# An Exact Schedulability Analysis for Global Fixed-Priority Scheduling of the AER Task Model

Thilanka Thilakasiri, Matthias Becker

KTH Royal Institute of Technology

January 20, 2023

# Overview

1. Introduction
2. The Proposed Exact Schedulability Analysis
3. Experiments and Results
4. Summary

# Introduction

Safety-critical systems needs to be carefully designed to meet the specific requirements they have as catastrophic consequences can be resulted otherwise.

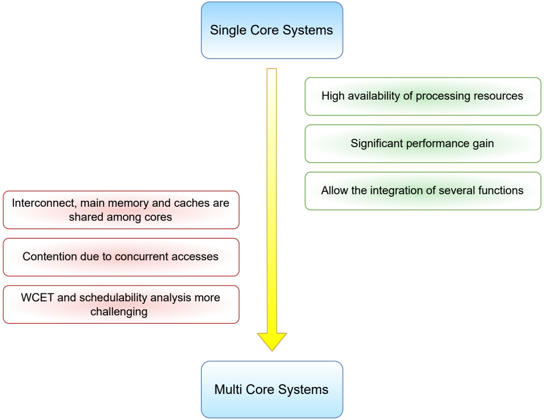
Examples for safety-critical systems,

- Avionics applications
- Automotive applications

Timing verification becomes crucial in these systems.

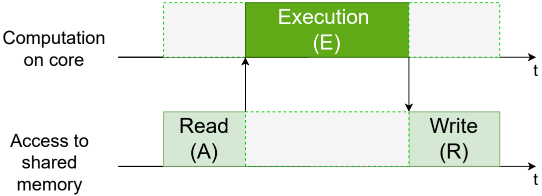
# Introduction

Shifting to multi-core processors is considered as the only solution to the performance limitation challenge faced with single-core systems.



# Phased-Execution Task Models

- Phased-execution models are a good solution when addressing these challenges.
- The PRedictable Execution Model (PREM) [5] and the AER Model [2] falls under this category.
- In the AER Model, a task is divided into 3 phases.
  1. **Acquisition** - Read and copy all necessary data and instructions for the task from the main memory to the local memory
  2. **Execution** - Execution of the task without having to access the main memory
  3. **Restitution** - Write-back the results to the main memory after the execution



# Schedulability Analysis for the AER Model

- Schedulability analysis for the AER model under global fixed-priority scheduling has been proposed in [1] and [4].
- However, these schedulability tests are non-exact and only sufficient.
- Due to their pessimism they may classify a schedulable task set as unschedulable.

★ An *exact* schedulability test provides the result as unschedulable *only* when it is not actually schedulable.

★ Therefore, having an exact schedulability test is needed to correctly evaluate the schedulability of a task set.

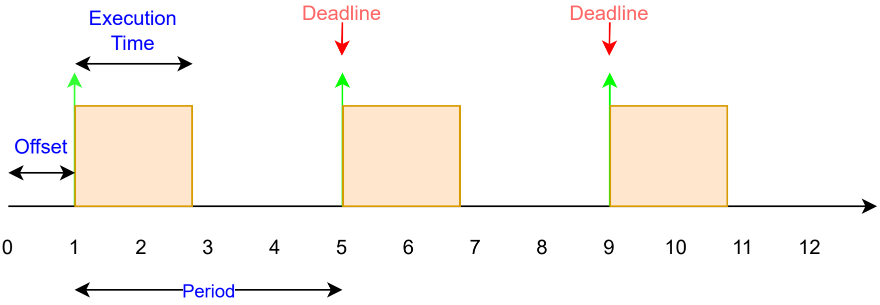
★ Also, an exact test provides a reference for the comparison of the accuracy of the existing schedulability tests.

# Contribution

- Our work introduces the first exact schedulability test for the AER model under global fixed-priority scheduling.
- Our schedulability test utilizes timed automata (TA) where the schedulability problem is described as a reachability problem.
- Evaluation of the proposed analysis using synthetic task sets against the state-of-the-art, which shows that the proposed analysis provides up to 65% more schedulable task sets than the state-of-the-art, while providing acceptable solving times.

# System Model

- The multiprocessor system we consider has *m identical* cores.
- The task set is comprised of *n independent* tasks $\tau = \{\tau_1, \tau_2, ..., \tau_n\}$.
- Each task $\tau_i$ can be represented by the tuple $(T_i, D_i, O_i, P_i, C_i)$.




A task set is considered schedulable when all of its tasks finish before their deadlines.

# Task Model

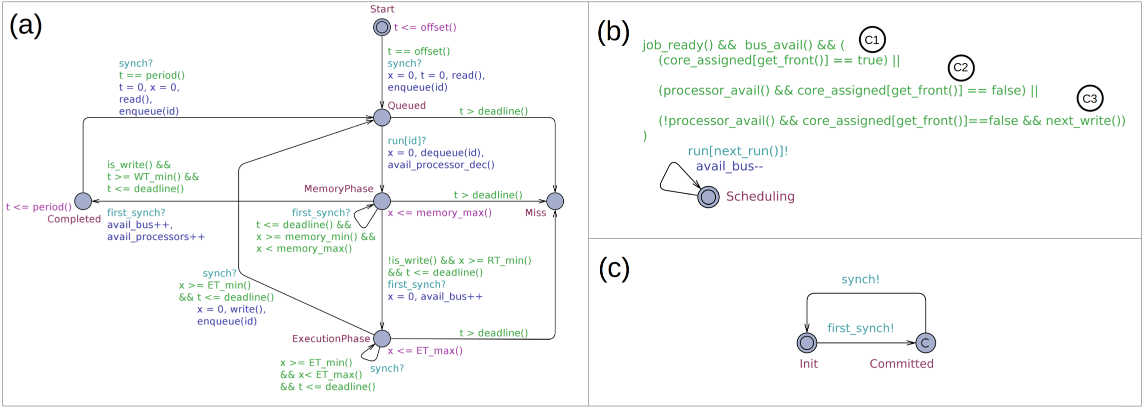
- The total execution time, $C_i = C_i^r + C_i^e + C_i^w$
- The AER model executes each phase of the task without any interference from other tasks executing on other cores.
- However, $C_i^x \in [C_i^{x_{min}}, C_i^{x_{max}}]$, where $x = \{r, e, w\}$
- Therefore, the total execution time, $C_i \in [C_i^{min}, C_i^{max}]$

# Runtime Execution Model

- We assume global non-preemptive fixed-priority scheduling.
- The tasks are scheduled from a memory perspective.
- The memory phases are added to a common global priority queue until they are scheduled.
- When the ready queue is not empty and the bus is available, 3 conditions are checked.

  **C1** - If the task at the front wants its write phase to be scheduled.
  **C2** - If the task at the front wants its read phase to be scheduled and if a core is available.
  **C3** - If the task at the front wants its read phase to be scheduled and if a core is not available.

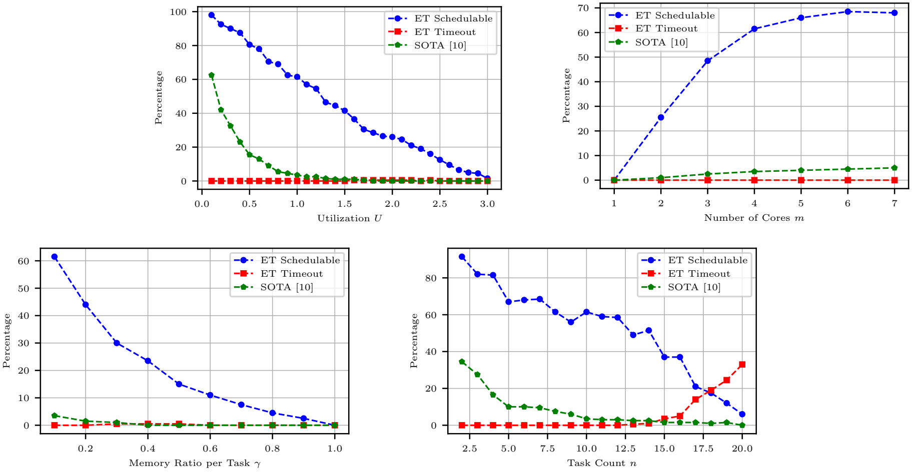
# Network of Timed-Automata in UPPAAL



# Experiments and Task Set Generation

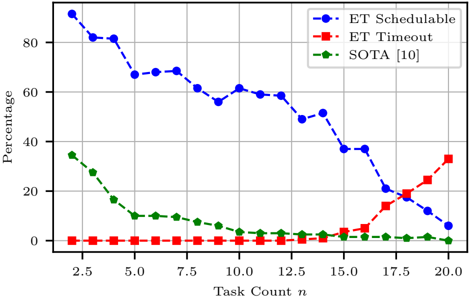
1. The common period values reported by Kramer et al. [3] for automotive applications were used.
   - Period values for tasks ($T_i$): $\{1, 2, 5, 10, 20, 50, 100, 200, 1000\} ms$
   - Respective probability percentages: $\{3, 2, 2, 25, 25, 3, 20, 1, 4\}$
2. For a given total utilization $U$, the Dirichlet-Rescale (DRS) algorithm is used to randomly generate $u_i$ utilization values for each task in the task set.

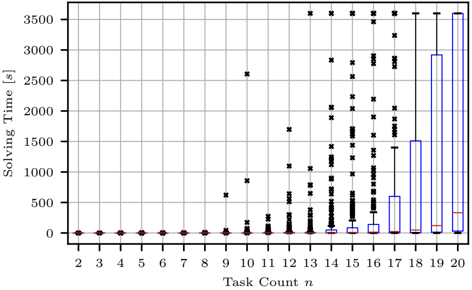
# Schedulability Evaluation




- Basic configuration : $U = 1, m = 4, \gamma = 0.1$ and $n = 10$
- Each plotted point represents 200 task sets.

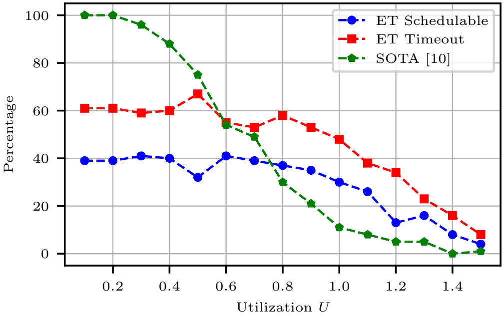
# Runtime Evaluation




- Basic configuration : $U = 1, m = 4, \gamma = 0.1$
- Each plotted point represents 200 task sets.

# Log-uniform Period Distribution




- Basic configuration : $m = 4, \gamma = 0.1, n = 10$
- Log-uniform period distribution between 10ms and 100ms
- Each plotted point represents 100 task sets.

# Conclusions

1. We present an exact schedulability test using timed automata for the globally scheduled AER model.
2. The presented schedulability test reduces pessimism of the existing tests by a large margin.
3. It provides a baseline against which other tests can be compared.

# Reference

[1] Ahmed Alhammad and Rodolfo Pellizzoni. "Schedulability analysis of global memory-predictable scheduling". In: *Proceedings of the 14th International Conference on Embedded Software*. 2014, pp. 1–10.

[2] Guy Durrieu et al. "Predictable flight management system implementation on a multicore processor". In: *Embedded Real Time Software (ERTS'14)*. 2014.

[3] Simon Kramer, Dirk Ziegenbein, and Arne Hamann. "Real world automotive benchmarks for free". In: *6th International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS)*. Vol. 130. 2015.

[4] Cláudio Maia et al. "Schedulability analysis for global fixed-priority scheduling of the 3-phase task model". In: *2017 IEEE 23rd International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. IEEE. 2017, pp. 1–10.

[5] Rodolfo Pellizzoni et al. "A predictable execution model for COTS-based embedded systems". In: *2011 17th IEEE Real-Time and Embedded Technology and Applications Symposium*. IEEE. 2011, pp. 269–279.


thank you
thank you
thank you
thank you