Automatic competitive analysis of real-time scheduling algorithms for firm-deadline tasks with non-preemptible sections and precedence constraints

Nico Schaumberger
Master programme Computer Engineering

Context and motivation
Extend existing framework for competitive analysis of real-time scheduling algorithms for firm-deadline tasks [CPKS18] by:
- non-preemptible sections
- precedence constraints
and implement a parallelized version of Madani’s algorithm [Mad02] in CUDA for GPUs to handle larger and more complex task sets.

Competitive ratio computation
Problem: Online- and offline algorithm work on potentially different job sequences due to dependencies.
Solution: Split job sequence into fixed part and dynamic part, which is calculated with release functions:

$$CR_J(A) = \inf_{(\sigma_J \cup \sigma_T) \in J} \lim_{k \to \infty} \frac{1 + V(\pi_A(\sigma_J \cup \sigma_T), k)}{1 + V(\pi_C(\sigma_J \cup \sigma_T), k)}$$

Event-based precedences
A dependent task instance is released on completion of its precursor task, with no other possible task releases.

Release functions as LTS
Release functions can be expressed as labeled transition systems, where edges represent precursor task completions and dependent task releases:

Time-based precedences
A dependent task instances can be released any time, but can be paired or unpaired based on completion status of its precursor task. Some only have paired instances.

Results
Evaluating 8 well-known online algorithms on 4 task sets inspired by representative use cases:

References