Multi-Objective Simulated Annealing for an industrial job shop scheduling problem in an online setting

Quentin Perrachon[1,2]
Alexandru-Liviu Olteanu[1]
Marc Sevaux[1]

[1] Lab-STICC, UMR 6285, CNRS, Université Bretagne Sud, Lorient, France
[2] HERAKLES, Vannes, France

1. Introduction

Scheduling optimization tools have the potential to improve production for many small to medium-sized manufacturing industries. However, the overhead cost and effort required to implement such tools may be too high to be practical. In order to propose a model that can be sued across multiple sectors, we extend the job shop scheduling problem (JSSP) by considering the following additional constraints:

- **Flexibility**: each assigned resource may be selected from a subset of resources
- **Non-linear routing**: multiple operation may precede or succeed another operation
- **Multi-Resource**: each operation may require more than one resource
- **Partially necessary resources**: some resources may not be required for the entire duration of an operation

To integrate these extensions into our model, we decompose operations into stages and allow resources to be required for only a subset of consecutive stages. This approach enables us to better handle the complexity of industrial problems and to provide solutions that are easier to implement.

2. Graph-based representation

We chose a directed acyclic graph model to represent our solutions (Figure 1). This graph corresponds to a precedence graph containing our operations as vertices, potentially divided into multiple vertices if the operation is composed of multiple stages, to which we add paths for each resource, going from a source node to a sink node and passing through every operation assigned to the corresponding resource. Based on the graph representation, we can now introduce a neighborhood based on reinsertions of operations [1] (Figure 2). This movement has been expanded to enable an operation to be reinserted simultaneously on multiple resources, which is crucial to prevent and move past certain deadlock formations in our solutions, particularly those involving 2 operations sharing multiple resources. However, due to the considerable size of the neighborhood, and the possibility of generating unfeasible solutions, a preliminary filter has been implemented based on several conditions on the current solution graph. Despite this filter, the neighborhood remains substantial, thus a simulated annealing-based approach has been proposed.

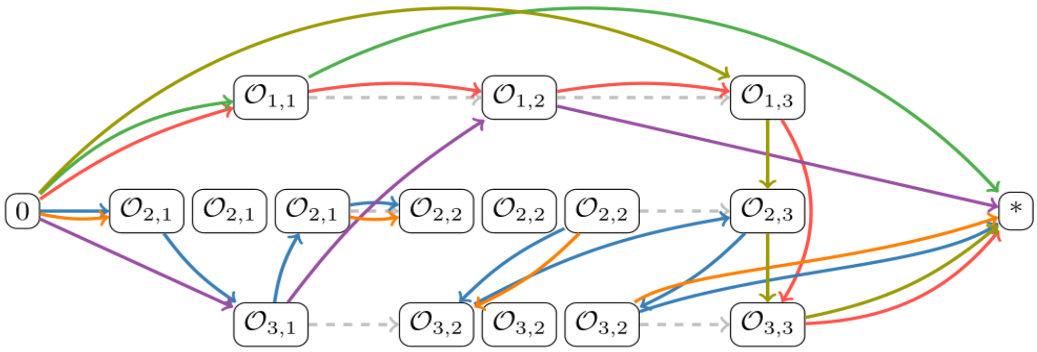

Figure 1: A directed acyclic graph representing a solution

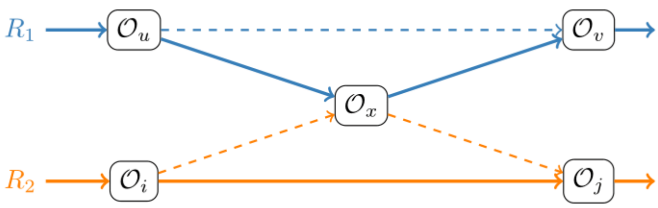

Figure 2: Reinsertion of an operation

3. Online rescheduling

Solving the problem on a fixed set of jobs isn't sufficient, as in practice, new jobs may arrive on regularly in a workshop. As a result, scheduling must account for new and top priority jobs, as well as those scheduled previously [2]. Given the uncertainty of new jobs arrival and their nature, prior decisions may need to be modified to achieve optimal outcomes. However, it is essential to minimize the adjustments to previous decisions, particularly decisions on the short-term schedule. To this goal, we propose a way to model the penalty incurred by a resource when moving one of its assigned tasks as the absolute difference between the projection of the original and new stat date on an exponential scale $f(t): e\hat{}(-t/H)$ with t the time being transposed to the exponential function and H a parameter of the function which we fix based on the makespan of

the previous schedule.
Considering one operation and one resource at a time, we consider two types of modification with st_{op} the starting time of an operation and st_{op}^{prev} the starting time of the same operation on the previous schedule.

- **An operation starts earlier or later on the same resource**: In this case, we compute the penalty applied as the difference: $\left|f\left(st_{op}^{prev}\right) - f(st_{op})\right|$ (Figure 3)
- **An operation has been assigned to a different resource:** In this case, we compute the penalty as the following sum: $f\left(st_{op}^{prev}\right) + f(st_{op})$

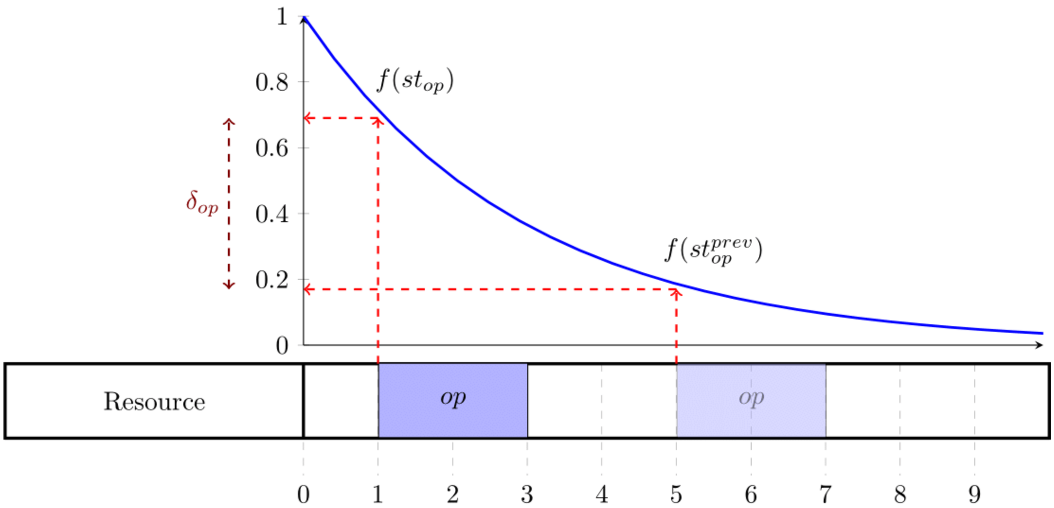

Figure 3: Penalty δ_{op} of operation op starting earlier than on the previous schedule

By computing the penalty this way, differences close to the start of the schedule results in higher penalties. It is worth noting that reassignment is generally more heavily penalized than a mere adjustment in timing. This is because reassignment is regarded as being more disruptive to the overall schedule. Operations and jobs finished before the previous schedule and the current rescheduling are no longer considered, moreover operations in progress at the start of the rescheduling are now fixed and cannot be moved or reassigned. By aggregating the penalties for each operation on each resource to which it is assigned, a quantification of the modifications made to a prior schedule is calculated by summing all the penalties incurred by every pair of operation and resource.

4. Multi-Objective Simulated Annealing

To properly responds to industrial demands, it is essential to consider multiple objectives when proposing solutions to them. In an online setting, on top of objectives related to industrial requirements, we will also want to minimize the deviations done to a schedule compared to its previous iteration using the method presented in the previous section as an objective. The three objective we will consider here are as follows:

- Minimize the sum of the duration of each job
- Minimize the maximum work load on a resource
- Minimize the deviation between the current schedule and the previous one

We present an approach that utilizes a population-based Multi-Objective Simulated Annealing [3] (MOSA), also known as Pareto Simulated Annealing (PSA) in the literature. Our algorithm employs a population of multiple solutions that explore the solution space in parallel via a simulated annealing procedure. To achieve an efficient exploration of the solution space, we introduce directional vectors to each solution that encourage them to move towards a specific direction in the solution space. The directional vectors are updated at each iteration by applying a repulsive force from the remaining population.

5. Perspectives

As of now, we only consider pairs of operations and resources one at a time to compute the penalties, considering every resource assigned to an operation at the same time or considering changes in the sequencing order of a resource may lead to a more precise quantification of the divergence between two schedules. Furthermore, we do not penalize adding operations to previously idle time at all, even though a sudden addition of operations to the immediate future may be disruptive to the proper functioning of a workshop.

6. References

[1] S. Dauzère-Pérès, W. Roux, et J. B. Lasserre, « Multi-resource shop scheduling with resource flexibility », *Eur. J. Oper. Res.*, vol. 107, nº 2, p. 289-305, juin 1998, doi: 10.1016/S0377-2217(97)00341-X.

[2] D. Gupta, C. T. Maravelias, et J. M. Wassick, « From rescheduling to online scheduling », *Chem. Eng. Res. Des.*, vol. 116, p. 83-97, déc. 2016, doi: 10.1016/j.cherd.2016.10.035.

[3] K. Amine, « Multiobjective simulated annealing: Principles and algorithm variants », *Adv. Oper. Res.*, vol. 2019, 2019.