Flexible Job Shop Optimization with Simulated Annealing

M.C. Bak (M.C.Bak@student.tudelft.nl) - Supervisors: K.C. van den Houten, M.M. de Weedt

Delft University of Technology

1. Background

Flexible Job Shop Problem

Optimization for process industries, done with a case study for a DSM production line. The production line is represented by a Flexible Job Shop Scheduling Problem (FJSP):

- Set of 3 **operations**;
- Distinctive set of **machines** for each operation;
- **Processing times** for the operations on the machines;
- N **jobs**, made up of 1 to 3 operations;
- **Changeover** times, for each machine an NxN matrix.

Goals

- Minimize the **makespan** of an FJSP instance;
- Find possible **bottlenecks**.

2. Research Questions

Is Simulated Annealing better suited for the FJSP than the MILP implementation?

- How well does the given **MILP** work of the **FSJP** instances?
- How is **Simulated Annealing** applied to the **FJSP**?
- What are effective **initialization** and **neighbourhood** functions for Simulated annealing?
- How does a SA optimization **compare** to the MILP formulation for the FJSP instances?
- Are there any **bottlenecks** in the production line?

References

[1] Monaldo Mastrolilli and Luca Maria Gambardella. "Effective neighbourhood functions for the flexible job shop problem". In: *Journal of Scheduling* 3.1 (Jan. 2000), pp. 3–20. ISSN: 1094-6136. DOI: https://doi.org/10.1002/(SICI)1099-1425(200001/02)3:1<3::AID-JOS32>3.0.CO;2-Y. URL: https://doi.org/10.1002/(SICI)1099-1425(200001/02)3:1%3C3::AID-JOS32%3E3.0.CO%202-Y.

[2] Yaghout Nourani and Bjarne Andresen. "A comparison of simulated annealing cooling strategies". In: *Journal of Physics A: Mathematical and General* 31.41 (Oct. 1998), pp. 8373–8385. DOI: 10.1088/0305-4470/31/41/011. URL: https://doi.org/10.1088/0305-4470/31/41/011.

[3] Shi Yang et al. "A novel initialization method for solving Flexible Job-shop Scheduling Problem". In: *2009 International Conference on Computers Industrial Engineering*. July 2009, pp. 68–73. DOI: 10.1109/ICCIE.2009.5223891.

3. Method

Simulated Annealing flow-chart

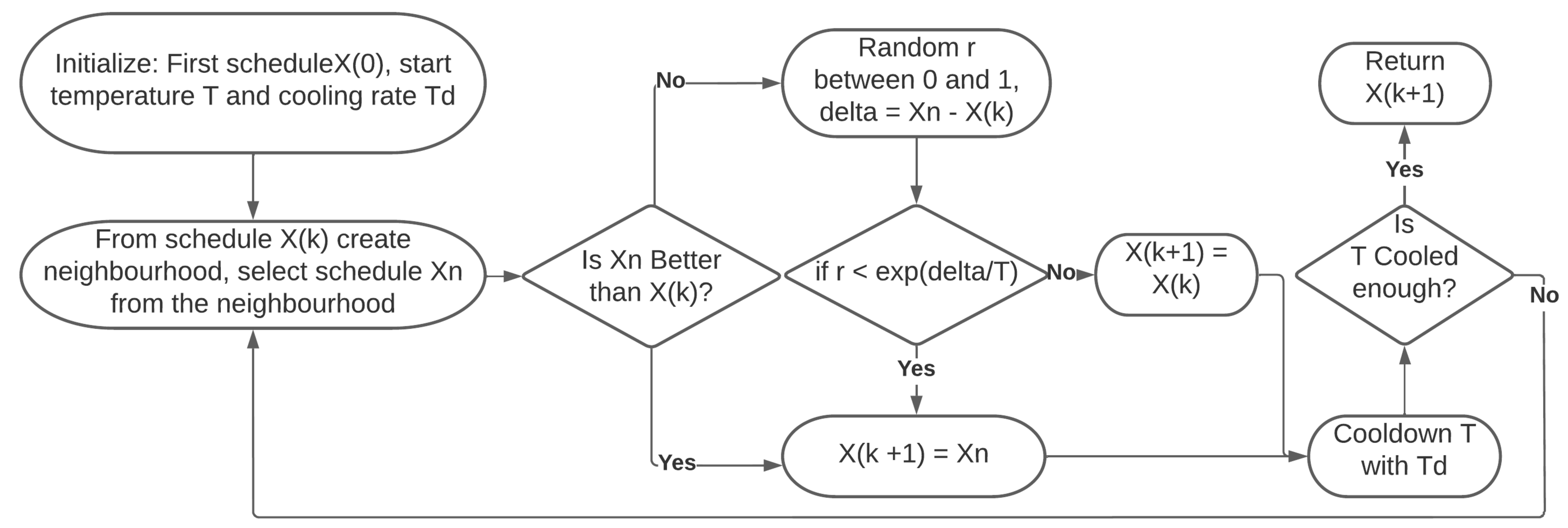

Fig. 1: Workflow of the Simulated Annealing algorithm.

Initialization

Global Selection is used to create the first schedule. This method is partially illustrated in Figure 2

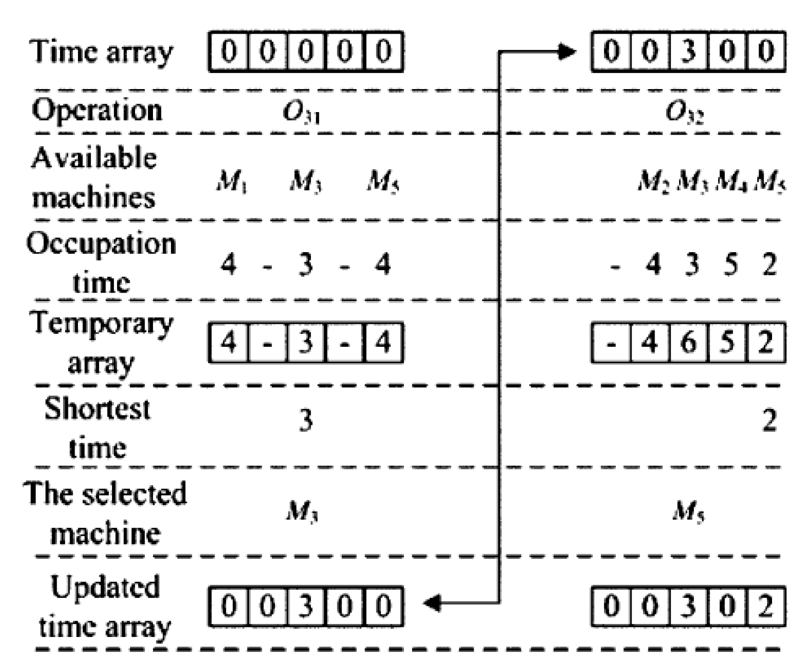

Fig. 2: Partial operation of the Global Selection method[3].

Neighbourhoods

For the neighbourhood generation a **graph representation** is used to move operations[1]. A complete graph is shown in Figure 3

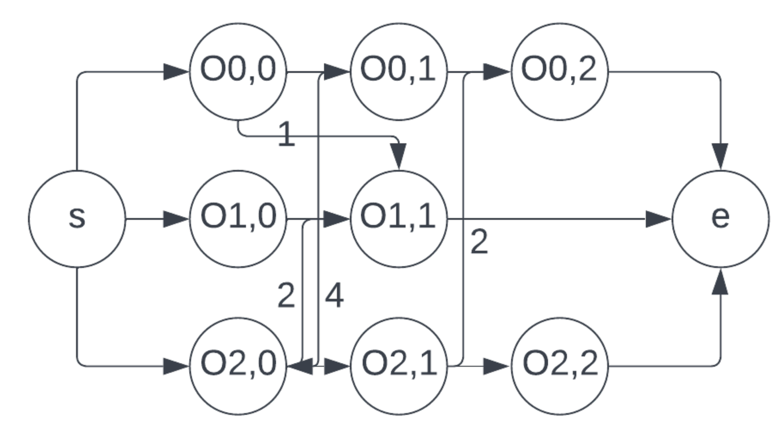

Fig. 3: A complete graph used for the neighbourhood creation.

Annealing

Annealing enables the algorithm to escape unfavourable local search spaces. The annealing of the algorithm is done with **exponential cooling**.

Formula for exponential cooling:

$T(t) = T_0 * \alpha^t$ [2].

The convergence is shown in Figure 4

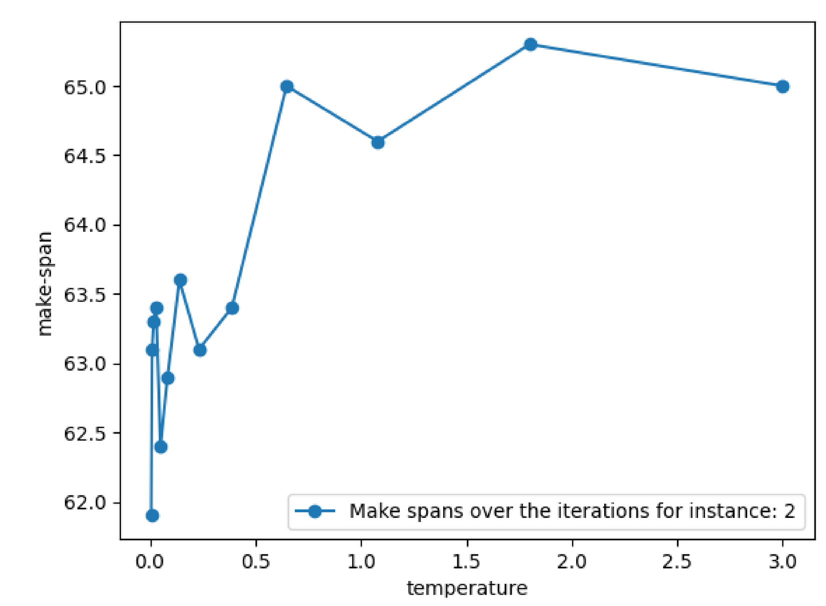

Fig. 4: Convergence plot of the annealing.

4. Results

MILP vs. Simulated Annealing

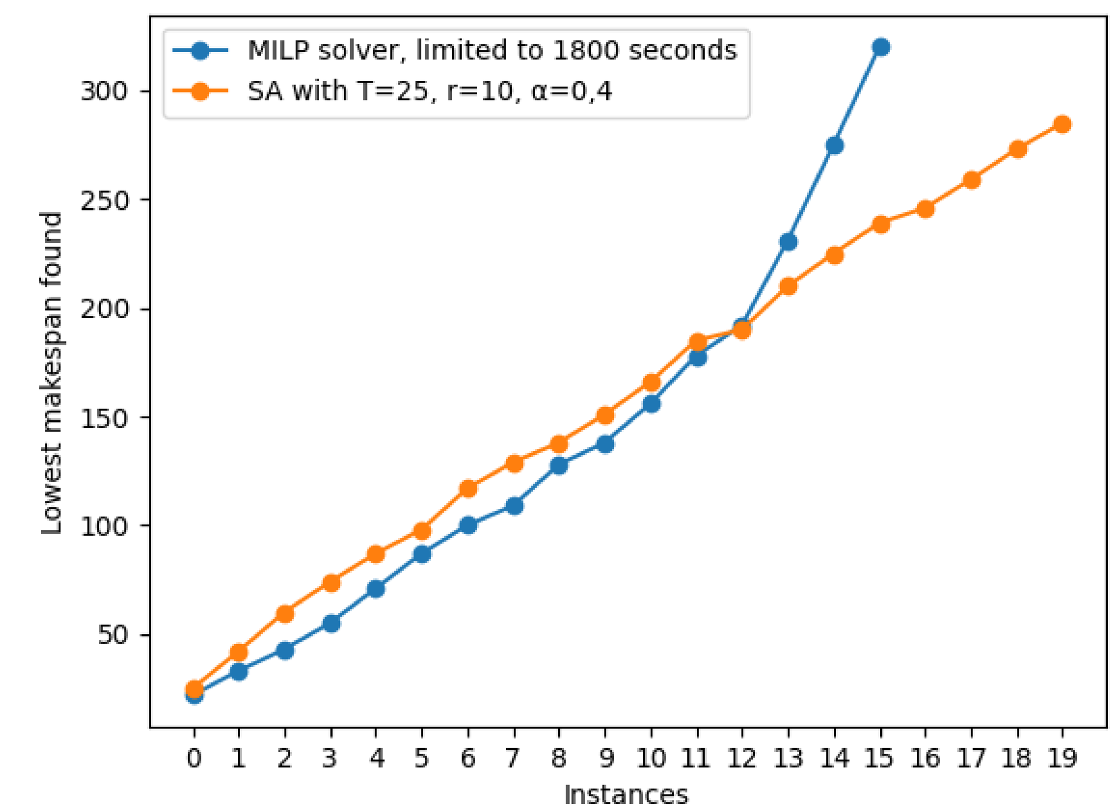

Fig. 5: Results of MILP and SA..9

Bottleneck

Results extra machine on operation 3					
Original	extra machine		Original	extra machine	
MILP-900	MILP-900	Improved	SA	SA	Improved
22	18	18%	25	19	24%
33	24	27%	42	36	14%
43	32	26%	57	50	12%
56	43	23%	71	60	15%
72	53	26%	86	77	10%
87	67	23%	99	86	13%
100	76	24%	112	97	13%
109	91	17%	129	108	16%
128	102	20%	141	115	18%
138	116	16%	156	127	19%
156	135	13%	169	139	18%
178	156	12%	176	155	12%
200	180	10%	191	159	17%
Average Improvement		20%	Average improvement		16%

Fig. 6: Improvements for extra machine for operation 3.

5. Conclusion and future work

Conclusion

- MILP is better for smaller instances (0-11);
- SA is better for bigger instances (12-19);
- Bottleneck found in operation 3.

Future work

- Remove **randomness** from Global Selection;
- Improve neighbourhood function.