Work-Already-Published: Reliability Optimization on Multi-Core Systems with Multi-Tasking and Redundant Multi-Threading

Kuan-Hsun Chen, Georg von der Brüggen, Jian-Jia Chen

Department of Informatics
TU Dortmund University, Germany

11th of April 2018

This research is:

- supported in parts by the German Research Foundation (DFG) as part of the collaborative research center SFB876, subproject B2.
Introduction

Redundant Multi-threading (RMT) have been widely used.
- Simultaneous Redundant Threading (SRT) and
- Chip-level Redundant Multithreading (CRT)
- *Mixed Redundant Threading (MRT)* for balancing the costs

How to apply RMT while satisfying given design constraints?
- $M$ homogeneous cores connected by a communication fabric
- $N$ sporadic tasks with implicit deadline
- Preemptive fixed-priority scheduling

![Diagram of System Software, Scheduler in OS, and Multi-Core Fabric Architecture]

Jian-Jia Chen (TU Dortmund, DE)
Why we prefer MRT?

Figure: Each level has its own reliability penalty, which describes the probability that a fault during the execution leads to a visible error.

(a) SRT is not feasible.

(b) MRT can activate TMR.

Figure: Triple Modular Redundancy-based RMT on two cores.
Summary and Takeaways

- Mixed-Redundant Threading balances the usage of resource.

- We solve the problem via dynamic programmings with Federated Scheduling.

- Our approach outperforms the greedy approach in terms of the system reliability and the feasibility.

- This study is limited to implicit-deadline real-time tasks under federated scheduling.