A Note on Some Open Problems in Mixed-Criticality Scheduling

Pontus Ekberg & Wang Yi

Uppsala University

RTSOPS 2015
The setting

- “Vestal-type” mixed-criticality sporadic tasks.
- Two criticality levels (LO/HI).
- A preemptive uniprocessor.
Why is MC scheduling so hard?

Job sequences

The SAS
Why is MC scheduling so hard?

It is difficult to identify the worst cases!
Why is MC scheduling so hard?

It is difficult to identify the worst cases!
Job sequences

- Fixed sequences specifying *release times* of all jobs in a given runtime scenario.
- Execution times still unknown.
Claim 1

The synchronous arrival sequence is *not* a guaranteed worst case for sporadic MC tasks.
Claim 1

Job sequences

The SAS
In general, it is not enough to consider integer-valued release times when analyzing sporadic MC tasks.
Claim 2

With rational release times

With integer release times
For a given MC sporadic task set, can we \textit{efficiently find} some \textit{small set of job sequences}, such that it is online schedulable iff all job sequences in that set are online schedulable?
Are job sequences the right abstraction?
Are job sequences the right abstraction?
Are job sequences the right abstraction?

Common prefix
Are job sequences the right abstraction?
Are job sequences the right abstraction?

Common prefix
Are job sequences the right abstraction?
Are job sequences the right abstraction?
Is there an infeasible MC sporadic task set, such that all its job sequences are schedulable?
Reevaluating the choice of task model?

• Direct MC extension of ordinary (strictly) periodic tasks.
• Synchronous or asynchronous.
• Each task set can generate only a single job sequence!
Reevaluating the choice of task model?

• Direct MC extension of ordinary (strictly) periodic tasks.
• Synchronous or asynchronous.
• Each task set can generate only a single job sequence!
Is scheduling or analysis for MC periodic tasks significantly easier than for MC sporadic tasks?
Claim 1: The SAS is not a worst-case job sequence.
Claim 2: Rational release times can be worse than integer release times.
Claim 1: The SAS is not a worst-case job sequence.
Claim 2: Rational release times can be worse than integer release times.

Question 1: Is there a (reasonable) replacement for the SAS?
Question 2: Is it enough to look at job sequences?
Question 3: Is it easier to analyze and/or to schedule MC periodic tasks than MC sporadic tasks?