting the Renting interval

Lemma

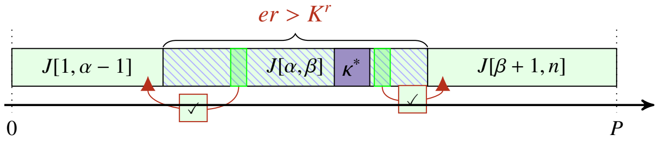
For each instance of $1|er|\sum w_jC_j$ *there exists* $X^*, Y^* \subseteq H$ *with* $\max X^* < \min Y^*$ *such that* σ_{X^*,Y^*} *is optimal.*

The benefit is that it shows: there is κ^* such that

$$X^* \subseteq H \cap J[\alpha+1, \kappa^*-1]$$

and

$$Y^* \subseteq H \cap J[\kappa^*, \beta-1].$$



**Remember: Jobs are re-numbered according to WSPT.

Splitting the Renting interval

Lemma

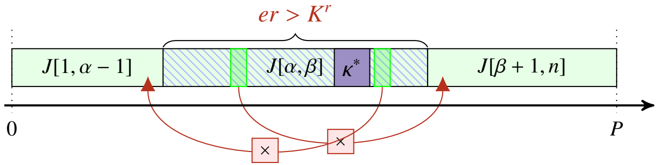
For each instance of $1|er|\sum w_jC_j$ *there exists* $X^*, Y^* \subseteq H$ *with* $\max X^* < \min Y^*$ *such that* σ_{X^*,Y^*} *is optimal.*

The benefit is that it shows: there is κ^* such that

$$X^* \subseteq H \cap J[\alpha+1, \kappa^*-1]$$

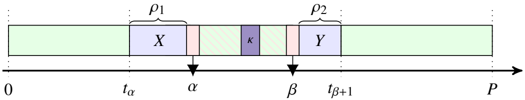
and

$$Y^* \subseteq H \cap J[\kappa^*, \beta-1].$$

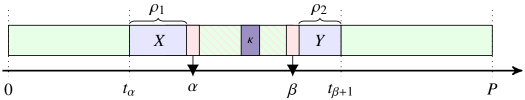


**Remember: Jobs are re-numbered according to WSPT.

Let us construct X^* and Y^*



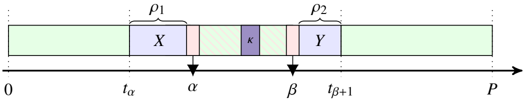
Let us construct X^* and Y^*



For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \to X_{\kappa,\rho_1} \text{ and } Y \to Y_{\kappa,\rho_2}$$

Let us construct X^* and Y^*



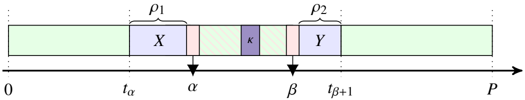
For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \to X_{\kappa,\rho_1} \text{ and } Y \to Y_{\kappa,\rho_2}$$

X_{κ,ρ_1} and Y_{κ,ρ_2} for every tuple (κ, ρ_1, ρ_2)

$$X_{\kappa,\rho_1} \in \arg\min_{X \in \mathcal{X}_{\kappa,\rho_1}} \{f_\kappa(X)\} \quad \text{and} \quad Y_{\kappa,\rho_2} \in \arg\min_{Y \in \mathcal{Y}_{\kappa,\rho_2}} \{g_\kappa(Y)\}.$$

Let us construct X^* and Y^*



For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \rightarrow X_{\kappa,\rho_1} \text{ and } Y \rightarrow Y_{\kappa,\rho_2}$$

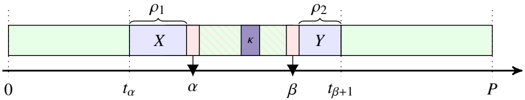
X_{κ,ρ_1} and Y_{κ,ρ_2} for every tuple (κ, ρ_1, ρ_2)

$$X_{\kappa,\rho_1} \in \arg\min_{X \in \mathcal{X}_{\kappa,\rho_1}} \{f_\kappa(X)\} \quad \text{and} \quad Y_{\kappa,\rho_2} \in \arg\min_{Y \in \mathcal{Y}_{\kappa,\rho_2}} \{g_\kappa(Y)\}.$$

We find the optimal triple (κ, ρ_1, ρ_2)

$$(\kappa^*, \rho_1^*, \rho_2^*) \in \arg\min_{(\kappa,\rho_1,\rho_2)|p(J[\alpha,\beta])-\rho_1-\rho_2 \leq K^r} \{f_\kappa(X_{\kappa,\rho_1}) + g_\kappa(Y_{\kappa,\rho_2})\}$$

Let us construct X^* and Y^*



For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \to X_{\kappa,\rho_1} \text{ and } Y \to Y_{\kappa,\rho_2}$$

X_{κ,ρ_1} and Y_{κ,ρ_2} for every tuple (κ, ρ_1, ρ_2)

$$X_{\kappa,\rho_1} \in \arg\min_{X \in \mathcal{X}_{\kappa,\rho_1}} \{f_\kappa(X)\} \quad \text{and} \quad Y_{\kappa,\rho_2} \in \arg\min_{Y \in \mathcal{Y}_{\kappa,\rho_2}} \{g_\kappa(Y)\}.$$

Done in $O(nP^2)$ → next slide

We find the optimal triple (κ, ρ_1, ρ_2)

$$(\kappa^*, \rho_1^*, \rho_2^*) \in \arg\min_{(\kappa,\rho_1,\rho_2)|p(J[\alpha,\beta])-\rho_1-\rho_2 \le K^r} \{f_\kappa(X_{\kappa,\rho_1}) + g_\kappa(Y_{\kappa,\rho_2})\}$$

Done in $O(nP)$.

Dynamic programming for computing $X_{\kappa,\rho}$

Computing $X_{\kappa,\rho}$

$$\theta_{1,\rho}(\alpha-1,\varrho) = \begin{cases} 0 & \text{if } \varrho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_{1,\rho}(j,\varrho) = \min\left\{ \begin{array}{l} \begin{cases} \theta_{1,\rho}(j-1,\varrho-p_j) + w_j\cdot(t_\alpha+\varrho) & \text{if } j\in J^o \\ \infty & \text{if } j \in J^r \end{cases} \\ \theta_{1,\rho}(j-1,\varrho) + w_j\cdot(p(J[1,j]) + \rho - \varrho) \end{array} \right\}.$$

Dynamic programming for computing $X_{\kappa,\rho}$

Computing $X_{\kappa,\rho}$ (Similarly for $Y_{\kappa,\rho}$)

$$\theta_{1,\rho}(\alpha-1,\varrho) = \begin{cases} 0 & \text{if } \varrho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_{1,\rho}(j,\varrho) = \min\left\{ \begin{array}{l} \begin{cases} \theta_{1,\rho}(j-1,\varrho-p_j) + w_j \cdot (t_\alpha + \varrho) & \text{if } j \in J^o \\ \infty & \text{if } j \in J^r \end{cases} \\ \theta_{1,\rho}(j-1,\varrho) + w_j \cdot (p(J[1,j]) + \rho - \varrho) \end{array} \right\}.$$

Done in $O(nP)$.

Dynamic programming for computing $X_{\kappa,\rho}$

Computing $X_{\kappa,\rho}$ (Similarly for $Y_{\kappa,\rho}$)

$$\theta_{1,\rho}(\alpha-1,\varrho) = \begin{cases} 0 & \text{if } \varrho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_{1,\rho}(j,\varrho) = \min\left\{ \begin{array}{l} \begin{cases} \theta_{1,\rho}(j-1,\varrho-p_j) + w_j\cdot(t_\alpha+\varrho) & \text{if } j\in J^o \\ \infty & \text{if } j \in J^r \end{cases} \\ \theta_{1,\rho}(j-1,\varrho) + w_j\cdot(p(J[1,j])+\rho-\varrho) \end{array} \right\}.$$

Done in $O(nP)$.

So for all (κ,ρ), the algorithm runs in $O(n^2P^2)$?

Dynamic programming for computing $X_{\kappa,\rho}$

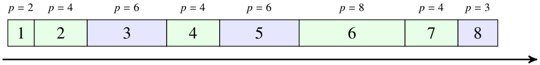
Computing $X_{\kappa,\rho}$ (Similarly for $Y_{\kappa,\rho}$)

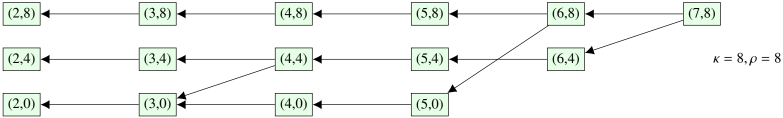
$$\theta_{1,\rho}(\alpha-1,\varrho) = \begin{cases} 0 & \text{if } \varrho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_{1,\rho}(j,\varrho) = \min \left\{ \begin{array}{l} \begin{cases} \theta_{1,\rho}(j-1,\varrho-p_j) + w_j \cdot (t_\alpha + \varrho) & \text{if } j \in J^o \\ \infty & \text{if } j \in J^r \end{cases} \\ \theta_{1,\rho}(j-1,\varrho) + w_j \cdot (p(J[1,j]) + \rho - \varrho) \end{array} \right\}.$$

Done in $O(nP)$.

So for all (κ,ρ), the algorithm runs in $O(n^2P^2)$?





Dynamic programming for computing $X_{\kappa,\rho}$

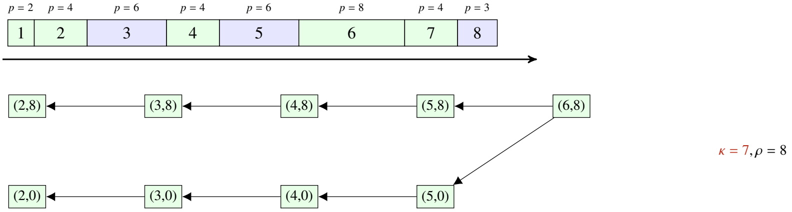
Computing $X_{\kappa,\rho}$ (Similarly for $Y_{\kappa,\rho}$)

$$\theta_{1,\rho}(\alpha-1,\varrho) = \begin{cases} 0 & \text{if } \varrho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_{1,\rho}(j,\varrho) = \min \left\{ \begin{array}{l} \begin{cases} \theta_{1,\rho}(j-1,\varrho-p_j) + w_j \cdot (t_\alpha + \varrho) & \text{if } j \in J^o \\ \infty & \text{if } j \in J^r \end{cases} \\ \theta_{1,\rho}(j-1,\varrho) + w_j \cdot (p(J[1,j]) + \rho - \varrho) \end{array} \right\}.$$

Done in $O(nP)$.

So for all (κ,ρ), the algorithm runs in $O(n^2P^2)$?



Dynamic programming for computing $X_{\kappa,\rho}$

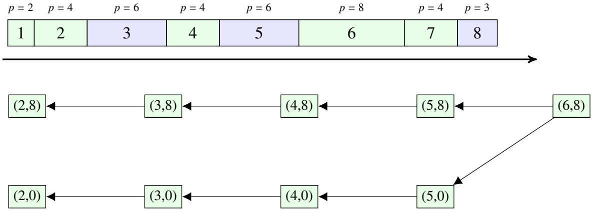
Computing $X_{\kappa,\rho}$ (Similarly for $Y_{\kappa,\rho}$)

$$\theta_{1,\rho}(\alpha-1,\varrho) = \begin{cases} 0 & \text{if } \varrho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_{1,\rho}(j,\varrho) = \min \left\{ \begin{array}{l} \begin{cases} \theta_{1,\rho}(j-1,\varrho-p_j) + w_j \cdot (t_\alpha + \varrho) & \text{if } j \in J^o \\ \infty & \text{if } j \in J^r \end{cases} \\ \theta_{1,\rho}(j-1,\varrho) + w_j \cdot (p(J[1,j]) + \rho - \varrho) \end{array} \right\}.$$

Done in $O(nP)$.

So for all (κ,ρ), the algorithm runs in $O(n^2P^2)$? Done in $O(nP^2)$.



$\kappa = 7, \rho = 8$

Final Result for $1|er|\sum w_jC_j$ and $1|er|\sum C_j$

Lemma

$1|er|\sum w_jC_j$ can be solved in $O(nP^2)$-time.

Final Result for $1|er|\sum w_jC_j$ and $1|er|\sum C_j$

Lemma

$1|er|\sum w_jC_j$ *can be solved in* $O(nP^2)$*-time.*

Lemma

$1|er|\sum w_jC_j$ *can be solved in* $O(nPW)$*-time.*

Final Result for $1|er|\sum w_jC_j$ and $1|er|\sum C_j$

Lemma

$1|er|\sum w_jC_j$ *can be solved in* $O(nP^2)$*-time.*

Lemma

$1|er|\sum w_jC_j$ *can be solved in* $O(nPW)$*-time.*

Theorem

$1|er|\sum w_jC_j$ *can be solved in* $O(nP\min\{P, W\})$*-time.*

Final Result for $1|er|\sum w_jC_j$ and $1|er|\sum C_j$

Lemma

$1|er|\sum w_jC_j$ *can be solved in* $O(nP^2)$*-time.*

Lemma

$1|er|\sum w_jC_j$ *can be solved in* $O(nPW)$*-time.*

Theorem

$1|er|\sum w_jC_j$ *can be solved in* $O(nP\min\{P, W\})$*-time.*

Corollary

$1|er|\sum C_j$ *can be solved in* $O(n^2P)$*-time.*

$1||\sum w_jC_j + \lambda \cdot er$

$1||\sum w_jC_j + \lambda \cdot er$

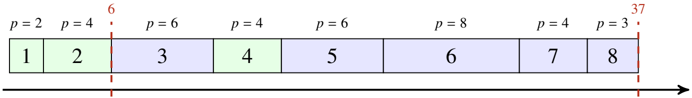
The objective

We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.

$1||\sum w_jC_j + \lambda \cdot er$

The objective

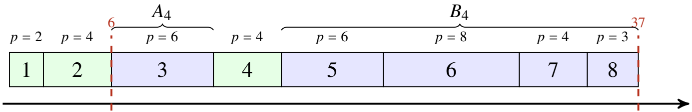
We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.



$1||\sum w_jC_j + \lambda \cdot er$

The objective

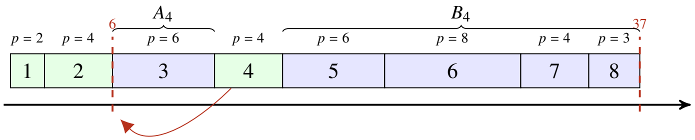
We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.



$1||\sum w_jC_j + \lambda \cdot er$

The objective

We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.

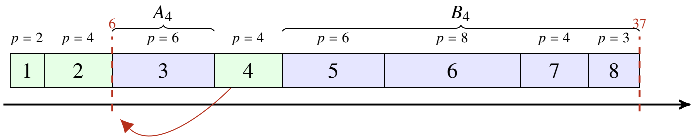


$$\underbrace{w(A_4)p_4 - \lambda \cdot p_4}_{\text{moving } A_4 \text{ forward}} - \underbrace{w_4 P(A_4)}_{\text{moving 4 backward}} < 0$$

$1||\sum w_jC_j + \lambda \cdot er$

The objective

We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.

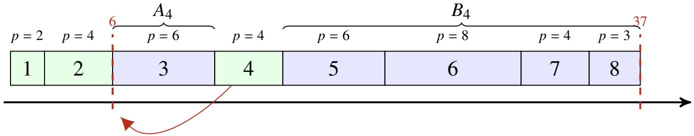


$$\underbrace{w(A_4)p_4 - \lambda \cdot p_4}_{\text{moving } A_4 \text{ forward}} - \underbrace{w_4P(A_4)}_{\text{moving 4 backward}} < 0 \quad \rightarrow \quad \frac{w_4}{p_4} > \frac{w(A_4) - \lambda}{p(A_4)}$$

$1||\sum w_jC_j + \lambda \cdot er$

The objective

We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.



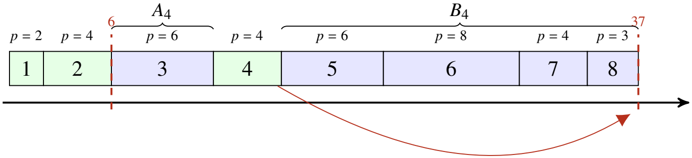
$$\underbrace{w(A_4)p_4 - \lambda \cdot p_4}_{\text{moving } A_4 \text{ forward}} - \underbrace{w_4P(A_4)}_{\text{moving 4 backward}} < 0 \quad \rightarrow \quad \frac{w_4}{p_4} > \frac{w(A_4) - \lambda}{p(A_4)}$$

$$\rightarrow \quad \frac{w_4}{p_4} \geq \frac{w(J^r)}{p(J^r)}$$

$1||\sum w_jC_j + \lambda \cdot er$

The objective

We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.

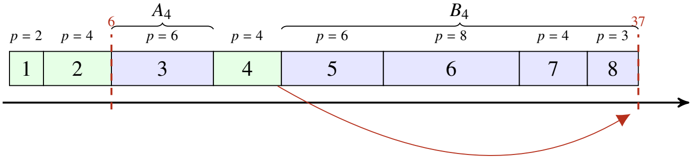


$$\underbrace{-w(B_4)p_4 - \lambda \cdot p_4}_{\text{moving } B_4 \text{ backward}} + \underbrace{w_4P(B_4)}_{\text{moving 4 forward}} < 0$$

$1||\sum w_jC_j + \lambda \cdot er$

The objective

We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.

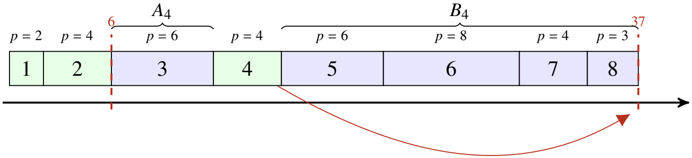


$$\underbrace{-w(B_4)p_4 - \lambda \cdot p_4}_{\text{moving } B_4 \text{ backward}} + \underbrace{w_4 P(B_4)}_{\text{moving 4 forward}} < 0 \quad \rightarrow \quad \frac{w_4}{p_4} < \frac{w(B_4) + \lambda}{p(B_4)}$$

$1||\sum w_jC_j + \lambda \cdot er$

The objective

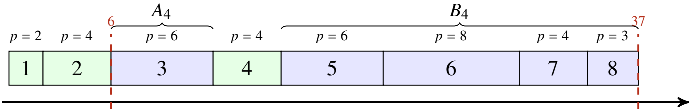
We want to minimize the sum $\sum w_jC_j + \lambda \cdot er$.



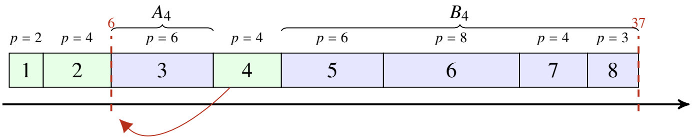
$$\underbrace{-w(B_4)p_4 - \lambda \cdot p_4}_{\text{moving } B_4 \text{ backward}} + \underbrace{w_4 P(B_4)}_{\text{moving 4 forward}} < 0 \quad \rightarrow \quad \frac{w_4}{p_4} < \frac{w(B_4) + \lambda}{p(B_4)}$$

$$\rightarrow \quad \frac{w_4}{p_4} < \frac{w(J^r)}{p(J^r)}$$

$1||\sum w_jC_j + \lambda \cdot er$ for job 4



$1||\sum w_jC_j + \lambda \cdot er$ for job 4

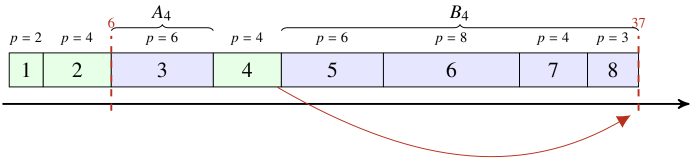


So, if

$$\frac{w_4}{p_4} > \frac{w(A_4) - \lambda}{p(A_4)} \quad \text{and} \quad \frac{w_4}{p_4} \geq \frac{w(J^r)}{p(J^r)},$$

then 4 is moved before the Renting interval ($\rightarrow X$).

$1||\sum w_jC_j + \lambda \cdot er$ for job 4



So, if

$$\frac{w_4}{p_4} > \frac{w(A_4) - \lambda}{p(A_4)} \quad \text{and} \quad \frac{w_4}{p_4} \geq \frac{w(J^r)}{p(J^r)},$$

then 4 is moved before the Renting interval ($\rightarrow X$).

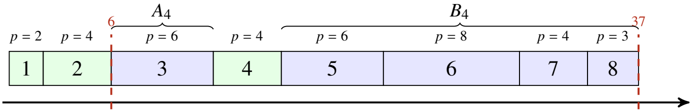
---

and if

$$\frac{w_4}{p_4} < \frac{w(B_4) + \lambda}{p(B_4)} \quad \text{and} \quad \frac{w_4}{p_4} < \frac{w(J^r)}{p(J^r)},$$

then 4 is moved after the Renting interval ($\rightarrow Y$).

$1||\sum w_jC_j + \lambda \cdot er$ for job 4



So, if

$$\frac{w_4}{p_4} > \frac{w(A_4) - \lambda}{p(A_4)} \quad \text{and} \quad \frac{w_4}{p_4} \geq \frac{w(J^r)}{p(J^r)},$$

then 4 is moved before the Renting interval ($\rightarrow X$).

---

and if

$$\frac{w_4}{p_4} < \frac{w(B_4) + \lambda}{p(B_4)} \quad \text{and} \quad \frac{w_4}{p_4} < \frac{w(J^r)}{p(J^r)},$$

then 4 is moved after the Renting interval ($\rightarrow Y$).

Otherwise 4 stays inside the Renting interval.

$1||\sum w_jC_j + \lambda \cdot er$

$$X_\lambda := \left\{ j \in H : \frac{w_j}{p_j} > \frac{w(A_j) - \lambda}{p(A_j)} \text{ and } \frac{w_j}{p_j} \geq \frac{w(J^r)}{p(J^r)} \right\},$$

$$Y_\lambda := \left\{ j \in H : \frac{w_j}{p_j} < \frac{w(B_j) + \lambda}{p(B_j)} \text{ and } \frac{w_j}{p_j} < \frac{w(J^r)}{p(J^r)} \right\}.$$

$1||\sum w_jC_j + \lambda \cdot er$

$$X_\lambda := \left\{ j \in H : \frac{w_j}{p_j} > \frac{w(A_j) - \lambda}{p(A_j)} \text{ and } \frac{w_j}{p_j} \geq \frac{w(J^r)}{p(J^r)} \right\},$$

$$Y_\lambda := \left\{ j \in H : \frac{w_j}{p_j} < \frac{w(B_j) + \lambda}{p(B_j)} \text{ and } \frac{w_j}{p_j} < \frac{w(J^r)}{p(J^r)} \right\}.$$

Lemma

$\sigma_{X_\lambda,Y_\lambda}$ is an optimal sequence for $1||\sum w_jC_j + er$ with unit rental cost λ.

$1||\sum w_jC_j + \lambda \cdot er$

$$X_\lambda := \left\{ j \in H : \frac{w_j}{p_j} > \frac{w(A_j) - \lambda}{p(A_j)} \text{ and } \frac{w_j}{p_j} \geq \frac{w(J^r)}{p(J^r)} \right\},$$
$$Y_\lambda := \left\{ j \in H : \frac{w_j}{p_j} < \frac{w(B_j) + \lambda}{p(B_j)} \text{ and } \frac{w_j}{p_j} < \frac{w(J^r)}{p(J^r)} \right\}.$$

Lemma

$\sigma_{X_\lambda,Y_\lambda}$ is an optimal sequence for $1||\sum w_jC_j + er$ with unit rental cost λ.

Theorem

$1||\sum w_jC_j + \lambda \cdot er$ can be solved in time $O(n \log n)$.

$1|er|\max L_j$

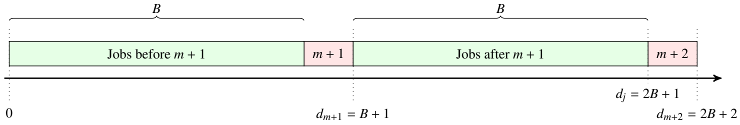
$1|er|\max L_j$ is NP-hard

PARTITION: Given integer numbers $a_1, \ldots, a_m$, is there a subset of $\{a_1, \ldots, a_m\}$ with total value of $B = \frac{1}{2}\sum_{i=1}^{m} a_i$?

Given an instance of PARTITION, we construct an instance of $1|er|\max L_j$ with $m+2$ jobs as follows:

- $J = \{1, \ldots, m+2\}$ and $J^r = \{m+1, m+2\}$,
- $p_j = a_j$ and $d_j = 2B + 1$ for each $j = 1, \ldots, m$,
- $p_{m+1} = 1$, $d_{m+1} = B + 1$, $p_{m+2} = 1$ and $d_{m+2} = 2B + 2$, and
- $K^r = B + 2$.

$1|er|\max L_j$ is NP-hard

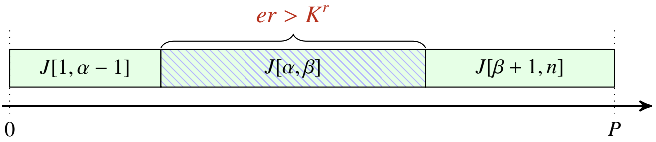


Claim

There is a feasible schedule with maximum lateness of at most zero if and only if the answer to the instance of PARTITION *is yes.*

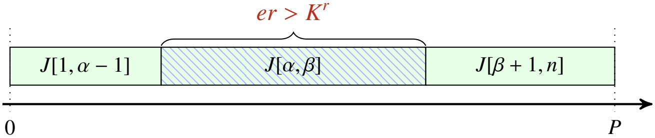
$\sigma \rightarrow \sigma_{X,Y}$

Given sequence σ (the EDD sequence)

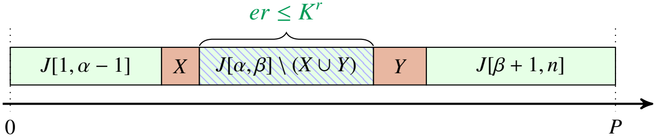


**Remember: Jobs are re-numbered according to EDD.
**Note: $S = X \cup Y \subseteq H$.

$\sigma \to \sigma_{X,Y}$



Given sequence σ (the EDD sequence)



I try to find sequence $\sigma_{X,Y}$

that minimizes $\max L_j$.

**Remember: Jobs are re-numbered according to EDD.
**Note: $S = X \cup Y \subseteq H$.

Splitting the Renting interval

Lemma

For each instance of $1|er|\max L_j$ *there exists* $X^*, Y^* \subseteq H$ *with* $\max X^* < \min Y^*$ *such that* σ_{X^*,Y^*} *is optimal.*

**Remember: Jobs are re-numbered according to EDD.

Splitting the Renting interval

Lemma

For each instance of $1|er|\max L_j$ *there exists* $X^*, Y^* \subseteq H$ *with* $\max X^* < \min Y^*$ *such that* σ_{X^*,Y^*} *is optimal.*

The benefit is that it shows: there is κ^* such that

$$X^* \subseteq H \cap J[\alpha + 1, \kappa^* - 1]$$

and

$$Y^* \subseteq H \cap J[\kappa^*, \beta - 1].$$

**Remember: Jobs are re-numbered according to EDD.

Splitting the Renting interval

Lemma

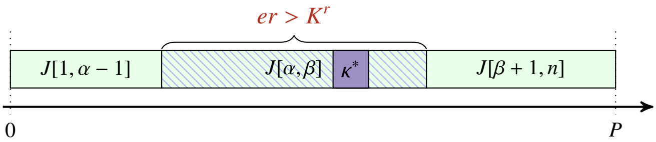
For each instance of $1|er|\max L_j$ there exists $X^*, Y^* \subseteq H$ with $\max X^* < \min Y^*$ such that σ_{X^*,Y^*} is optimal.

The benefit is that it shows: there is κ^* such that

$$X^* \subseteq H \cap J[\alpha+1, \kappa^* - 1]$$

and

$$Y^* \subseteq H \cap J[\kappa^*, \beta - 1].$$



**Remember: Jobs are re-numbered according to EDD.

Splitting the Renting interval

Lemma

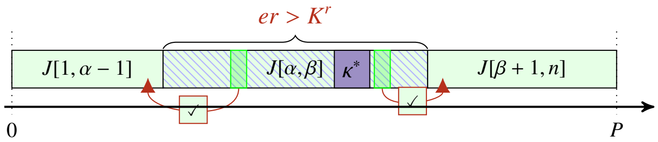
For each instance of $1|er|\max L_j$ *there exists* $X^*, Y^* \subseteq H$ *with* $\max X^* < \min Y^*$ *such that* σ_{X^*,Y^*} *is optimal.*

The benefit is that it shows: there is κ^* such that

$$X^* \subseteq H \cap J[\alpha + 1, \kappa^* - 1]$$

and

$$Y^* \subseteq H \cap J[\kappa^*, \beta - 1].$$



**Remember: Jobs are re-numbered according to EDD.

Splitting the Renting interval

Lemma

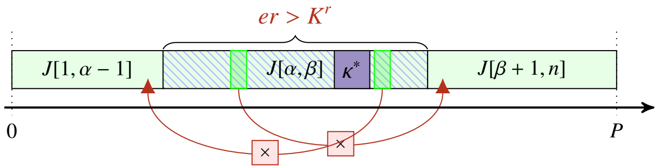
For each instance of $1|er|\max L_j$ *there exists* $X^*, Y^* \subseteq H$ *with* $\max X^* < \min Y^*$ *such that* σ_{X^*,Y^*} *is optimal.*

The benefit is that it shows: there is κ^* such that

$$X^* \subseteq H \cap J[\alpha + 1, \kappa^* - 1]$$

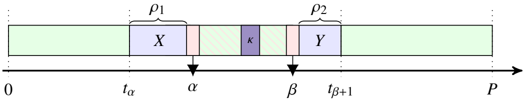
and

$$Y^* \subseteq H \cap J[\kappa^*, \beta - 1].$$

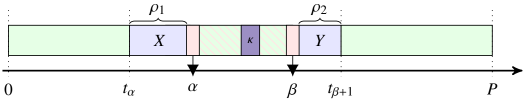


**Remember: Jobs are re-numbered according to EDD.

Let us construct X^* and Y^*



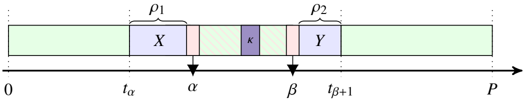
Let us construct X^* and Y^*



For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \to X_{\kappa,\rho_1} \text{ and } Y \to Y_{\kappa,\rho_2}$$

Let us construct X^* and Y^*



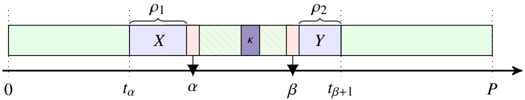
For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \to X_{\kappa,\rho_1} \text{ and } Y \to Y_{\kappa,\rho_2}$$

X_{κ,ρ_1} and Y_{κ,ρ_2} for every tuple (κ, ρ_1, ρ_2)

$$X_{\kappa,\rho_1} \in \arg\min_{X \in \mathcal{X}_{\kappa,\rho_1}} \{f'_\kappa(X)\} \quad \text{and} \quad Y_{\kappa,\rho_2} \in \arg\min_{Y \in \mathcal{Y}_{\kappa,\rho_2}} \{g'_\kappa(Y)\}.$$

Let us construct X^* and Y^*



For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \to X_{\kappa,\rho_1} \text{ and } Y \to Y_{\kappa,\rho_2}$$

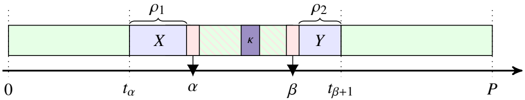
X_{κ,ρ_1} and Y_{κ,ρ_2} for every tuple (κ, ρ_1, ρ_2)

$$X_{\kappa,\rho_1} \in \arg\min_{X \in \mathcal{X}_{\kappa,\rho_1}} \{f'_\kappa(X)\} \quad \text{and} \quad Y_{\kappa,\rho_2} \in \arg\min_{Y \in \mathcal{Y}_{\kappa,\rho_2}} \{g'_\kappa(Y)\}.$$

We find the optimal triple (κ, ρ_1, ρ_2)

$$(\kappa^*, \rho_1^*, \rho_2^*) \in \arg\min_{(\kappa,\rho_1,\rho_2) | p(J[\alpha,\beta]) - \rho_1 - \rho_2 \le K^r} \{\max\{f'_\kappa(X_{\kappa,\rho_1}), g'_\kappa(Y_{\kappa,\rho_2})\}\}$$

Let us construct X^* and Y^*



For any given trio of values κ, ρ_1, and ρ_2, we want to find the best X and Y. And then

$$X \rightarrow X_{\kappa,\rho_1} \text{ and } Y \rightarrow Y_{\kappa,\rho_2}$$

X_{κ,ρ_1} and Y_{κ,ρ_2} for every tuple (κ, ρ_1, ρ_2)

$$X_{\kappa,\rho_1} \in \arg\min_{X \in \mathcal{X}_{\kappa,\rho_1}} \{f'_\kappa(X)\} \quad \text{and} \quad Y_{\kappa,\rho_2} \in \arg\min_{Y \in \mathcal{Y}_{\kappa,\rho_2}} \{g'_\kappa(Y)\}.$$

Done in $O(nP)$ → next slide

We find the optimal triple (κ, ρ_1, ρ_2)

$$(\kappa^*, \rho_1^*, \rho_2^*) \in \arg\min_{(\kappa,\rho_1,\rho_2) | p(J[\alpha,\beta]) - \rho_1 - \rho_2 \leq K^r} \{\max\{f'_\kappa(X_{\kappa,\rho_1}), g'_\kappa(Y_{\kappa,\rho_2})\}\}$$

Done in $O(nP)$.

Dynamic programming for computing $X_{\kappa,\rho}$

Accelerated jobs do not have maximum lateness

Jobs in X do not imply maximum lateness since $d_\alpha \leq d_j$ and $C_\alpha \geq C_j$ for each $j \in X$.

Computing $X_{\kappa,\rho}$ (Similarly for $Y_{\kappa,\rho}$)

$$\theta_3(\alpha+1,\rho) = \begin{cases} p(J[1,\alpha]) - d_\alpha & \text{if } \rho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_3(\kappa+1,\rho) = \min \left\{ \begin{array}{l} \max \left\{ \begin{array}{l} \theta_3(\kappa,\rho), \\ p(J[1,\kappa]) - d_\kappa \end{array} \right\}, \\ \begin{cases} \theta_3(\kappa,\rho-p_\kappa) + p_\kappa & \text{if } \kappa \in J^o \\ \infty & \text{if } \kappa \in J^r \end{cases} \end{array} \right\}.$$

Dynamic programming for computing $X_{\kappa,\rho}$

Accelerated jobs do not have maximum lateness

Jobs in X do not imply maximum lateness since $d_\alpha \leq d_j$ and $C_\alpha \geq C_j$ for each $j \in X$.

Computing $X_{\kappa,\rho}$ (Similarly for $Y_{\kappa,\rho}$)

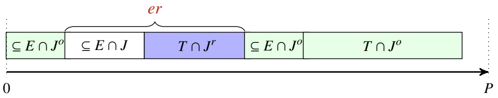
$$\theta_3(\alpha+1,\rho) = \begin{cases} p(J[1,\alpha]) - d_\alpha & \text{if } \rho = 0 \\ \infty & \text{otherwise} \end{cases},$$

$$\theta_3(\kappa+1,\rho) = \min\left\{ \begin{array}{l} \max\left\{ \begin{array}{l} \theta_3(\kappa,\rho), \\ p(J[1,\kappa]) - d_\kappa \end{array} \right\}, \\ \begin{cases} \theta_3(\kappa,\rho-p_\kappa) + p_\kappa & \text{if } \kappa \in J^o \\ \infty & \text{if } \kappa \in J^r \end{cases} \end{array} \right\}.$$

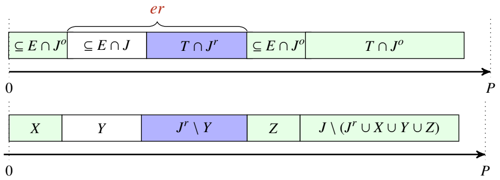
Done in $O(nP)$.

$1|er|\sum w_jU_j$

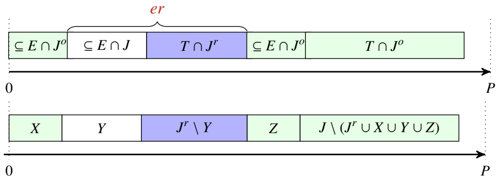
Structure of optimal schedules



Structure of optimal schedules



Structure of optimal schedules

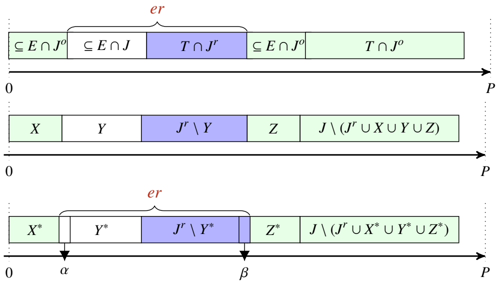


Lemma

For each instance of $1|er|\sum w_jU_j$ *there exist disjoint sets* $X^*, Y^*, Z^* \subseteq J$ *such that the sequence* σ_{X^*,Y^*,Z^*} *is optimal and, moreover,*

1. *all jobs in* $X^* \cup Y^* \cup Z^*$ *are non-tardy in* σ_{X^*,Y^*,Z^*}*,*
2. $(X^* \cup Z^*) \cap J^r = \emptyset$*, and*
3. $\max(X^* \cup (Y^* \cap J^o)) < \kappa \leq \min Z^*$.

Structure of optimal schedules

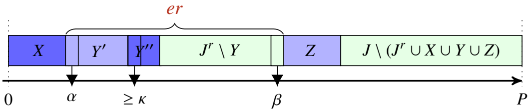


Lemma

For each instance of $1|er|\sum w_jU_j$ *there exist disjoint sets* $X^*, Y^*, Z^* \subseteq J$ *such that the sequence* σ_{X^*,Y^*,Z^*} *is optimal and, moreover,*

1. *all jobs in* $X^* \cup Y^* \cup Z^*$ *are non-tardy in* σ_{X^*,Y^*,Z^*}*,*
2. $(X^* \cup Z^*) \cap J^r = \emptyset$*, and*
3. $\max(X^* \cup (Y^* \cap J^o)) < \kappa \leq \min Z^*$.

Let us construct X^*, Y^*, and Z^*



$\mathcal{X}_{\kappa,\rho_1,\rho_2}$, $\mathcal{Y}''_{\kappa,\rho_1}$, and $\mathcal{Z}_{\kappa,\rho_2}$ for every tuple (κ, ρ_1, ρ_2)

$$\mathcal{X}_{\kappa,\rho_1,\rho_2} = \left\{ (X, Y') \;\middle|\; \begin{array}{l} X, Y' \subseteq J[1, \kappa - 1],\ X \cap Y' = \emptyset = X \cap J^r, \\ \sum_{j' \in X \cup Y' : j' \leq j} p_{j'} \leq d_j, \forall j \in X \cup Y', \\ p(Y' \cap J^o) \leq K^r - p(J^r), \\ p(X \cup Y') = \rho_1,\ \rho_1 + p(J^r \setminus Y') = \rho_2 \end{array} \right\}$$

$$\mathcal{Y}''_{\kappa,\rho_1} = \left\{ Y'' \;\middle|\; Y'' \subseteq J^r[\kappa, n],\ \rho_1 + \sum_{j' \in Y'' : j' \leq j} p_{j'} \leq d_j, \forall j \in Y'' \right\}$$

$$\mathcal{Z}_{\kappa,\rho_2} = \left\{ Z \;\middle|\; Z \subseteq J^o[\kappa, n],\ \rho_2 + \sum_{j' \in Z : j' \leq j} p_{j'} \leq d_j, \forall j \in Z \right\}$$

Let us construct X^*, Y^*, and Z^*

Optimal job sets for given triple $(\kappa, \rho_1, \rho_2) \in \Xi$

$$(X_{\kappa,\rho_1,\rho_2}, Y'_{\kappa,\rho_1,\rho_2}) \in \underset{(X,Y')\in\mathcal{X}_{\kappa,\rho_1,\rho_2}}{\arg\max}\; w(X) + w(Y'),$$

$$Y''_{\kappa,\rho_1} \in \underset{Y''\in\mathcal{Y}''_{\kappa,\rho_1}}{\arg\max}\, w(Y''), \text{ and}$$

$$Z_{\kappa,\rho_2} \in \underset{Z\in\mathcal{Z}_{\kappa,\rho_2}}{\arg\max}\, w(Z)$$

Let us construct X^*, Y^*, and Z^*

Optimal job sets for given triple $(\kappa, \rho_1, \rho_2) \in \Xi$

$$(X_{\kappa,\rho_1,\rho_2}, Y'_{\kappa,\rho_1,\rho_2}) \in \operatorname*{arg\,max}_{(X,Y') \in \mathcal{X}_{\kappa,\rho_1,\rho_2}} w(X) + w(Y'),$$

$$Y''_{\kappa,\rho_1} \in \operatorname*{arg\,max}_{Y'' \in \mathcal{Y}''_{\kappa,\rho_1}} w(Y''), \text{ and}$$

$$Z_{\kappa,\rho_2} \in \operatorname*{arg\,max}_{Z \in \mathcal{Z}_{\kappa,\rho_2}} w(Z)$$

We find the optimal triple (κ, ρ_1, ρ_2)

$$(\kappa^*, \rho_1^*, \rho_2^*) \in \operatorname*{arg\,max}_{(\kappa,\rho_1,\rho_2) \in \Xi} \{w(X_{\kappa,\rho_1,\rho_2}) + w(Y'_{\kappa,\rho_1,\rho_2}) + w(Y''_{\kappa,\rho_1}) + w(Z_{\kappa,\rho_2})\}$$

Let us construct X^*, Y^*, and Z^*

Optimal job sets for given triple $(\kappa, \rho_1, \rho_2) \in \Xi$

$$
\begin{aligned}
(X_{\kappa,\rho_1,\rho_2}, Y'_{\kappa,\rho_1,\rho_2}) &\in \underset{(X,Y')\in\mathcal{X}_{\kappa,\rho_1,\rho_2}}{\arg\max}\ w(X) + w(Y'), \\
Y''_{\kappa,\rho_1} &\in \underset{Y''\in\mathcal{Y}''_{\kappa,\rho_1}}{\arg\max}\ w(Y''), \text{ and} \\
Z_{\kappa,\rho_2} &\in \underset{Z\in\mathcal{Z}_{\kappa,\rho_2}}{\arg\max}\ w(Z)
\end{aligned}
$$

We find the optimal triple (κ, ρ_1, ρ_2)

$$
(\kappa^*, \rho_1^*, \rho_2^*) \in \underset{(\kappa,\rho_1,\rho_2)\in\Xi}{\arg\max} \{w(X_{\kappa,\rho_1,\rho_2}) + w(Y'_{\kappa,\rho_1,\rho_2}) + w(Y''_{\kappa,\rho_1}) + w(Z_{\kappa,\rho_2})\}
$$

Done in $O(nP^4)$.

Results

Problem	γ	Complexity
$1\|\|\gamma$	$\sum C_j$	$O(n \log n)$ (Smith [4])
	$\sum w_jC_j$	$O(n \log n)$ (Smith [4])
	$\max L_j$	$O(n \log n)$ (Jackson [2])
	$\sum w_jU_j$	$O(nP)$ (Lawler and Moore [3])
$1\|er\|\gamma$	$\sum C_j$	$O(n^2 P)$
	$\sum w_jC_j$	$O(nP\max(P, W))$
	$\max L_j$	$O(nP)$
	$\sum w_jU_j$	$O(nP^4)$
$1\|\gamma\|er$	$\sum C_j$	$O(n^2 P)$
	$\sum w_jC_j$	$O(nP\max(P, W))$
	$\max L_j$	$O(nP)$
	$\sum w_jU_j$	$O(nP^4 \log(P))$
$1\|\|(\gamma, er)$	$\sum C_j$	$O(nP^2)$
	$\sum w_jC_j$	$O(nP^2)$
	$\max L_j$	$O(nP^2)$
	$\sum w_jU_j$	$O(nP^5)$
$1\|\|\gamma + er$	$\sum C_j$	$O(n \log n)$
	$\sum w_jC_j$	$O(n \log n)$
	$\max L_j$	$O(nP^2)$
	$\sum w_jU_j$	$O(nP^5)$

Future Research

- Multiple machines
- Multiple resources
- Multiple renting intervals
- Specific applications...?

References

[1] Briskorn, D., Davari, M., and Matuschke, J. (2021). Single-machine scheduling with an external resource. *European Journal of Operational Research*, 293(2):457–468.

[2] Jackson, J. R. (1955). Scheduling a production line to minimize maximum tardiness. *Research Report 43, Management Science Research Project, University of California, Los Angeles, CA*.

[3] Lawler, E. and Moore, J. M. (1969). A functional equation and its application to resource allocation and scheduling problem. *Management Science*, 16:77–84.

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

Single-machine scheduling with an external resource

Speaker:
Dirk Briskorn
Department of Production and Logistics, University of Wuppertal, Germany

Co-authors:
Morteza Davari
SKEMA Business School, Lille, France
Jannik Matuschke
Research Centre for Operations Management, KU Leuven, Belgium

Oct 2023 @ schedulingseminar.com