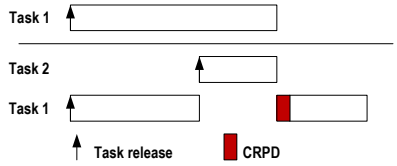
Integration of Cache Related Preemption Delay Analysis into a Priority Assignment Algorithm

Hai-Nam Tran, Frank Singhoff, Stéphane Rubini and Jalil Boukhobza
Univ. Bretagne Occidentale, UMR 6285, Lab-STICC, 29200 Brest, France
Email: hai-nam.tran,singhoff,rubini,boukhobza@univ-brest.fr

I. INTRODUCTION

Handling cache related preemption delay (CRPD) in preemptive scheduling context for real-time systems stays an open subject despite of its practical importance. We propose an approach to take into account the CRPD when performing priority assignment.



II. PROBLEM STATEMENT

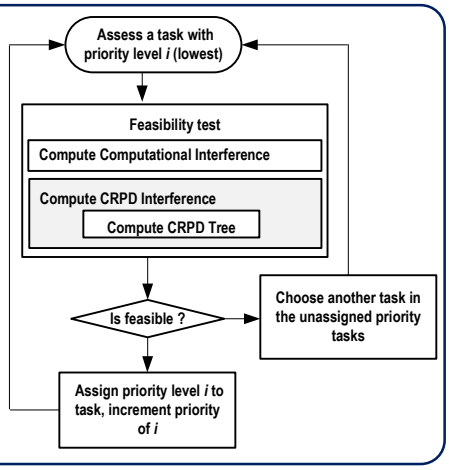
We assume an uniprocessor system running several periodic tasks with a preemptive fixed priority scheduler. If a system is not schedulable under a priority assignment policy because of the CRPD, there is no existing method reordering task priorities in order to decrease the impact of the CRPD and maintain a schedulable system.

III. OBJECTIVE

Assigning priorities to a set of tasks and guarantee that those tasks are schedulable while experimenting the CRPD.

IV. OUR APPROACH

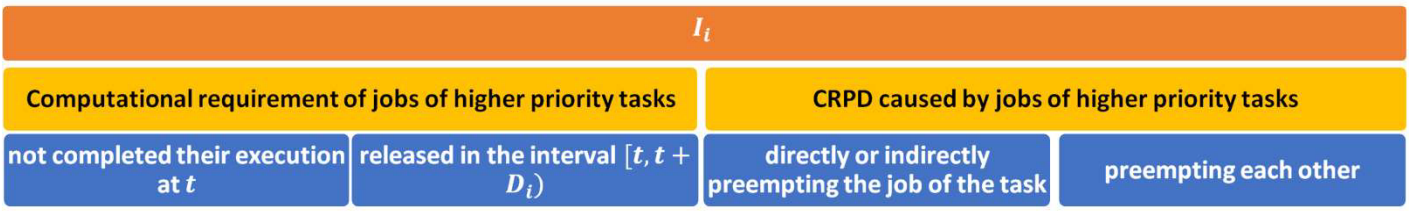
- Based on **Audsley's optimal priority assignment algorithm.** Assigning the **lowest priority level** to a task and verifying its feasibility while **priorities of other tasks are unknown**. If the task is feasible, assign this priority level to this task and move on to the next priority level.
- We propose a feasibility test taking into account the impact of CRPD by **integrating the computation of the CRPD into the original feasibility test of Audsley.**



IV.1 Interference Computation

A task is schedulable if all of its jobs released during the feasibility interval can meet their deadlines. Assuming a job of a task is released at time t, requires C_i units of computation time and must complete after D_i units of time, the job experiences the interference I_i caused by other jobs of higher priority tasks during the interval $[t, t + D_i)$. Then, the job of the task is feasible if:

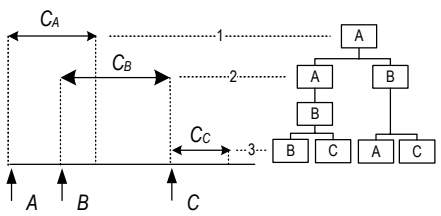
$$C_i + I_i < D_i$$



IV.2 CRPD Tree

Evaluating all possible preemptions and computing the upper-bound total CRPD. The tree ***T = (N;E)*** is defined by ***N***, the set of nodes and ***E***, the set of edges.

- Each node ***n*** of ***N*** models a running job at an instant.
- Each edge ***e*** of ***E*** models a decision of the scheduler: **allow preemption**, **deny preemption** and **select job.**



IV. EVALUATION

Space and time performances of the CRPD tree computation.

Tasks	*Jobs*	*Nodes*	*Computation Time (ms)*
5	5	98	4.7
6	6	307	6.1
7	7	1570	27.9
8	8	4793	193.2
9	9	21796	490.5
10	10	67365	1571.4

REFERENCES

- N. C. Audsley, "Optimal priority assignment and feasibility of static priority tasks with arbitrary start times," in Technical Report YCS 164, Dept. Computer Science. University of York, UK, 1991.
- C. Li, C. Ding, and K. Shen, "Quantifying the cost of context switch," in Proceedings of the 2007 workshop on Experimental computer science. ACM, 2007.

ACKNOWLEDGEMENT

The work is procedded in the context of the Cheddar project. Cheddar is supported by Ellidiss Technologies, Conseil régional de Bretagne, Conseil général du Finistère, BMO, EGIDE/Campus France PESSOA number 27380SA and Thales TCS.