

# The Longest Processing Time rule for identical parallel machines revisited

Federico Della Croce

Politecnico di Torino, Italy

`federico.dellacroce@polito.it`

**Tutoriel de l'axe OPA (Ordonnancement, Planification et Applications) du GDR RO.**

**Abstract** We consider the  $P||C_{max}$  scheduling problem where the goal is to schedule  $n$  jobs on  $m$  identical parallel machines to minimize makespan. We revisit the famous Longest Processing Time (LPT) rule proposed by Graham in 1969. LPT requires to sort jobs in non-ascending order of processing times and then to assign one job at a time to the machine whose load is smallest so far. We provide new insights into LPT and discuss the approximation ratio of a modification of LPT that improves Graham's bound. We use linear programming to analyze the approximation ratio of our approach. This performance analysis can be seen as a valid alternative to formal proofs based on analytical derivation. Also, we derive from the proposed approach an  $O(n \log n)$  time complexity heuristic called SLACK. The heuristic splits the sorted job set in tuples of  $m$  consecutive jobs  $(1, \dots, m; m+1, \dots, 2m; \text{etc.})$  and sorts the tuples in non-increasing order of the difference (SLACK) between largest and smallest job in the tuple. Given this new ordering of the jobset, list scheduling is applied. This approach strongly outperforms LPT on benchmark literature instances and is competitive with more involved approaches such as COMBINE and LDM.