Robustness Analysis for the Re-entrant Flow Shop Problem

Alexandru Bobe [1] Eghonghon Eigbe [1] Neil Yorke-Smith [1]

[1]Delft University of Technology

Introduction

- **Scheduling** has always been relevant to the real world and is present in almost every industry.
- Our **industry partner** is concerned with industrial printers.
- **Flexibility** is often forgotten when creating initial schedules.
- We analysed schedules represented as a re-entrant **flow shop** with relative due dates and sequence-dependent setup times.
- We created the means for robust scheduling with a list of steps for integrating a schedule in practice and a **new robustness measure**.

Background

- There exist several heuristics for generating initial schedules. However, their solutions lack a balance between productivity and flexibility.
- Some robustness measures have been created, but they do not work when idle time is inserted to increase the flexibility of the schedule.

Steps for Implementing Flexible Schedules

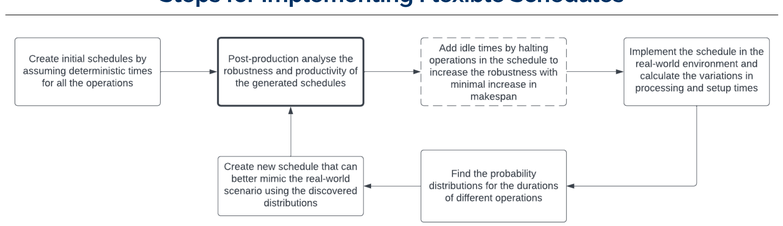

Figure 1. We were mainly interested in the first pass through the step with a bold border.

Robustness Analysis

- Started by bounding how much time a schedule has before it becomes infeasible.
- Continued by creating a new slack-based robustness measure:

$$\frac{\sum \left(D_{((j,o-1),(j,o))} - \left(Start_{(j,o)} - Start_{(j,o-1)} \right) \right)}{\sum D_{((j,o-1),(j,o))}} \quad (1)$$

- Implemented a known state-of-the-art technique [1] for measuring the robustness of a schedule to assess the validity of our solution.
- **The results of the experiments showed that our measurement works as expected even with added idle time, as opposed to the above technique.**
- Analysed over 800 schedules generated with the MPHCS algorithm [2].
- Found out the MPHCS algorithm is not reliable in generating schedules with a balance between productivity and flexibility.

Re-entrant Flow Shop

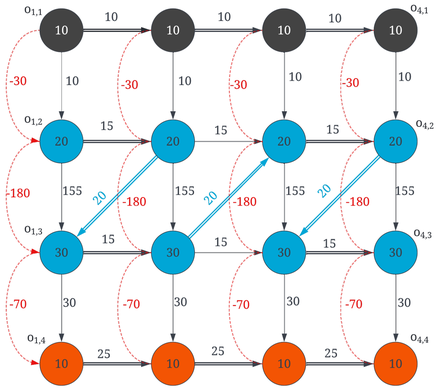

Figure 2. An example of a flow shop with three machines, four jobs and four operations. The operations that have the same colour are performed by the same machine, with the number inside the circles representing the processing times. The setup times are shown with black edges and the relative due dates are presented with red edges. The blue edges represent sequence-dependent setup times. The double edges illustrate a possible solution to this flow shop.

Analysis of the Generated Schedules

Even though there exists a flexibility term in the MPHCS algorithm, its behaviour related to our robustness measure is unpredictable.

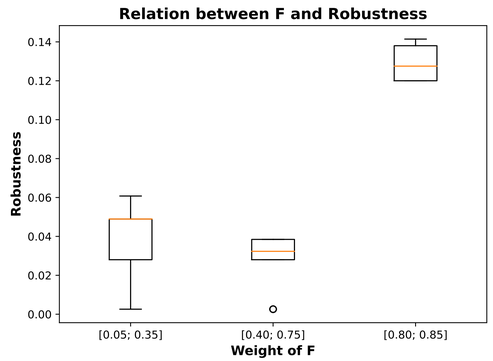

Figure 3. Box plots that illustrate the absence of linearity in robustness for schedules generated using the MPHCS heuristic

Experiments and Results

- We observed that both of the measures yielded similar results and our measure could be **used for** schedules with **500 jobs** without any unwanted value increase.

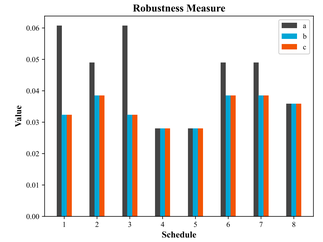

Figure 4. Comparison of different solutions for benchmarks with 100 jobs.

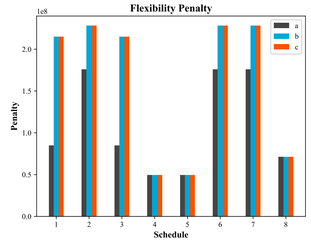

Figure 5. Comparison of different solutions for benchmarks with 100 jobs.

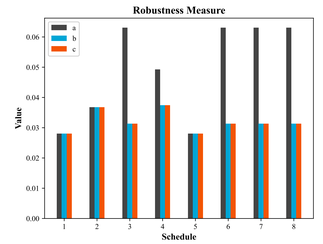

Figure 6. Comparison of different solutions for benchmarks with 500 jobs.

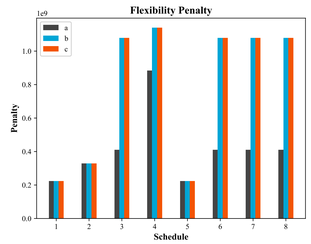

Figure 7. Comparison of different solutions for benchmarks with 500 jobs.

Conclusions and Future Work

- We empirically developed a new robustness measure.
- We confirmed that our robustness measure is as accurate as the previously known technique while having the ability to be used in flow shops with idle time inserted.
- For future studies, initial scheduling for re-entrant flow shops should take into account the possibility of inserting idle times to increase the flexibility.

References

[1] J. Branke and D. C. Mattfeld, "Anticipation and flexibility in dynamic scheduling," *International Journal of Production Research*, vol. 43, pp. 3103–3129, 8 2005.

[2] R. van der Tempel, J. van Pinxten, M. Geilen, and U. Waqas, "A heuristic for variable re-entrant scheduling problems," in *2018 21st Euromicro Conference on Digital System Design (DSD)*, pp. 336–341, 2018.