Fixed-Priority scheduling of dual-criticality systems

Sanjoy Baruah and Alan Burns

Fixed-Priority scheduling of dual-criticality systems

Sanjoy Baruah and Alan Burns

1. Context and Background
2. FP scheduling and mixed criticalities: A survey
3. Period Transformation (PT) And mixed criticalities

Context and Background

Requirements in real-time systems design

Correctness - by using rigorous design methodologies

Constrain the system designer to only choose safe designs

"Freedom from choice" – A. Sangiovanni-Vincentelli

Efficiency - by the application of scheduling theory

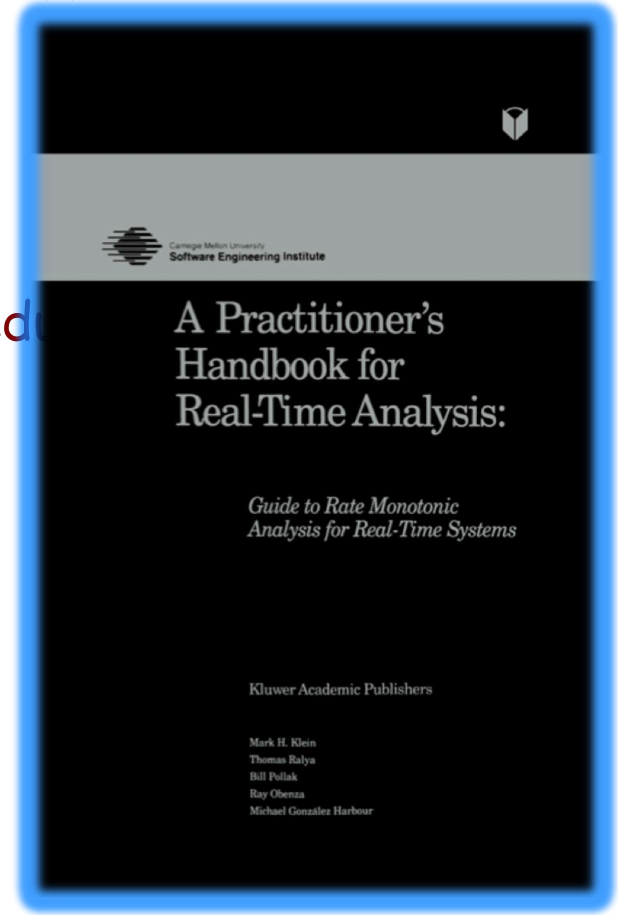
Requirements in real-time systems design

Correctness - by using rigorous design methodologies

Fixed Priority (FP) scheduling: the RMA approach

- Formalized industrial practice
- Based on strong mathematical foundations
- Good out-reach (and public relations! ☺)

Efficiency - by the application of sched



Requirements in real-time systems design

Correctness - by using rigorous design methodologies

Fixed Priority (FP) scheduling: the RMA approach

- Formalized industrial practice
- Based on strong mathematical foundations
- Good out-reach (and public relations! ☺)

Efficiency - by the application of scheduling theory

Increase efficiency:

"Better" priority assignment: Rate/ Deadline Monotonic

Choosing the periods appropriately: harmonic task systems

A complicating factor

RT systems becoming more complex ➡ implemented upon COTS platforms

COTS platforms are nondeterministic at run-time

Example: x := a + b on the Motorola PowerPC 755

- Best case: 3 cycles
- Worst case: 321 cycles

In the 1990's: the Motorola 64K had best case = worst case = 20 cycles

Resource over-provisioning to deal with unpredictability

Resource over-provisioning causes run-time inefficiencies

Mixed-criticality systems

Steve Vestal. *Preemptive scheduling of multi-criticality systems with varying degrees of execution time assurance.* RTSS 2007

Functionalities of different criticalities upon a shared platform

- use resource over-provisioning to validate highly critical functionalities at high levels of assurance

- design-time resource reclamation to validate less critical functionalities at lower levels of assurance

A **NEW** scheduling problem (finally! ☺)

Fixed-Priority scheduling of Mixed-Criticality systems: A review

Priority assignment schemes

Collection of mixed-criticality sporadic tasks executing on a single preemptive processor

Each task τ_i:

- HI or LO criticality
- Relative deadline D_i
- Minimum inter-arrival separation ("period") T_i
- Two WCETs: $C_i(LO)$ and $C_i(HI)$, with $C_i(LO) \leq C_i(HI)$

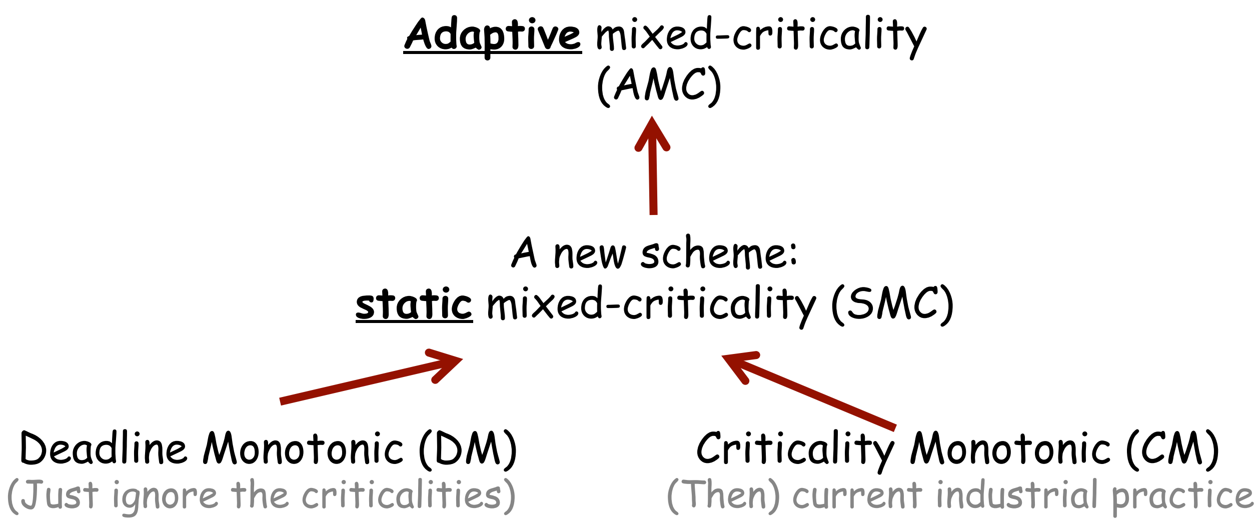
Determine a single scheduling strategy

- If each job of each τ_i completes in $\leq C_i(LO)$ units of execution, then all jobs of all tasks complete by their deadlines

- If each job of each τ_i completes in $\leq C_i(HI)$ units of execution, then all jobs of HI-criticality tasks complete by their deadlines

Vestal. Preemptive scheduling of multi-criticality systems with varying degrees of execution time assurance. RTSS 2007

Priority assignment schemes



Baruah, Burns, and Davis. Response-time analysis for mixed-criticality systems. RTSS 2011

Vestal. Preemptive scheduling of multi-criticality systems with varying degrees of execution time assurance. RTSS 2007

Period Transformation

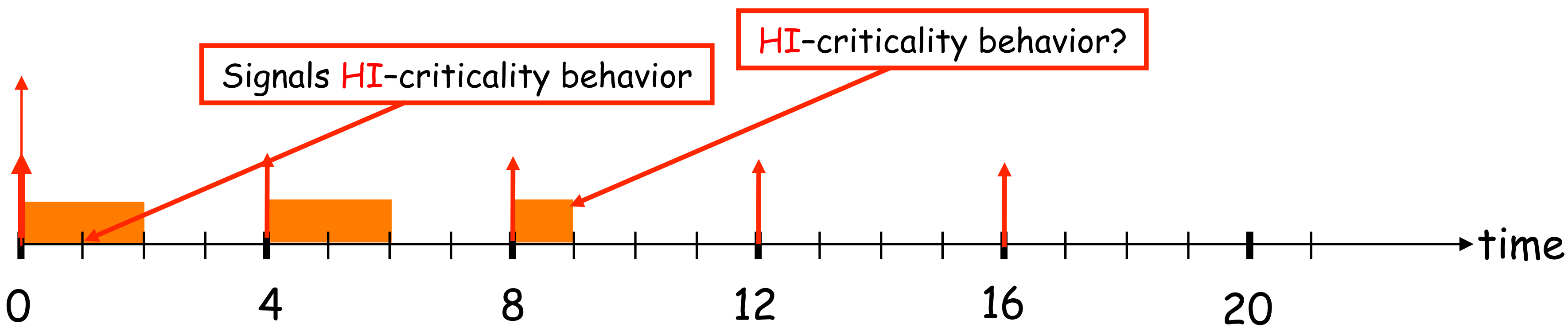
vs

Adaptive Mixed Criticality

Period transformation (PT): Scale down the parameters of HI-criticality tasks to make a DM priority assignment criticality-monotonic

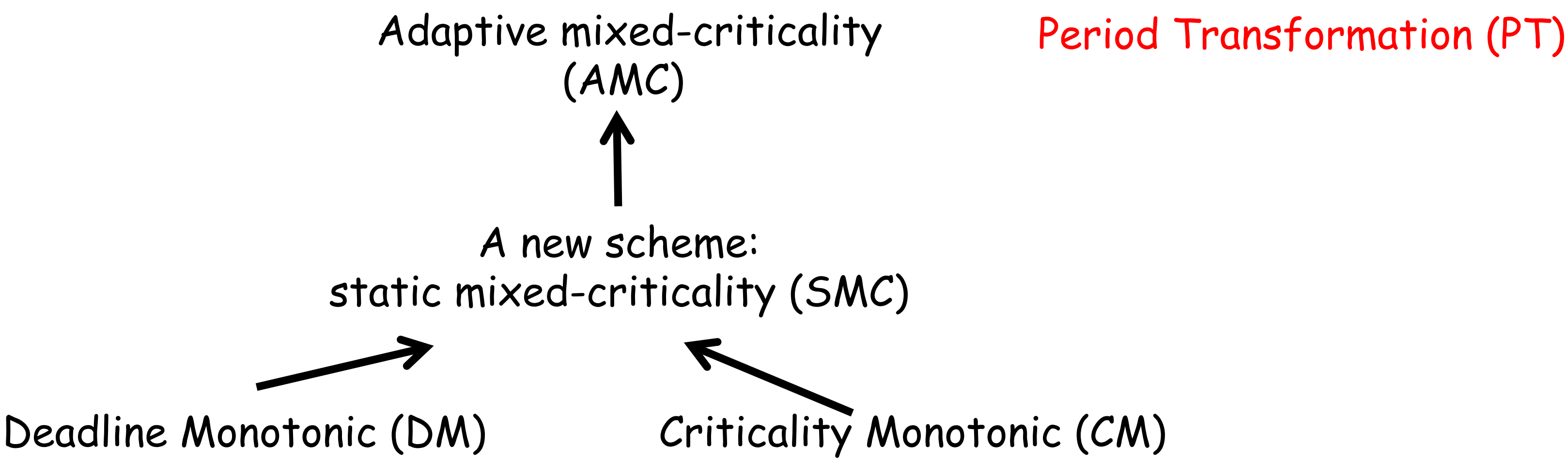
	criticality	$C_i(LO)$	$C_i(HI)$	$D_i = T_i$
τ_1	HI	~~5~~ 1	~~10~~ 2	~~20~~ 4
τ_2	LO	2	2	4

Scheduling the original task is not equivalent to scheduling a mixed-criticality task with the scaled-down parameters!!



Observations - 1

Result: AMC and PT are incomparable: each priority assignment scheme can schedule instances that the other cannot



Observations - 2

Result: AMC and PT are incomparable: each priority assignment scheme can schedule instances that the other cannot

PT on mixed-criticality instances

- Increase priority of HI tasks
- Improve utilization bound (as on non-MC tasks)

By making task periods harmonic

Result: AMC dominates PT for harmonic task systems

Context and Conclusions

Correctness and efficiency concerns in real-time systems

Design methodologies facilitate correctness

Scheduling theory: improve design methodologies to achieve efficiency

The Fixed-Priority (RMA) design methodology

- platforms increasingly unpredictable ➡ mixed-criticality implementations
- A survey of fixed-priority approaches to MC scheduling
- A comparative evaluation of the Period Transformation (PT) approach