Robustness Measures for Stochastic Parallel Machine Scheduling and the Train Unit Shunting Problem

Casper Loman Marjan van den Akker Loriana Pascual
Roel van den Broek Han Hoogeveen

Utrecht University

May 14, 2024

Robustness for local search problems

- Local search has been successful in solving difficult scheduling problems

Robustness for local search problems

- Local search has been successful in solving difficult scheduling problems, but how to deal with real-life disruptions?

Robustness for local search problems

- Local search has been successful in solving difficult scheduling problems, but how to deal with real-life disruptions?
- Introduce slacks in the schedule
  - Where, how much?

Robustness for local search problems

- Local search has been successful in solving difficult scheduling problems, but how to deal with real-life disruptions?
- Introduce slacks in the schedule
  - Where, how much?
- More general, include robustness in the objective function

Robustness for local search problems

- Local search has been successful in solving difficult scheduling problems, but how to deal with real-life disruptions?
- Introduce slacks in the schedule
  - Where, how much?
- More general, include robustness in the objective function
- Simulation (e.g. vdA, van Blokland, Hoogeveen, 2013):
  - Taking many samples from the processing time distributions.
  - Accurate estimation, but computationally expensive.

Robustness for local search problems

- Local search has been successful in solving difficult scheduling problems, but how to deal with real-life disruptions?
- Introduce slacks in the schedule
  - Where, how much?
- More general, include robustness in the objective function
- Simulation (e.g. vdA, van Blokland, Hoogeveen, 2013):
  - Taking many samples from the processing time distributions.
  - Accurate estimation, but computationally expensive.
- **Robustness measures** (RMs): analytical functions that estimate robustness based on the schedule structure.
- Many RMs have been suggested, no consensus in which one works best

Our contribution

- Overview of robustness measures, including a few new ones
- Elaborate computational study to assess their quality
- Application of these measures for the Train Unit Shunting Problem

Stochastic parallel machine scheduling

- We consider stochastic processing times, where p_j is from a given distribution D_j.
- Construct a **baseline schedule**
  - Use **Local Search** to find baseline schedule
- Policy in schedule execution:
  - Ordering of jobs and machine assignment from base line schedule
  - Job j can not start earlier than its planned start time PST_j in the baseline schedule
  - Jobs are started as soon as possible

Stochastic parallel machine scheduling

- We consider stochastic processing times, where p_j is from a given distribution D_j.
- Construct a **baseline schedule**
  - Use **Local Search** to find baseline schedule
- Policy in schedule execution:
  - Ordering of jobs and machine assignment from base line schedule
  - Job j can not start earlier than its planned start time PST_j in the baseline schedule
  - Jobs are started as soon as possible

Jobs may not start on their planned start time and the makespan will not be as planned.

We want to create **robust** baseline schedules.

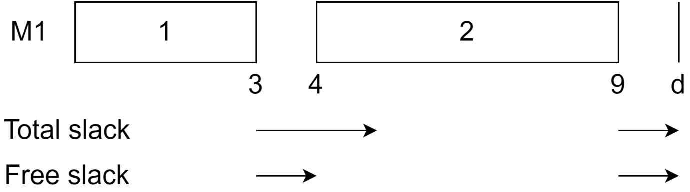
Robustness

Robustness: ability of a schedule to withstand disruptions

- **Quality robustness**: the objective function is unlikely to degrade under uncertainty.
  - e.g. schedule is likely to meet deadline
- **Solution robustness**: the schedule itself is unlikely to degrade under uncertainty.
  - e.g. jobs are likely to start on their planned start time.

Slack

- **Total slack** of a job (Leon, Wu, Storer, 1994):
  - The maximum amount of time that we can delay a job without exceeding the deadline.
- **Free slack** of a job (Al-Fawzam and Haouari, 2005):
  - The amount of time by which we can delay the job without delaying any other job in the schedule.



job	1	2
p_j	3	5
TS_j	2	1
FS_j	1	1

Slack based robustness measures

Sum:	$RM_1(S) = \sum_j TS_j^S$
	$RM_2(S) = \sum_j FS_j^S$
Minimum:	$RM_3(S) = \min_j\{TS_j^S\}$
	$RM_4(S) = \min_j\{FS_j^S / p_j\}$
	$RM_5 = \sum_j \min\{FS_j, \lambda p_j\}$
Counting:	$RM_6(S) = \sum_j \alpha_j$ where $\alpha_j = 1$ if $FS_j^S > 0$
Weighted sum:	$RM_7(S) = \sum_j FS_j^S \times p_j$
	$RM_8(S) = \sum_j FS_j^S \times NDP_j^S$
	$RM_9(S) = \sum_j FS_j^S \times NDS_j^S$
	$RM_{10}(S) = \sum_j FS_j^S \times p_j \times NDS_j^S$
Slack sufficiency:	$RM_{11}(S) = \sum_j \lvert\{i \mid i \in prec_j^S \cup \{j\}, FS_j^S \geq \lambda p_i\}\rvert$
	$RM_{12}(S) = \sum_j \lvert\{i \mid i \in prec_j^S \cup \{j\}, FS_j^S < \lambda p_i\}\rvert$
	New, cost of being non-robust, RM_{11} favors spreading prec over machines
Interval schedule	$RM_{13}(S) = \max \sum_j (l_j - e_j)$, $[e_j; l_j]$ execution interval for job j, LP
	$RM_{14}(S) = \max \min_j (l_j - e_j)$, $[e_j; l_j]$ execution interval for job j, LP

Robustness measures: Normal approximation

Approximation based on normal distribution (Passage, vdA, Hoogeveen,2016):

- Assumption that all job times are normally distributed.
- Compute **normal distributions** of start and completion times.
  - Start time: $S_j = \max\{PST_j, \max_{i \prec j} C_i\}$
  - Completion time: $C_j = S_j + D_j$
  - Use order imposed by precedence relations and machine sequence
  - Find the expected value and variance of the maximum of 2 normal distributions in every step (Nadarajah and Kotz, 2008)

Robustness measures: Normal approximation

Approximation based on normal distribution (Passage, vdA, Hoogeveen,2016):

- Assumption that all job times are normally distributed.
- Compute **normal distributions** of start and completion times.
  - Start time: $S_j = \max\{PST_j, \max_{i \prec j} C_i\}$
  - Completion time: $C_j = S_j + D_j$
  - Use order imposed by precedence relations and machine sequence
  - Find the expected value and variance of the maximum of 2 normal distributions in every step (Nadarajah and Kotz, 2008)

Resulting robustness measures

- $RM_{15}(S) = P(\max_j CT_j \leq d)$, probability of meeting target d
- $RM_{16}(S) = \sum_j P(ST_j \leq PST_j^S)$ (new), total probability of jobs starting on time

Measures based on predecessor slack (new)

- $RM_{17}(S) = \sum_j \alpha_j$
  - α_j is the fraction of direct predecessors with a free slack of at least λp_i (and 1 if there are no direct predecessors)
- $RM_{18}(S) = \sum_j ESD_j^S$
  - total estimated start time delay for the case in which job j has a delay of exactly λp_j

Simulation study: Evaluating robustness measures

For each instance (we did 24):

1. Determine a collection of 970 baseline schedule (with slack)
   - Basic local search and rules to assign random slacks
2. Determine **correlation** between RMs and “true” robustness by simulation.

Simulation study: Evaluating robustness measures

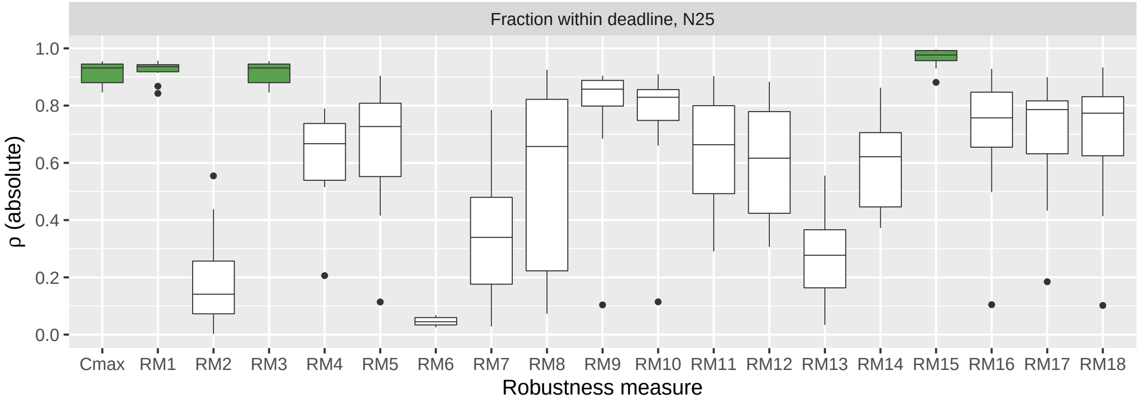
For each instance (we did 24):

1. Determine a collection of 970 baseline schedule (with slack)
   - Basic local search and rules to assign random slacks
2. Determine **correlation** between RMs and "true" robustness by simulation.

Determine correlation between RMs and "true" robustness.

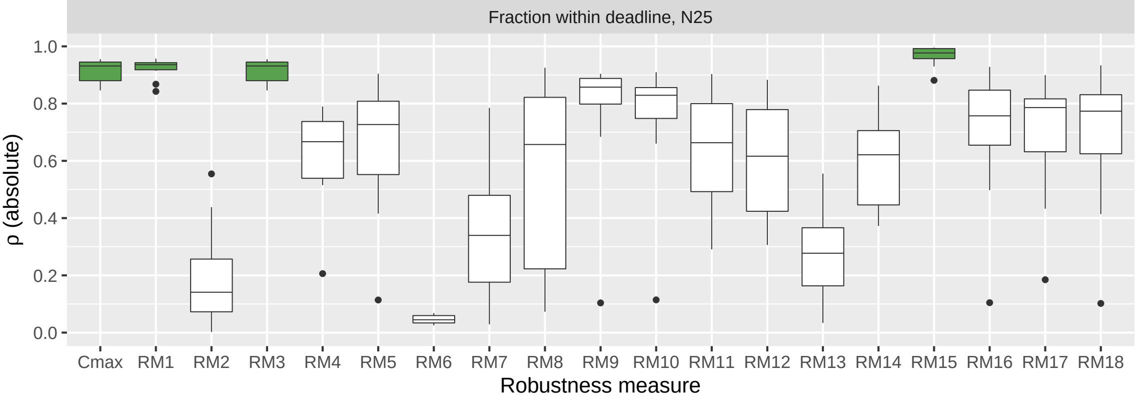
- Simulation to accurately estimate true robustness (1000 runs).
  - Average makespan (quality robustness).
  - Fraction of runs within deadline (quality robustness).
  - Average fraction on time jobs (solution robustness).
  - Average sum of job delays (solution robustness).
- Report Spearman's Rank correlation coefficient ρ

Simulation study: Results quality robustness



Spearman's Rank correlation coefficient ρ (absolute) for fraction within deadline. Green highlight is mean ≥ 0.9

Simulation study: Results quality robustness



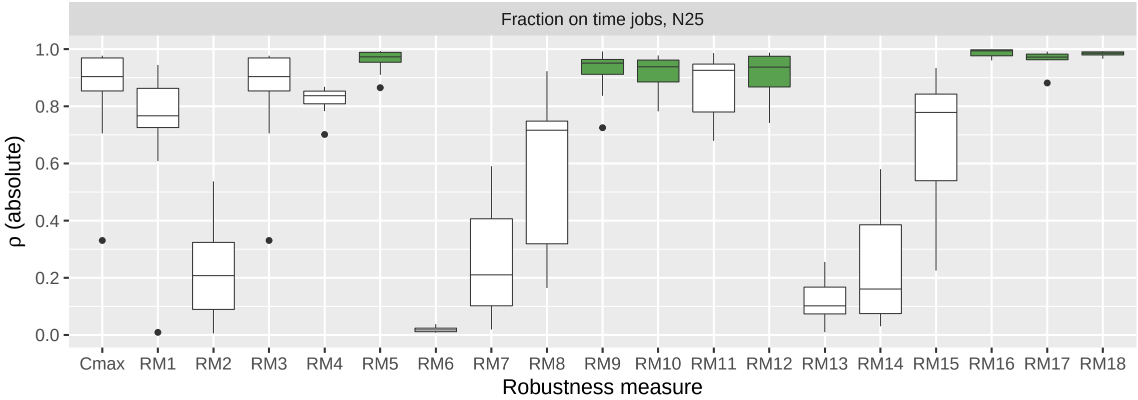
Spearman's Rank correlation coefficient ρ (absolute) for fraction within deadline. Green highlight is mean ≥ 0.9

Sum of total slack $RM_1 = \sum_j TS_j$

Minimum total slack (related to $C_{\max}$) $RM_3 = \min_j TS_j$

Normal approximated probability within deadline $RM_{15} = P(\max_j C_j \leq d)$

Simulation study: Results solution robustness



Spearman's Rank correlation coefficient ρ (absolute) for fraction on time jobs using distribution N_{25}. Green highlight indicates mean ≥ 0.9.

Best RMs solution robustness

$RM_5 = \sum_j \min\{FS_j, \lambda p_j\}$	Where λp_j increase in processing time we want to survive
$RM_{16} = \sum_j P(S_j \leq PST_j)$	Normal approximated probability of jobs starting on time
$RM_{17}(S) = \sum_j \alpha_j$	Where α_j is the fraction of direct predecessors with a free slack of at least λp_i (and 1 if there are no direct predecessor)
$RM_{18}(S) = \sum_j ESD_j^S$	Total estimated start time delay for the case in which job j has a delay of exactly λp_j
$RM_9 = \sum_j FS_j \times NDS_j$	Sum of free slack scaled by number of direct successors
$RM_{10} = \sum_j FS_j \times NDS_j \times p_j$	Sum of free slack scaled by number of direct successors and processing time

Conclusion from simulation study

- Normal approximation works well
- Total slack good measure for quality robustness
  - Minimum total slack (RM_3) is slack on critical path
  - Equivalent to makespan for given machine assignment and order
- Including λp_i as delay that we want to survive seems to work
- Free slack works best if weighted by number of direct successors or maximized at 'survivable' delay λp_i.

The Train Unit Shunting Problem (TUSP)

- Making a schedule for a shunting yard using local search (van den Broek, Hoogeveen, van den Akker, Huisman 2018).
  - Matching incoming to outgoing trains.
  - Combining and splitting train compositions.
  - Parking trains.
  - Routing trains.
  - Planning service activities.

The Train Unit Shunting Problem (TUSP)

- Making a schedule for a shunting yard using local search (van den Broek, Hoogeveen, van den Akker, Huisman 2018).
  - Matching incoming to outgoing trains.
  - Combining and splitting train compositions.
  - Parking trains.
  - Routing trains.
  - Planning service activities.

- Many of these tasks have stochastic durations!

The Train Unit Shunting Problem (TUSP)

- Making a schedule for a shunting yard using local search (van den Broek, Hoogeveen, van den Akker, Huisman 2018).
  - Matching incoming to outgoing trains.
  - Combining and splitting train compositions.
  - Parking trains.
  - Routing trains.
  - Planning service activities.

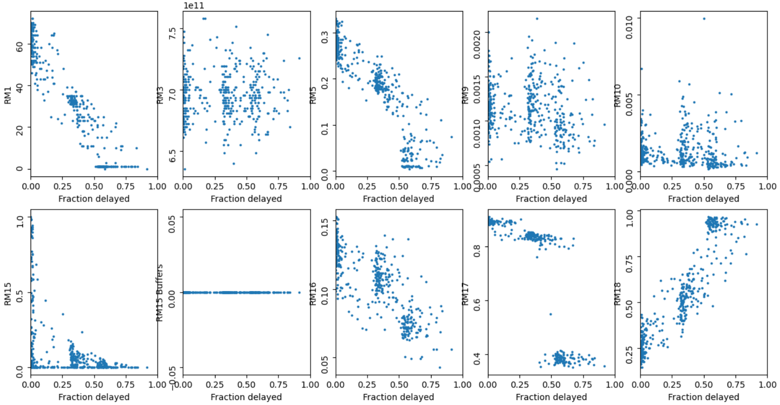
- Many of these tasks have stochastic durations!

Robustness measures for the TUSP

- Algorithm by Van den Broek et al. Gives us a full schedule including start times of jobs, and precedence constraints between different jobs.

Robustness measures for the TUSP

- Algorithm by Van den Broek et al. Gives us a full schedule including start times of jobs, and precedence constraints between different jobs.
- Use this schedule to calculate the robustness measures from earlier.



Local Search study

- Use measures with best correlation result as objective in Local search for TUSP.
  - Single Robustness measure.
  - Multiple Robustness measures (quality & solution).

Local Search study

- Use measures with best correlation result as objective in Local search for TUSP.
  - Single Robustness measure.
  - Multiple Robustness measures (quality & solution).
- Test for multiple Execution policies.
  - Left-aligned schedule.
  - planned start time.
  - At most X minutes before planned start time.

Local Search study

- Use measures with best correlation result as objective in Local search for TUSP.
  - Single Robustness measure.
  - Multiple Robustness measures (quality & solution).
- Test for multiple Execution policies.
  - Left-aligned schedule.
  - planned start time.
  - At most X minutes before planned start time.
- Measures based on normal approximation seem most effective (RM_{15}, RM_{16}).

Local Search study

- Use measures with best correlation result as objective in Local search for TUSP.
  - Single Robustness measure.
  - Multiple Robustness measures (quality & solution).
- Test for multiple Execution policies.
  - Left-aligned schedule.
  - planned start time.
  - At most X minutes before planned start time.
- Measures based on normal approximation seem most effective (RM_{15}, RM_{16}).
- Effectiveness of measures dependent on execution policy.

Local Search study

- Use measures with best correlation result as objective in Local search for TUSP.
  - Single Robustness measure.
  - Multiple Robustness measures (quality & solution).
- Test for multiple Execution policies.
  - Left-aligned schedule.
  - planned start time.
  - At most X minutes before planned start time.
- Measures based on normal approximation seem most effective (RM_{15}, RM_{16}).
- Effectiveness of measures dependent on execution policy.
- Solution robust measures can also increase the quality robustness.

Wrap up

Recap

- Elaborate simulation study to determine correlations of RMs with true robustness.
- A few new robustness measures
- Local search procedure for the TUSP with RM-based objective functions.

Wrap up

Recap

- Elaborate simulation study to determine correlations of RMs with true robustness.
- A few new robustness measures
- Local search procedure for the TUSP with RM-based objective functions.

Conclusion

- RMs can provide a good alternative to simulation for generating solution and quality robust schedules bounded by a deadline
- Success of normal approximation, using information about the processing time distribution helps
- Our new measure RM_{16} performs well in both the parallel machine scheduling setting as well as the TUSP setting.

References

- Loriana Pascual, supervisors: J.M. van den Akker, R.W. van den Broek. (2023) Improving Robustness For Stochastic Parallel Machine Scheduling With Robustness Measures. Master's thesis UU.
- J.M. van den Akker, C.H.M. van Blokland, and J.A. Hoogeveen (2013). Finding robust solutions for the stochastic job shop scheduling problem by including simulation in local search. In: V. Bonifaci et al (Eds.) Symposium on Experimental Algorithms 2013, Lecture Notes in Computer Science 7933, pp. 402-413, Springer.
- R.W. van den Broek, J.A. Hoogeveen, J.M. van den Akker, B. Huisman (2022) A Local Search Algorithm for Train Unit Shunting with Service Scheduling. Transportation Science Vol. 56, No. 1, pp. 141–161
- G.J.P.N. Passage, J.M. van den Akker, and J.A. Hoogeveen. "Local search for stochastic parallel machine scheduling: improving performance by estimating the makespan." European Conference on Stochastic Optimization. 2017.