jian-jia.chen [☺] tu-dortmund.de
georg.von-der-brueggen [☺] tu-dortmund.de
wen-hung.huang [☺] tu-dortmund.de
jan.kleinsorge [☺] tu-dortmund.de

Exercises for Lecture
Real-Time Systems and Applications
Summer Semster 15

Exercise Sheet 6

(11 Punkte)

Exercise Due at Wednesday, June 17, 2015, 12:00 Uhr

Hinweise: Gruppenarbeit von bis zu drei Personen aus der gleichen Übungsgruppe ist möglich. Bitte vergessen Sie nicht Ihre Namen und Ihre Matrikelnummern auf die Lösung zu schreiben. **Die Abgaben können in den beschrifteten Briefkasten vor dem Sekretariat des LS12 (OH16/E22) eingeworfen oder per Mail (PDF Format) an georg.von-der-brueggen [☺] tu-dortmund.de abgegeben werden.**

Note: It is allowed to work in a group of up to three persons, if these persons are from the same practice group. Please do not forget to write your name and your Matrikelnummer on the solutions. **The solutions can either be placed in the mailbox in front of the secretary's office of LS 12 (OH/E22) or sent by mail (PDF format) to georg.von-der-brueggen [☺] tu-dortmund.de**

Exercise Sessions:

Do, 10:15 - 11:45	OH16/E18
Do, 14:15 - 15:45	OH16/E18

6.1 Self-Suspension (2 Punkte)

Prove that the speedup factor of the *Proportional Fixed-Relative Deadline Assignment* for 1-segmented suspension sporadic task systems is at least $\Omega(n)$. The assignment is

- $D_{i,1} = \frac{C_{i,1}}{C_{i,1}+C_{i,2}}(T_i - S_i)$
- $D_{i,2} = \frac{C_{i,2}}{C_{i,1}+C_{i,2}}(T_i - S_i)$

Hint: If Equal Deadline Assignment (EDA) is feasible, then the optimal schedule is also feasible.

6.2 Necessary condition (3 Punkte)

Consider task systems with constrained deadlines.

- Write down the necessary condition for arbitrary self-suspension sporadic task systems under uniprocessor fixed-priority scheduling.
- Write down the sufficient condition for arbitrary self-suspension sporadic task systems under uniprocessor fixed-priority scheduling.

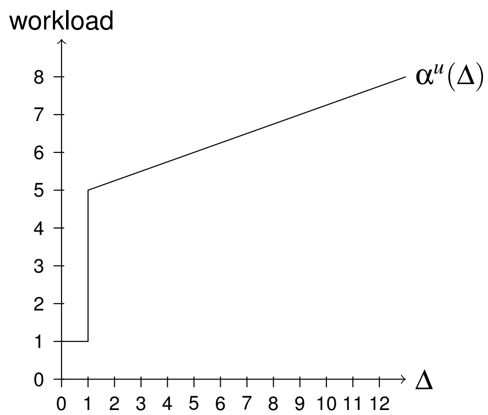
Explain/prove your answer.

6.3 RTC (4 Punkte)

1. Mr. Smart wants to define the feasibility of his design as follows: "the system is feasible if $\alpha^u(\Delta) \leq \beta^l(\Delta)$ for any $\Delta \geq 0$", where α and β are arrival curve and service curve, respectively. Is this a reasonable definition?
2. How to derive the arrival curves from a set of given traces of events? How to derive the service curves from a set of traces of services?
3. What are the limitations of using Greedy Processing Components (GPC) in Real-Time Calculus (RTC)? How to perform schedulability analysis for fixed-priority scheduling based on GPCs? For fixed-priority scheduling, what are the similarity and difference of worst-case response time analysis in RTC to and from the time-demand schedulability tests for sporadic real-time tasks with arbitrary deadlines?

Hint: 1+1+2

6.4 Real-Time Calculus (1 Punkt)



Is the arrival curve in the left correct? Is it tight? If yes, why? If not, how to make it tighter?

6.5 Challenge on Arbitrary Schedules with 1-Segmented Self-Suspension (optional) (1 Punkt)

Prove the following necessary condition for any implicit-deadline arbitrary 1-segmented self-suspension sporadic task model:

$$\forall t > 0, \qquad \sum_{\tau_i \in \mathbf{T}} dbf_i^*(t) \leq t. \tag{1}$$

where

$$dbf_i^*(t) = \begin{cases} 0 & 0 \leq t < T_i - S_i \\ C_{i,\max} + \\ \left\lfloor \frac{t-(T_i-S_i)}{T_i} \right\rfloor (C_{i,1} + C_{i,2}) & t \geq T_i - S_i \end{cases} \tag{2}$$

General Hints: