Analyzing the Dynamics of Priorities of Weakly-Hard Real-Time Jobs under Overload Conditions

Background

Hard real-time tasks are those tasks whose execution must be strictly within a defined time limit, which is known as a deadline. However, some of the systems that are modeled as hard real-time can tolerate some degree of occasional deadline misses without critical consequences. This offers more scope for resource optimization and adaptability in environments where deadline rigidity is not critical.
It is in order to model this tolerance that the weakly-hard real-time model appears. In the Weakly-hard model, the tolerance to deadline misses is described by using the pair (m, K), in which "m" indicates the tolerable deadline misses in a windows of "K" executions.

Furthermore, a particular scheduling algorithm is required to schedule weakly-hard real-time tasks. Job-class-level (JCL) scheduling is single-core job-level fixed priority scheduling approach, in which the jobs receive different priorities upon meeting/missing their deadlines. When a job meets its deadline, the next job is assigned a lower priority from a predefined job-level priority classes. When a deadline miss occurs, the next job will get a higher priority predefined by a new priority class. Hence, the jobs of a given task change between different job-classes based on the interference of other tasks.

In JCL, we observed different behaviors of "job-classes transitioning" based on the utilization of the task set. Jobs remain in job-classes with higher priority for high utilization while the contrary happens for low utilization. We created a model named "stress-model" for describing the last mentioned behavior which models the state of the whole system based on the state of every task. This model aims to evaluate how much the system is loaded during runtime. In this master thesis, we want to evaluate the existing "stress-model".

Thesis Objective and Research questions:

How the stress-level in the model is affected in the following scenarios:

- low total utilization,
- high total utilization,
- same total utilization but different amount of tasks,
- a task execution takes longer than expected,
- the inter-arrival time of a task is shorter than expected

Your tasks:

- Implement algorithms in Rust
- Evaluate the existing model for answering the research questions
- Write a master thesis

Moreover, you may have a chance to visit DLR in Germany and present your findings.

Your qualifications:

Required

- Experience programing in C
- Knowledge of POSIX system calls
- You are able to communicate in English (supervision will be in English)

Optional

- Experience programing in Rust
- Real-time knowledge

Learning Objectives

- Mastery of weakly-hard real-time theory and models
- Implementation and evaluation of system-level models in Rust
- Insight into dynamic scheduling strategies like JCL
- Practical skills in performance modeling and stress analysis

Interested?

Please send us an email, write a bit about your motivation to work on the subject, your background, and at what stage of the master program you are. We will then set a meeting.

- Supervisor: Dr.-Ing. Kuan-Hsun Chen, Email: k.h.chen@utwente.nl
- External Supervisor: Dr.-Ing. Zain A. H. Hammadeh, Email: Zain.HajHammadeh@dlr.de