Mathematical Formulations for JobShop Scheduling

Chew A., Jaffar J.

November 7, 2023

1 Data Initialization

Let i represents the task (k) index, and B represents the defined challenge bound.

$$\lfloor t_{k_i} \rfloor = \sum_{k_j} d_{k_j}$$

where $\lfloor t_{k_i} \rfloor$ is the lower bound for the starting time of task k_i, d_{k_j} is the duration of the predecessor task(s) of k_i within the same job n.

$$\lceil t_{k_i} \rceil = B - \sum_{k_j} d_{k_j}$$

where $\lceil t_{k_i} \rceil$ is the upper bound for the starting time of task k_i, d_{k_j} is the duration of the successor task(s) of k_i within the same job n.

2 Constraints

2.1 Disjunctions

For all tasks k_i and k_j on the same machine m, the following disjunction constraint is adhered.

$$(t_{k_i} + d_{k_i} \leq t_{k_j}) \vee (t_{k_j} + d_{k_j} \leq t_{k_i})$$

Modification of the above disjunction constraint results in the following possible conditions:

$$t_{k_i} \leq \alpha$$

$$t_{k_i} \geq \beta$$

where α and β are known constants. With respect to the above formulations, we consider two scenarios: (a) ***Case A:*** Task k_i has been assigned, and (b) ***Case B:*** Task k_i has not yet been assigned.

In the above ***Case A:***, if the original disjunction evaluates to FALSE, then backtrack, while in ***Case B:***, we consider the disjunction of $(t_{k_i} \leq \alpha) \vee (t_{k_i} \geq \beta)$. If both the bounds of t_i are violated, then the disjunction evaluates to FALSE. followed by initializing the backtracking process. If either bound is violated, then we choose the correct and unviolated bound and update it accordingly for task k_i.

3 Bounds Updating

We consider the scenarios where the bounds are updated for the remaining unassigned tasks, k_i on the: (a) SAME MACHINE, and (b) SAME JOB.

3.1 Same Machine

$$\lfloor t_{k_i} \rfloor = F(m) + \sum_{k_j} d_{k_j}$$

where if task k_j is not the predecessor of task k_i on the same machine m, then the duration of task k_j is not considered in the summation term above. Note that $F(m)$ represents the current time frontier of machine m.

$$\lceil t_{k_i} \rceil = B - F(m) - \sum_{k_j} d_{k_j}$$

where if task k_j is not the successor of task k_i on the same machine m, then the duration of task k_j is not considered in the summation term above.

Once the respective bounds for the relevant tasks have been updated, we check if any of them have the REJECT condition of $\lceil t_{k_i} \rceil < \lfloor t_{k_i} \rfloor$ for any of those tasks. If the REJECT condition is true, backtracking takes place.

3.2 Same Job

$$\lfloor t_{k_i} \rfloor = F(m) + \sum_{k_j} d_{k_j}$$

where k_j is the predecessor of k_i on a different machine m for task k_j.

$$\lceil t_{k_i} \rceil = B - F(m) - \sum_{k_j} d_{k_j}$$

where k_j is the successor of k_i on a different machine m for task k_j.