Wroclaw University of Science and Technology

On problems and methods of coordinated scheduling and location

Jerzy Józefczyk, Mirosław Ławrynowicz, Grzegorz Filcek

Wroclaw University of Science and Technology,
Division of Intelligent Decision Support Systems
Wroclaw, Poland

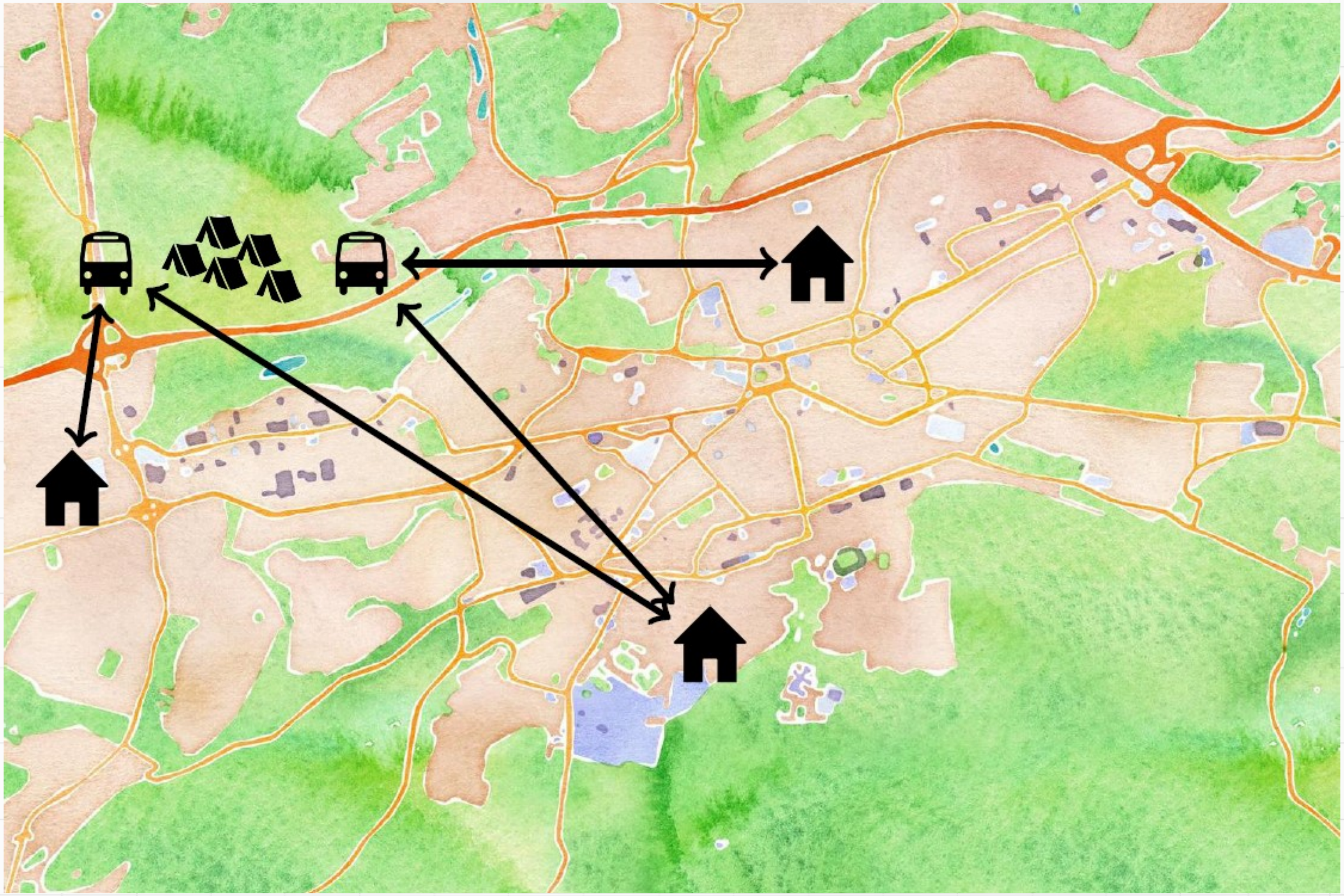
Wrocław University of Science and Technology

Agenda

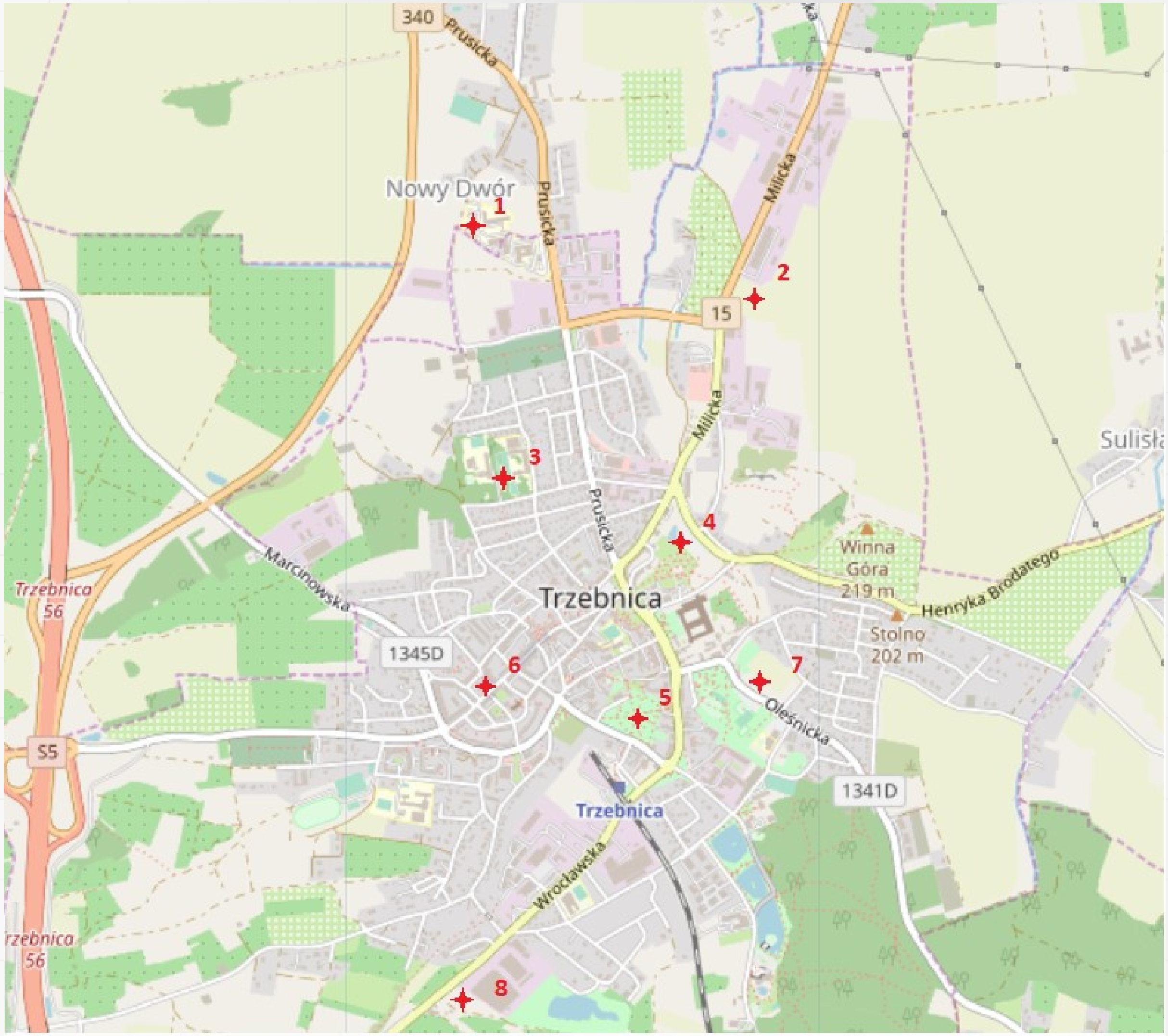
1. Motivation
2. Scheduling sub-problem
3. Location sub-problem
4. Coordinated scheduling and location (ScheLoc)
5. Optimization schemes for ScheLoc
6. Selected results
7. Other cases and final remarks

Motivation (1/2)

Applications:
- Loading/unloading of ships at an embankment
- Planning of evacuation from danger zones



Motivation (2/2)



Wroclaw University of Science and Technology

Scheduling sub-problem

Scheduling of jobs on parallel machines

$\mathrm{J} = \{J_1, J_2, \dots, J_j, \dots, J_n\}$ – set of n jobs

$\mathrm{M} = \{M_1, M_2, \dots, M_i, \dots, M_m\}$ – set of m machines

$\sigma \in \mathrm{D}_\sigma$ – schedule

$p = [p_{ij}]_{\substack{i=\overline{1,m} \\ j=\overline{1,n}}}$ – processing times

$w = [w_j]_{j=\overline{1,n}}$ – weights $r = [r_j]_{j=\overline{1,n}}$ – release dates $d = [d_j]_{j=\overline{1,n}}$ – due dates

$\mathrm{A_S} = (p, d, w, r)$ – sequence of parameters

$[C_j(\sigma; \mathrm{A_S})]_{j=\overline{1,n}}$ – completion times of jobs

$\bar{Q}_\mathrm{S}([C_j(\sigma; \mathrm{A_S})]_{j=\overline{1,n}}) \triangleq Q_\mathrm{S}(\sigma; \mathrm{A_S})$ – job scheduling criteria

Sub-problem PS $\min\limits_{\sigma \in \mathrm{D}_\sigma} Q_\mathrm{S}(\sigma; \mathrm{A_S})$

Wroclaw University of Science and Technology

Location sub-problem (1/2)

Location of machines

$a = \left[a_j \right]_{j=\overline{1,n}}$ – given locations of jobs

$x = \left[x_i \right]_{i=\overline{1,m}}$ – sought locations of machines

$v = \left[v_j \right]_{j=\overline{1,n}}$ – speeds of job movements

$c = [c_{ij}]_{\substack{i=\overline{1,m} \\ j=\overline{1,n}}}$ – unit movement/transportation costs

$g = [g_i]_{i=\overline{1,m}}$ – unit location/investment costs

$\mathrm{A_L} = (a, v, c, g)$ – sequence of parameters

Sub-problem PL $\min\limits_{x \in \mathrm{D}_x} Q_{\mathrm{L}}(x; \mathrm{A_L})$ (*mainly, location and/or movement costs*)

where D_x represents available machines' positions taken from the discrete or continuous area

Location sub-problem (2/2)

$r_{ij}(x_i) = \rho_j + v_j^{-1} d(a_j, x_i)$ – generalized release dates, where $d(\cdot)$ – distance

$\mathrm{D}_\sigma \to \mathrm{D}_{\sigma x} \qquad Q_{\mathrm{S}}(\sigma; \mathrm{A}_{\mathrm{S}}) \to Q_{\mathrm{SL}}(\sigma, x; \mathrm{A}_{\mathrm{S}})$

$\alpha \,|\, \beta \,|\, \gamma \,|\, \delta$ – extended Graham's notation

$\beta:\ r_j\_\mathrm{V}, \qquad r_j^- \le r_j \le r_j^+$

$\delta:$
- – UFLP (*full UFLP*)
- – UFLP_T (*UFLP with the restriction to a transportation cost*)
- – UFLP_I (*UFLP with the limitation to an investment cost*)
- – P_MEDIAN (*p-median problem*)
- – CFLP (*facility location with continuous location area*)

Coordinated scheduling and location /ScheLoc/ (1/2)

Table 1. The track record on coordinated scheduling-location

$1 \mid r_j\_V \mid C_{\max} \mid \text{UFLP\_T}$	Hennes and Hamacher (2002)
$1 \mid r_j\_V \mid C_{\max} \mid \text{CFLP}$	Elvikis et al. (2009), Kalsch and Drezner (2010)
$1 \mid p_j = p, r_j\_V \mid C_{\max} \mid \text{CFLP}$	Kalsch and Drezner (2010)
$P \mid r_j\_V \mid C_{\max}$	Rajabzadeh et al. (2016)
$P \mid r_j\_V \mid C_{\max} \mid \text{UFLP\_T}$	Hessler and Deghdak (2017)
$P \mid r_j\_V \mid \sum C_j \mid \text{CFLP}$	Piasecki (2018), Piasecki and Józefczyk (2018)
$R \mid r_j\_V \mid C_{\max} \mid \text{P\_MEDIAN}$	Ławrynowicz and Józefczyk (2019)
$R \mid r_j\_V \mid C_{\max}$	Ławrynowicz and Józefczyk (2019)
$P \mid r_j\_V \mid \sum C_j \mid \text{UFLP\_I}$	Filcek et al. (2020)

Coordinated scheduling and location /ScheLoc/ (2/2)

1. Hennes H., Hamacher H. Integrated Scheduling and Location Models: Single Machine Makespan Problems. Report in Wirtschaftsmathematik 82, Univ. of Kaiserslautern, 2002
2. Elvikis D., Hamacher H. W., Kalsch M. T. Simultaneous scheduling and location (ScheLoc): The planar ScheLoc makespan problem. ***Journal of Scheduling*** 12 (4), 2009, 361-374
3. Kalsch M. T., Drezner Z. Solving scheduling and location problem in the plane simultaneously. ***Computers & Operations Res***. 37(2), 2010, 256-264.
4. Rajabzadeh M., Ziaee M., Bozorgi-Amiri A. Integrated approach in solving parallel machine scheduling and location (ScheLoc) problem. ***International Journal of Industrial Engineering Computations*** 7(4), 2016
5. Hessler, C.J., Deghdak, K. (2017) Discrete parallel machine makespan ScheLoc problem, ***Journal of Combinatorial Optimization***, 34(4), 1159-1186.
6. Piasecki B., Józefczyk J. (2018) Evolutionary algorithm for joint task scheduling and deployment of executors, In: *Automation of Discrete Processes. Theory and Applications* (in Polish), Silesian University of Technology, 1, 169-178.
7. Liu M., Liu X., Zhang E., Chu F., Chu Ch. (2019) Scenario-based heuristic to two-stage stochastic program for the parallel machine ScheLoc problem, ***International Journal of Production Research***, 57(6),1706-1723.
8. Ławrynowicz M., Józefczyk J (2019) A memetic algorithm for the discrete scheduling-location problem with unrelated executors; Proc. of 24th Int. Conf. on Models and Methods in Automation and Robotics MMAR, Międzyzdroje, Poland, August 26-29 (in press).

Optimization schemes for ScheLoc (1/9)

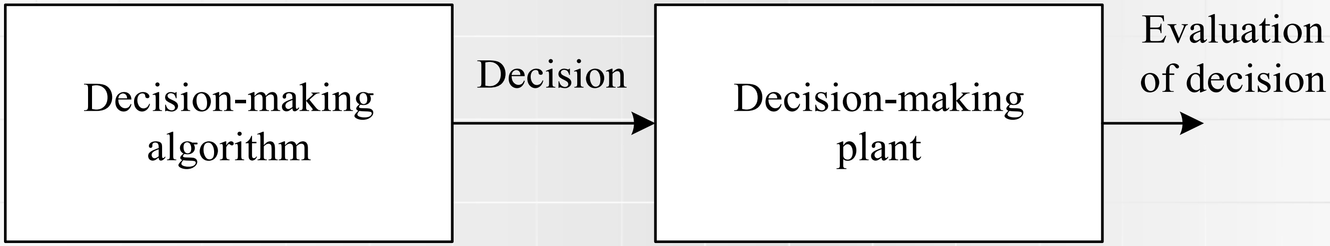


Fig.1 Decision-making system

Decision-making problem

Given:

– model of a decision-making plant

– requirement for a result of a decision (evaluation of a decision)

Determine:

– decision fulfilling a requirement

Optimization schemes for ScheLoc: /S_Seq/ (2/9)

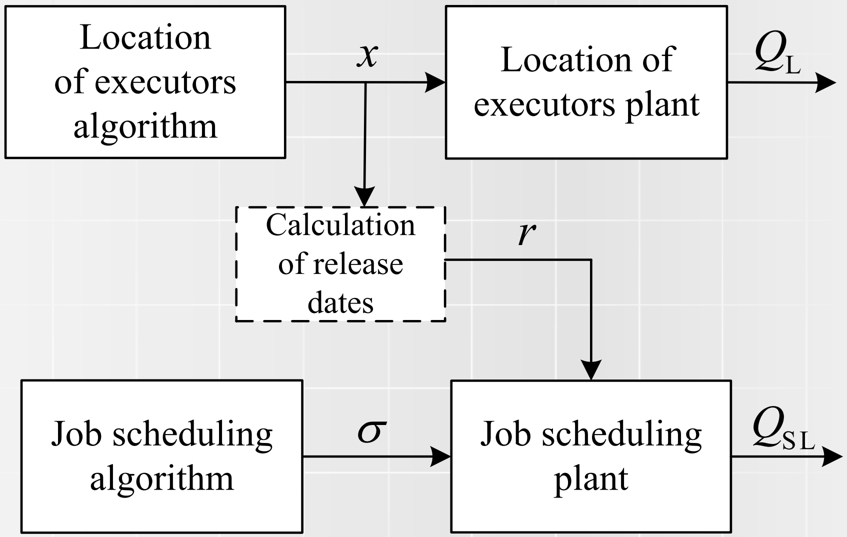


Fig. 2 Sequential approach for solving ScheLoc /S_Seq/

Wroclaw University of Science and Technology

Optimization schemes for ScheLoc: /S_Seq/ (3/9)

Sequential optimization scheme S_Seq

Require: J, M, $\mathrm{A_S}$, $\mathrm{A_L}$, $\mathrm{D}_{\sigma x}$, D_x

Ensure: $\tilde{x}, \tilde{\sigma}, \tilde{Q}_{\mathrm{L}} \square Q_{\mathrm{L}}(\tilde{x}; \mathrm{A_L}), \tilde{Q}_{\mathrm{SL}} \square Q_{\mathrm{SL}}(\tilde{\sigma}, \tilde{x}; \mathrm{A_S})$,

1. Solve the locational problem $\min_{x \in \mathrm{D}_x} Q_{\mathrm{L}}(x; \mathrm{A_L})$ to obtain $\tilde{x}$ and $\tilde{Q}_{\mathrm{L}}$.
2. Calculate $r_{ij} = \rho_j + v_j^{-1} d(a_j, \tilde{x}_i), i = \overline{1,m}, j = \overline{1,n}$.
3. Set the release dates of jobs using the selected operator Φ to have $r_j = \Phi(r_{1j}, ..., r_{mj})$.
4. Solve the job scheduling problem with different release dates r_j $\min_{\sigma \in \mathrm{D}_{\sigma x}} Q_{\mathrm{SL}}(\sigma, \tilde{x}; \mathrm{A_S})$ to determine $\tilde{\sigma}$ and $\tilde{Q}_{\mathrm{SL}}$.

Optimization schemes for ScheLoc: /S_Jnt/ (4/9)

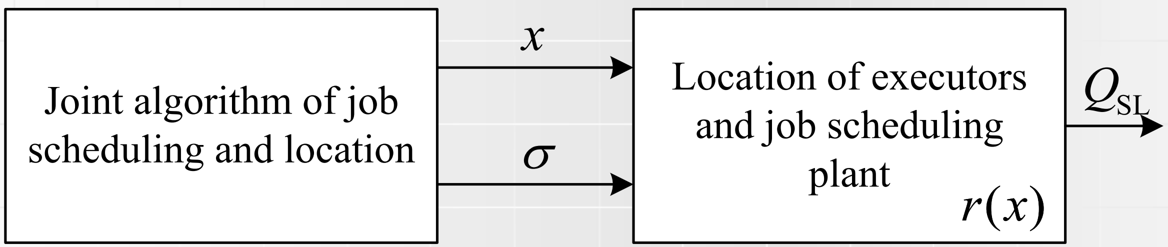


Fig. 3 Joint approach for solving Schec with an only criterion Q_{SL} /S_Jnt/

Optimization schemes for ScheLoc: /S_Jnt/ (5/9)

Joint optimization scheme S_Jnt

Require: J, M, A_S, A_L, D

Ensure: $\hat{\sigma}$, $\hat{x}$, $\hat{Q}_{SL} \square Q_{SL}(\hat{\sigma},\hat{x};A_S)$

1. Solve the joint scheduling-location problem $\min_{(\sigma,x)\in D} Q_{SL}(\sigma,x;A_S)$ to obtain $\hat{\sigma}$, $\hat{x}$, and $\hat{Q}_{SL}$.

Optimization schemes for ScheLoc: /S_Jnt_B/ (6/9)

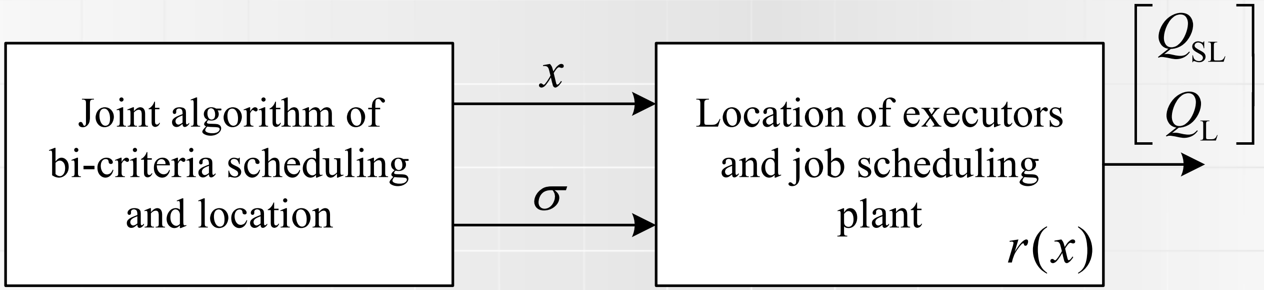


Fig. 4 Bi-criteria joint approach for solving ScheLoc /S_Jnt_B/

Optimization schemes for ScheLoc: /S_Jnt_B/ (7/9)

Joint bi-criteria optimization scheme S_Jnt_B

Require: J, M, $\mathrm{A_S}$, $\mathrm{A_L}$, D

Ensure: Set of λ pairs of decisions $\mathrm{Arg}_{\mathrm{PF}} = \{(\sigma_l^*, x_l^*) : l = \overline{1,\lambda}\}$ and corresponding Pareto front

$\mathrm{PF} = (Q_l^* \square [Q_{\mathrm{SL},l}^*, Q_{\mathrm{L},l}^*]^{\mathrm{T}} : Q_{\mathrm{SL},l-1}^* < Q_{\mathrm{SL},l}^*, l = \overline{2,\lambda})$.

1. Solve the joint scheduling-location problem $\min\limits_{(\sigma,x)\in \mathrm{D}} \begin{bmatrix} Q_{\mathrm{SL}}(\sigma, x; \mathrm{A_S}) \\ Q_{\mathrm{L}}(x; \mathrm{A_L}) \end{bmatrix}$ to have $\mathrm{Arg}_{\mathrm{PF}}$ and PF.

Optimization schemes for ScheLoc: /S_Unc/ (8/9)

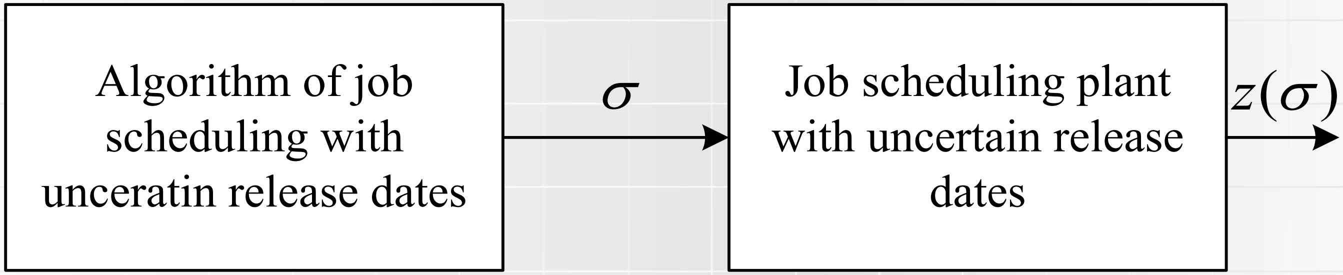


Fig. 5 Uncertain approach for solving ScheLoc – addressing as an uncertain version of PS /S_Unc/

Optimization schemes for ScheLoc: /S_Unc/ (9/9)

Uncertain optimization scheme S_Unc

Require: J, M, A_S , D_σ , R

Ensure: $\sigma'', r^\sigma, z(\sigma''), Q_S(\sigma'', r^\sigma) \square Q''_S$

1. Solve $\min_{\pi \in D_\pi} \hat{Q}_S(\pi, r)$ to have $Q'_S(r)$.
2. Obtain the worst-case scenario as $r^\sigma = \arg\max_{r \in R}[\hat{Q}_S(\sigma, r) - Q'_S(r)]$ and calculate $z(\sigma) = \hat{Q}_S(\sigma, r^\sigma) - Q'_S(r^\sigma)$.
3. Solve $\min_{\sigma \in D_\sigma} z(\sigma)$ to obtain σ'' , $z(\sigma'')$, and Q''_S .

Selected results: Case $\mathrm{P} \mid r_j\_\mathrm{V} \mid \sum C_j \mid \mathrm{UFLP\_I}$ (1/4)

Filcek G., Józefczyk J., Ławrynowicz M. (2020) An evolutionary algorithm for joint bi-criteria location-scheduling problem. *International Journal of Industrial Engineering Computations* (accepted)

discrete locations, the Euclidean distances

μ locations ready for deploying m machines

A_S [illegible] $\qquad \mathrm{A}_\mathrm{L} = (a, v, g)$

$$Q_{\mathrm{SL}}(\sigma, x; \mathrm{A}_\mathrm{S}) \rightarrow \quad Q_{\mathrm{SL}}(s, x; \mathrm{A}_\mathrm{S}) = \sum_{i=1}^{\mu} \sum_{k=1}^{n} x_i C_{ik}(s, x)$$

s.t.

$$1 \le \sum_{i=1}^{\mu} x_i \le \mu,$$

$$\sum_{i=1}^{\mu} \sum_{k=1}^{n} s_{jik} = 1, \quad j = 1, 2, \ldots, n,$$

$$\sum_{j=1}^{n} s_{jik} \le x_i, \quad i = 1, 2, \ldots, \mu, \quad k = 1, 2, \ldots, n,$$

$$C_{ik}(s, x) \ge \sum_{j=1}^{n} (r_j + p_j) s_{jik}, \quad i = 1, 2, \ldots, \mu, \quad k = 1, 2, \ldots, n,$$

$$C_{ik}(s, x) \ge C_{i,k-1}(s, x) + \sum_{j=1}^{n} p_j s_{jik}, \quad i = 1, 2, \ldots, \mu, \quad k = 2, 3, \ldots, n$$

Selected results: Case $\mathrm{P} \mid r_j\_\mathrm{V} \mid \sum C_j \mid \mathrm{UFLP\_I}$ (2/4)

Algorithm ALG_BC

Require: Data of BC_ScheLoc: J, B, A, μ, p_j, ρ_j, v_j, c_i, and parameters of the algorithm: φ_m, φ_c, α_i, $\bar{\alpha}$, $\tilde{\alpha}$, Γ, Γ_{max}.

Ensure: Set of non-dominated solutions S_{PF} and the Pareto front PF.

1: $i:=1$, $z:=1$, $\tilde{z}:=1$, $S_M(iz):=\varnothing$

2: **while** $i \le \mu$ **do**

3: Generate the initial population S_i of α_i solutions for exactly i executors and set $\overline{S}_i(0) := \varnothing$

4: **while** $iz < \Gamma_{max}$ or $\tilde{z} < \Gamma$ **do**

5: Perform a non-dominated sorting and crowding-distance assignment.

6: Use a crowded comparison operator for a selection.

7: Generate the set $\overline{S}_i(z)$ of offsprings using the crossover (CPC-2/OX2) and mutation (SIM/SM) operators.

8: **for each** l th element in $\overline{S}_i(z)$ that is non-dominated by any element in $S_M(iz)$ **do**

9: $S_M(iz) := S_M(iz) \cup \overline{E}_{il}(z)$

10: **end for**

11: **if** $\left|\overline{S}_i(z)\right| = \left|\overline{S}_i(z-1)\right|$ **then** $\tilde{z} := \tilde{z}+1$ **else** $\tilde{z} := 1$ **end if**

13: **if** $\Gamma_{\max} \bmod z = 0$ or $\tilde{z} := \Gamma$ **then** set $i := i+1$, $z := 1$ and **go to 2, end if**

16: **end while**

17: Set $z := z := z+1$.

18: **end while**

19: **return** $S_{PF} = \left\{\left(x_{\mathrm{PF},1} = f_x\left(E_{L,l}\right), y_{\mathrm{PF},1} = f_y\left(E_{Lu}\right)\right), l = 1,2,\ldots,L(iz)\right\}$

and $PF = \left(\left(q^{(1)}(x_{\mathrm{PF},1}, y_{\mathrm{PF},1}), q^{(2)}(y_{\mathrm{PF},1})\right) \square\, (q_l^{(1)}, q_l^{(2)}) : q_{l-1}^{(1)} < q_l^{(1)}, l = 1,2,\ldots,L(iz)\right)$

Selected results: Case $P \mid r_j\_V \mid \sum C_j \mid UFLP\_I$ (3/4)

Table 2 Dependence of $Q_{SL,d}, Q_{L,d}$ and *Dist* on *n* and μ for both algorithms

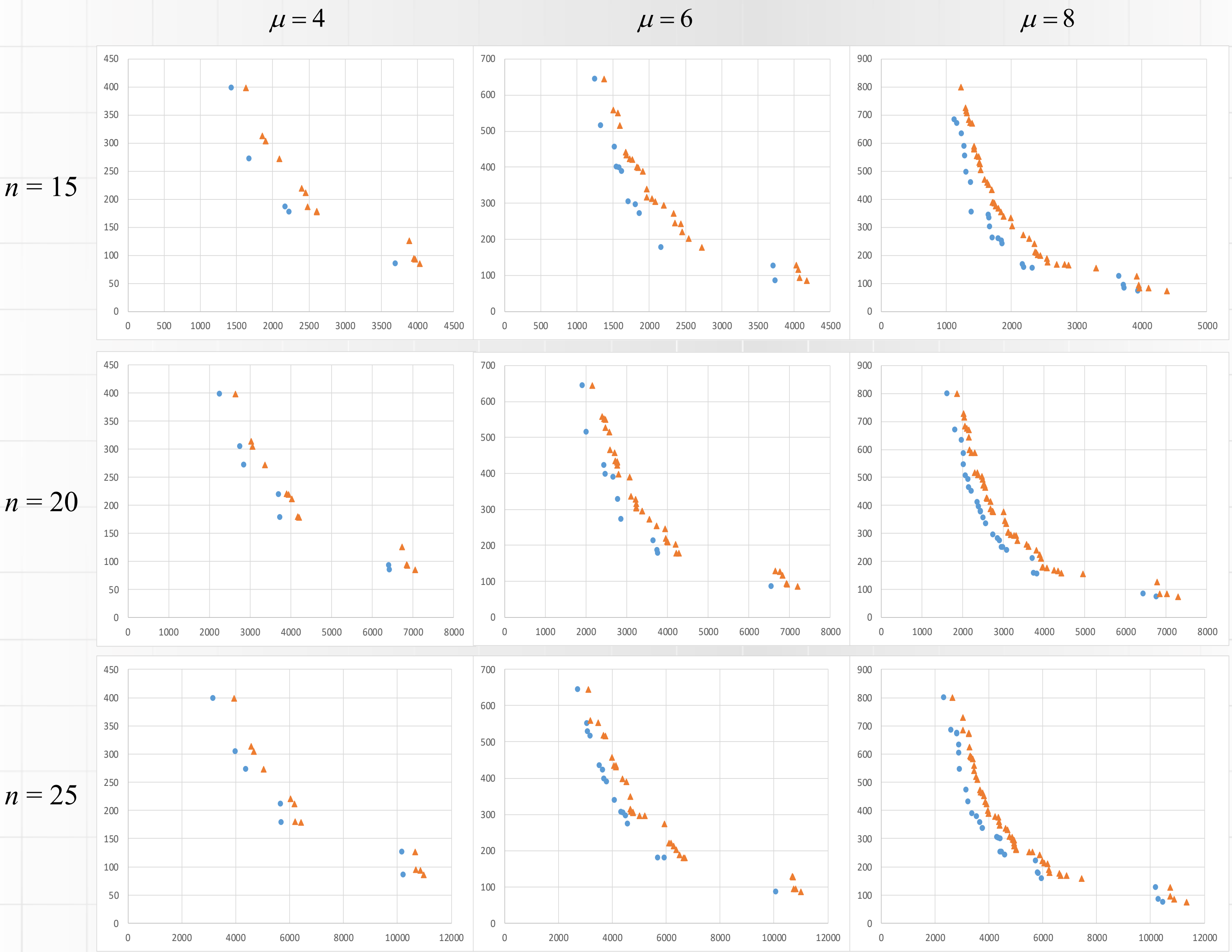
(n, μ)	$Q^*_{SL,l}$	$Q^*_{L,l}$	$Q^{NS}_{SL,l}$	$Q^{NS}_{L,l}$	$Dist^*$	$Dist^{NS}$	$Dist^{NS} / Dist^*$
(10, 4)	266.776	182	**266.776**	**182**	322.945	**322.945**	1.000
(15, 4)	432.052	285	**432.052**	**285**	517.584	**517.584**	1.000
(20, 4)	596.663	385	624.574	**385**	710.093	733.701	1.033
(25, 4)	866.800	385	956.164	**385**	948.455	1030.765	1.087
(10, 6)	265.559	177	**265.559**	**177**	319.140	**319.140**	1.000
(15, 6)	406.760	257	448.850	**257**	481.147	517.218	1.075
(20, 6)	458.741	406	477.497	452	612.600	725.684	1.185
(25, 6)	611.681	452	716.951	536	760.564	911.585	1.199

Dist – distance between (0,0) and Pareto front /$(Q_{SL,l}, Q_{L,l})$/

NS – NSGS II-based algorithm

Wroclaw University of Science and Technology

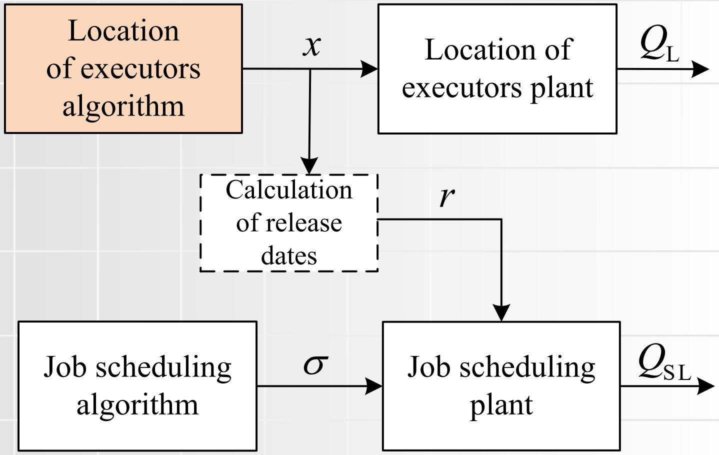
Selected results: Case $P \mid r_j\_V \mid \sum C_j \mid UFLP\_I$ (4/4)



Selected results: Cases $R|r_j\_V|C_{max}|P\_MEDIAN$ and $R|r_j\_V|C_{max}$ (1/5)

Ławrynowicz M. and Józefczyk J (2019) A memetic algorithm for the discrete scheduling-location problem with unrelated executors; *Proc. of 24th Int. Conf. on Models and Methods in Automation and Robotics MMAR*. Międzyzdroje. Poland, August, 26-29.

$R|r_j\_V|C_{max}|P\_MEDIAN \rightarrow$ Scheme S_Seq



discrete locations, the Euclidean distances

μ locations ready for deploying m machines

$A_L = (a, v)$

Sub-problem PL (*p-median problem*)

$$Q_L(x; A_L) = \sum_{i=1}^{\mu} \min_{x \in D_x} r_{ij} \quad r_{ij}(x_i) = \rho_j + v_j^{-1} d(a_j, x_i)$$

Rosing, K. E., Revelle, C. S., & Schilling, D. A. (1999) A gamma heuristic for the p-median problem. *European Journal of Operational Research*, **117**, 3, 522-532.

Selected results: Cases $\mathrm{R} \mid r_j\_V \mid C_{\max} \mid \mathrm{P\_MEDIAN}$ and $\mathrm{R} \mid r_j\_V \mid C_{\max}$ (2/5)

Sub-problem PS

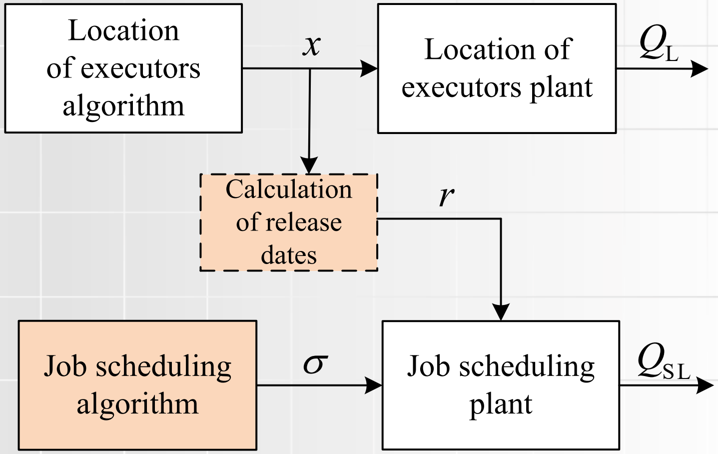
$$\mathrm{A_S} = (p, r)$$

$$Q_{\mathrm{SL}}(\sigma, x; \mathrm{A_S}) = \max_{1 \le i \le m} C_{\sigma_i^{n_i}}(x_i)$$

$$\sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_i, \ldots, \sigma_m\}$$

$$\sigma_i = \left(\sigma_i^1, \sigma_i^2, \ldots, \sigma_i^l, \ldots, \sigma_i^{n_i}\right)$$

$$\mathrm{D}_{\sigma x} = \{\sigma : \sigma_{i_1}^{l_1} \ne \sigma_{i_2}^{l_2},\ \sum_{i=1}^{m} n_i = n\}$$

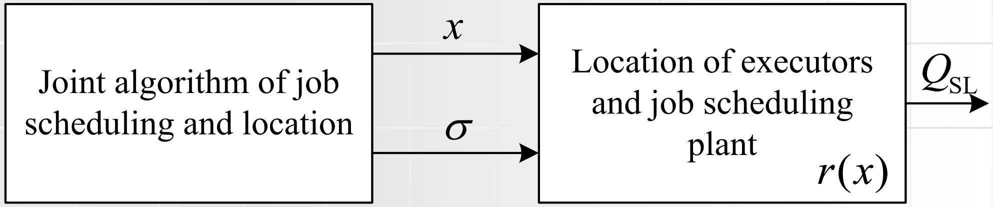


PSO–based solution algorithm

Lin, Y. K. (2013). Particle swarm optimization algorithm for unrelated parallel machine scheduling with release dates. *Mathematical Problems in Engineering*, Article ID 409486.

Selected results: Cases $R\,|\,r_j\_V\,|\,C_{\max}\,|\,P\_MEDIAN$ and $R\,|\,r_j\_V\,|\,C_{\max}$ (3/5)

$R\,|\,r_j\_V\,|\,C_{\max} \rightarrow$ Scheme S_Jnt



Algorithm TSMA. Tabu Search based Memetic algorithm

Require: Data: $D=\{\mathrm{M},\mathrm{Y},\tilde{\mathrm{Y}},\mathrm{J},\mathrm{S},\mathrm{V},p\}$ and parameters of the algorithm: α_{init}, α_{par}, φ_m, φ_r, Γ, $\Gamma_{\max}$, ρ, $\tilde{\rho}$, κ^{bin}, κ^{integer}

Ensure: Heuristic solution X and Π as well as $C_{\max}^{\text{TSMA}}$.

1. Generate the initial population $\tilde{\mathbf{E}}(z)$ for $z:=1$ and set $\mathbf{E}^* := \mathbf{E}^{\text{best}}(z)$.
2. **while** stop condition defined by $(\Gamma, \Gamma_{\max}, \mathbf{E}^*)$ is not satisfied
3. Evaluate $\tilde{\omega}(z)$ by population $\tilde{\mathbf{E}}(z)$.
4. Select parents $\tilde{\mathbf{E}}^{\text{sus}}(z)$ with the use of Stochastic Universal Sampling.
5. Generate the set $\tilde{\mathbf{E}}^{\text{off}}(z)$ of offsprings using the recombination (CPC-2/OX1) and mutation operators.
6. Run the Local Search Heuristic with arguments: $\tilde{\omega}^{\text{off}}(z)$, D and return $\tilde{\mathbf{E}}^{\text{TS}}$.
7. **if** $C_{\max}(\mathbf{E}^*) > C_{\max}(\mathbf{E}^{\text{TS}_{\text{best}}})$ **then** $\mathbf{E}^* := \mathbf{E}^{\text{TS}_{\text{best}}}$ **end if**
8. Set $z := z+1$ and $\tilde{\mathbf{E}}(z) := \tilde{\mathbf{E}}^{\text{TS}} \cup \tilde{\mathbf{E}}^{\text{off}_{\text{best}}}(z)$.
9. **end while** (2)
10. $X := f_X(\mathbf{E}^*;\mathrm{Y})$, $\Pi := f_\Pi(\mathbf{E}^*)$, $C_{\max}^{\text{TSMA}} := C_{\max}(\Pi, X)$

Selected results: Cases $R \mid r_j\_V \mid C_{max} \mid P\_MEDIAN$ and $R \mid r_j\_V \mid C_{max}$ (4/5)

Table 3 Dependence of Q_{SL} and t on n for joint and sequential approaches

n	$\hat{Q}_{SL}$	$\tilde{Q}_{SL}$	δ_Q	$\hat{t}_{SL}$ [s]	$\tilde{t}_{SL}$ [s]	δ_t
100	39.764	52.780	32.733	9.221	7.035	-23.707
200	71.551	100.954	41.094	16.934	14.697	-13.210
300	146.625	188.781	28.751	25.755	26.968	4.710
400	235.912	297.873	26.264	34.777	36.089	3.773
500	300.651	369.762	22.987	41.048	47.114	14.778
600	348.684	462.543	32.654	49.912	58.281	16.768
700	405.646	527.986	30.159	58.995	72.092	22.200
800	489.871	659.451	34.617	67.665	83.668	23.650
900	579.543	778.992	34.415	80.019	99.769	24.682
1000	645.671	868.621	34.530	88.894	117.249	31.898

$\hat{Q}_{SL},\ \hat{t}_{SL} - S\_\mathrm{Jnt}$ $\qquad$ $\tilde{Q}_{SL},\ \tilde{t}_{SL} - S\_\mathrm{Seq}$

$$\delta_Q = \frac{\tilde{Q}_{SL} - \hat{Q}_{SL}}{\hat{Q}_{SL}} 100\% \qquad \delta_t = \frac{\tilde{t}_{SL} - \hat{t}_{SL}}{\hat{t}_{SL}} 100\%$$

Selected results: Cases $R \mid r_j\_V \mid C_{max} \mid$ P_MEDIAN and $R \mid r_j\_V \mid C_{max}$ (5/5)

Property. The optimization scheme S_Seq does not outperform the optimization scheme S_Jnt, i.e., $Q_{\mathrm{SL}}(\tilde{\sigma},\tilde{x};\mathrm{A_S}) \ge Q_{\mathrm{SL}}(\hat{\sigma},\hat{x};\mathrm{A_S})$

Justification:

$$Q_{\mathrm{SL}}(\hat{\sigma},\hat{x};\mathrm{A_S}) = \min_{(\sigma,x)\in\mathrm{D}} Q_{\mathrm{SL}}(\sigma,x;\mathrm{A_S}) \le \min_{(\sigma,x)\in\mathrm{D}'} Q_{\mathrm{SL}}(\sigma,x;\mathrm{A_S}) = \min_{\sigma\in\mathrm{D}_{\sigma\tilde{x}}} Q_{\mathrm{SL}}(\sigma,\tilde{x};\mathrm{A_S}) = Q_{\mathrm{SL}}(\tilde{\sigma},\tilde{x};\mathrm{A_S}),$$

where $\tilde{x} = \arg\min_{x\in\mathrm{D}_x} Q_{\mathrm{L}}(x;\mathrm{A_L})$, $\mathrm{D}' = \{(\sigma,x) : \sigma\in\mathrm{D}_{\sigma\tilde{x}} \wedge \tilde{x} = \arg\min_{x\in\mathrm{D}_x} Q_L(x;\mathrm{A_L})\}$, and $\mathrm{D}' \subseteq \mathrm{D}$.

Selected results: Uncertain case $1|r_j^- \le r_j \le r_j^+|C_{\max}$ (1/2)

Release dates belong to given intervals, i.e., $r_j \in [r_j^-, r_j^+]$.

How to determine intervals?

$$\hat{Q}_S(\sigma,r) \triangleq C_n(\sigma,r)$$

where $C_k(\sigma,r) = p_{\sigma(k)} + \max\{C_{k-1}(\sigma,r), r_{\sigma(k)}\}, \quad k = 2,3,\ldots,n, \;\; C_0(\sigma,r) = 0$

$\hat{Q}_S(\sigma,r) - Q'_S(\sigma'_r)$ – regret

where $\sigma' = \arg\min_{\pi \in D_\pi} Q(\pi,r)$ – optimal solution for fixed scenario r

D_π – set of permutations without repetitions

$$Q'_S \triangleq \min_{\pi \in D_\pi} Q(\pi,r)$$

$z(\sigma) = \max_{r \in R}[\hat{Q}_S(\sigma,r) - Q'_S(\sigma'_r)]$ – maximum regret for given σ

$r^\sigma = \arg\max_{r \in R}[\hat{Q}_S(\sigma,r) - Q'_S(r)]$ – worst-case scenario

$\min_{\sigma \in D_\sigma} z(\sigma)$ $\qquad$ $\sigma'' = \arg\min_{\sigma \in D_\sigma} z(\sigma)$ – solution

Selected results: Uncertain case $1|r_j^- \le r_j \le r_j^+|C_{max}$ (2/2)

Table 4 Dependence of makespan and computational time on n for S_Jnt and S_Unc, $\mu = 50$

n	any intervals				disjoint intervals			
	$\hat{Q}_{SL}$	$Q_S(\sigma'')$	$\hat{t}_{SL}$ $[ms]$	t_S $[ms]$	$\hat{Q}_{SL}$	$Q_S(\sigma'')$	$\hat{t}_{SL}$ $[ms]$	t_S $[ms]$
40	2741.530	2776.632	8	39	**556.55**	**556.55**	11	9
80	5719.114	5747.075	10	296	**709.33**	**709.33**	16	12
120	8737.921	8785.887	26	1241	**811.91**	**811.91**	29	22
160	11505.807	11533.889	29	4410	**987.38**	**987.38**	39	27
200	14482.500	14503.128	31	6712	**1108.99**	**1108.99**	55	40
240	17534.556	17557.584	47	12744	**1293.77**	**1293.77**	74	50
280	20638.523	20680.748	63	22208	**1671.41**	**1671.41**	92	67
320	23102.561	23138.121	78	35916	**1822.76**	**1822.76**	111	80
360	**26457.697**	**26457.697**	92	49812	**2098.41**	**2098.41**	133	97
400	**29299.100**	**29299.100**	110	71251	**2223.91**	**2223.91**	161	117

Other cases and final remarks (1/3)

Connection of the **deployment of machines with the assignment of jobs to machines**.

Next, ***m*** **separate job scheduling sub-problems on single machines**.

Advantage – substantial simplification of PS.

The detailed analysis is an open issue, in particular, on the connection between location and assignment.

The first attempt can be found in

Hessler C.J. and Deghdak K. (2017) Discrete parallel machine makespan ScheLoc problem. *Journal of Combinatorial Optimization*, **34**, 4, 1159-1186.

clusters of jobs are determined around launched sites for machines.

Other cases and final remarks (2/3)

Machine-dependent release dates r_{ij} are figured in S_Seq.

PS needs a single release date r_j .

The substantiation other than max, min or mean value needs research.

We would have a new issue with machine-dependent release dates; the comparison to similar known problems required.

Other cases and final remarks (3/3)

How to calculate r_{ij} ?

Applied proposition: $r_{ij}(x_i) = \rho_j + v_j^{-1} d(a_j, x_i)$

Other methods for S_Unc, e.g., the probabilistic one
and
release date values acquisition

Wroclaw University of Science and Technology

Thank you for your attention.