Impact of job dropping on the probabilistic schedulability of uniprocessor deterministic real-time systems

Olivier Buffet & *Liliana Cucu-Grosjean*
firstname.lastname@inria.fr

INRIA Nancy Grand-Est

EVOLVE – May 2011

Model and Associated Notations

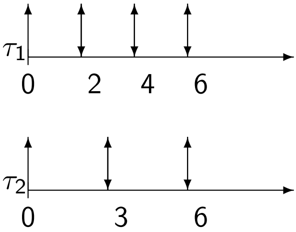
A set of n constrained deadline periodic tasks τ_i where:

- O_i is the *offset* of task τ_i;
- C_i is the *worst-case execution time* (WCET) of task τ_i;
- T_i is the *period* of task τ_i;
- D_i is the *deadline* of task τ_i;
- p_i is the *minimum success rate* of task τ_i.

Problem: schedule n such tasks on a processor such that they respect their respective minimum success rates.

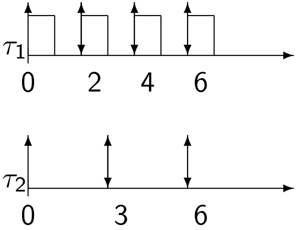
What is uniprocessor fixed-priority scheduling?

$\tau_i =$(offset, execution time, period, deadline, ratio miss)
$\tau_1 = (0, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 100\%)$



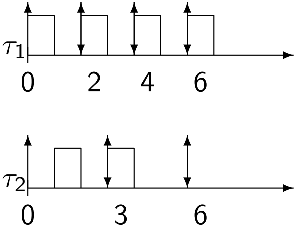
What is uniprocessor fixed-priority scheduling?

$\tau_i =$(offset, execution time, period, deadline, ratio miss)
$\tau_1 = (0, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 100\%)$



What is uniprocessor fixed-priority scheduling?

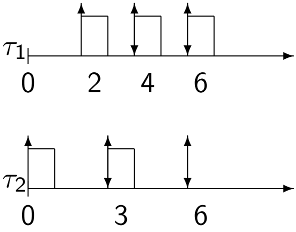
$\tau_i =$(offset, execution time, period, deadline, ratio miss)
$\tau_1 = (0, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 100\%)$



How do we check that a schedule is feasible?

Feasibility interval for a task set $\tau = \{\tau_1, \cdots, \tau_n\}$

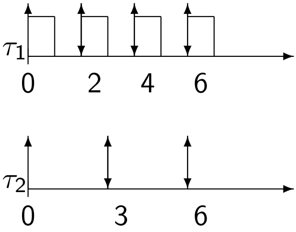
- $S_1 = O_1$;
- $S_i = \max\{O_i, O_{i-1} + \lceil \frac{S_{i-1} - O_i}{T_i} T_i \rceil\}, \forall i > 1.$
- $\tau_1 = (2, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 50\%)$.



$S_1 = 2$ and $S_2 = 3$. The feasibility interval is $[3, 9]$.

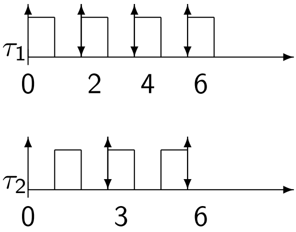
Meeting deadlines at 100% is not always possible

τ_i =(offset, execution time, period, deadline, ratio miss)
$\tau_1 = (0, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 100\%)$



Meeting deadlines at 100% is not always possible

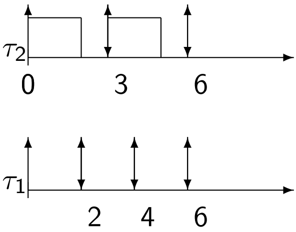
τ_i =(offset, execution time, period, deadline, ratio miss)
$\tau_1 = (0, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 100\%)$



τ_2 meets only 50% of its deadlines.

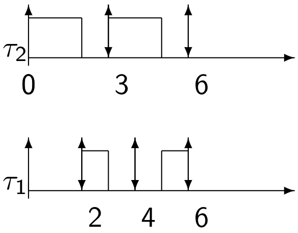
And meeting the imposed missing rates could be also difficult

τ_i =(offset, execution time, period, deadline, ratio miss)
$\tau_1 = (0, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 50\%)$



And meeting the imposed missing rates could be also difficult

τ_i =(offset, execution time, period, deadline, ratio miss)
$\tau_1 = (0, 1, 2, 2, 100\%)$ and $\tau_2 = (0, 1, 3, 3, 50\%)$



τ_2 meets 100% of its deadlines and τ_1 meets only 33% of its deadlines.

Minimal job dropping

Drop a job if it is doomed to fail

- Not starting a job if there is not enough time left for it;
- Stopping a job if some information tells us that it won't be able to finish on time.

Basic Test: remaining time for a job;
Advanced Test: droppability of higher priority jobs: chain reaction.

Minimal job dropping: preliminary results

Two propositions:

Compatibility with fixed priority: minimal job dropping (minJD) increases the succes rates for a system scheduled according a fixed priority policy (FP);

Periodicity of minJD+FP: when using fixed priorities and mininmal job dropping, any feasible schedule is periodic and it repeats every hyperperiod of lcm of the periods.

Conclusions

We have formulated the problem of real-time tasks with probability miss ratio.

We have proposed feasibility results and a job dropping mechanism.

Thank you for your attention

liliana.cucu@inria.fr