technische universität dortmund

computer science 12

Master Thesis

Suspension Absorption for Segmented Self-Suspending Tasks

Lars Willemsen
Dr.-Ing. Georg von der Brüggen

Otto-Hahn Str. 16
Technische Universität Dortmund
Email: lars.willemsen@tu-dortmund.de
September 15, 2026

In real-time scheduling theory the concept of a utilization bound is understood colloquially as *"the amount of work we can impose upon a system before deadline misses occur"*. More formally, it is commonly understood as the amount of accumulated period-relative execution demand a taskset can assume before it is no longer schedulable.

The most well-known utilization bound is discussed in the seminal work by Liu and Layland in 1973. In this work the authors provide a **sufficient least** upper bound, which takes the form of

$$U = \sum_{i=1}^{n} \frac{C_i}{T_i} \leq n\left(2^{\frac{1}{n}} - 1\right) \xrightarrow[n\to\infty]{} \ln 2 \approx 0.693 \quad (1)$$

This bound states that, if we assume a uniprocessor system that schedules only simple periodic tasks with implicit deadlines, we can *always* schedule the taskset if it stays under this utilization bound.

In general, under the same system assumptions, the **necessary** bound is much simpler: a uniprocessor can never offer more service to jobs than time available in the period of a job. Therefor it necessarily follows that for an deadline compliant schedule utilization stays below 1:

$$U = \sum_{i=1}^{n} \frac{C_i}{T_i} \leq 1 \quad (2)$$

However, there exists no such bound for the amount of *suspension* in a system. Suspension occurs when a job has to wait for some external event to complete. During a suspension other pending jobs may be scheduled. Once the external event is resolved the task may become ready again. Naturally we cannot establish a general bound on the amount of suspension in a schedulable task set in the same way we can for utilization; suspension does not consume any service directly.

In this thesis we assume the **segmented self-suspending** task set: task instances are expressed as execution segments and suspensions, in a perfectly interleaving fashion. Under the reasonable assumption that execution segments last at least one time unit we can derive bounds on the amount of suspension a task set can be allowed to permit. We will refer to such a bound as the *absorption bound* A.

A conjectured necessary bound is depicted below:

$$A \leq \frac{1}{H}\left\lfloor \frac{(H-1)^2}{4} \right\rfloor. \quad (3)$$

The intuition behind this bound is derived from the following illustration, in which all tasks have the same period and exactly one suspension interval connected by two execution segments.

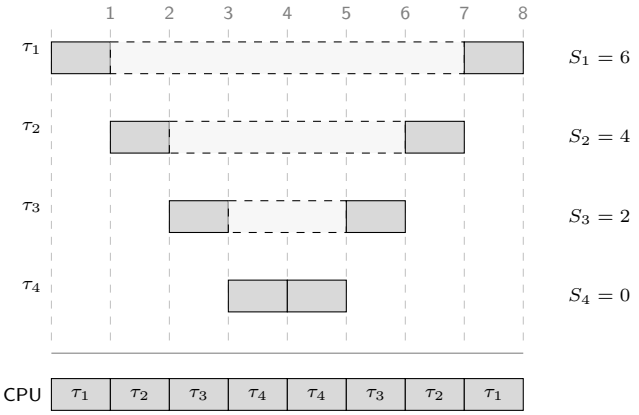

Figure 1: Four tasks sharing the period $T_i = H = 8$, each with at most one suspension interval (dashed) enclosed by two execution segments of one time unit.

This thesis revolves around investigating schedulability under absorption. What direction it takes depends on your preference. Lofty goals include a *sufficient* bound similar to the Liu and Layland bound, for the Segmented Self-Suspending task model scheduled by a Rate monotonic scheduler, as well potential tightening of this bound under certain assumptions on the tasks (harmonic or semi-harmonic periods).

It would furthermore be interesting to investigate this *experimentally*: when do schedules start to miss their deadlines? What would be a good *estimator* for when most tasksets start to break down?

References:

[1] C. L. Liu and J. W. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *Journal of the ACM*, 20(1):46–61, 1973.