Real-Time and Embedded Systems

<u>Problem 2</u>

Adrián Gómez Llorente

0904327G

Introduction

In this document second problem questions of Real Time and Embedded Systems are answered and explained.

Question 1

An appropiate frame size is based in the next 3 constraints:

Eq 1: To avoid preemption, wants jobs to start and complete execution within a single frame:

$$f > max(e_1, e_2, \ldots, e_n)$$

Eq 2: To minimize the number of entries in the cyclic schedule, the hyper-period should be an integer multiple of the frame size:

$$\exists i : mod(p_i, f) = 0$$

Eq 3: To allow scheduler to check that jobs complete by their deadline, should be at least one frame boundary between release time of a job and it s deadline:

$$2 * f - gcd(p_i, f) \leq D_i \, for \, i = 1,2 \ldots n$$

So we are going to look for a frame size that satisfy 3 constraints before

Eq 1:

$$f > max(e_1, e_2, \ldots, e_n) = max(1,3,4,6) = 6 \Rightarrow f \geq 6$$

Eq 2:

We have to calculate the Hyper period:

Hyper period = $lcm(p_1, p_2, p_3, p_4) = 1320$

So values that satisfy the constraint are shown next:

$$f \in \{8,10,11,15,20,22\}$$

Eq 3:

Right now we have to check if previous values satisfy the third constraint. After trying with all of them we get that it's only valid for 8:

$$T_1 : 2 * 8 - gcd(8,8) = 16 - 8 = 8 \leq 8$$

So the correct frame size obtained is 8

Question 2

To check if a system is schedulable using Rate Monotonic algorithm we have to check next theorem:

A system of n independent preemptable periodic tasks with $D_i = p_i$ *can be feasibly scheduled on one processor using RM if*

$$U \leq n*(2^{1/n}-1)$$

But this condition is sufficient but not necessary.

System 1

We are going to calculate the total utilization of the system:

$$U = u_1 + u_2 + u_3 = \frac{1}{5} + \frac{1}{3} + \frac{3}{15} = 0.733$$

In this system we have 3 tasks so we have to calculate U_{RM} with the parameter n=3:

$$U_{RM}(3) = 3(2^{\frac{1}{3}}-1) = 0.779$$

Using previous theorem we get that **a feasible monotonic schedule is guaranteed** as can be seen below:

$$U \leq U_{RM} \Rightarrow 0.733 \leq 0.779$$

System 2

Now we are going to check previous algorithm with the system 2. First step is calculating the total utilization of the system:

$$U = u_1 + u_2 + u_3 + u_4 = \frac{2}{5} + \frac{1}{4} + \frac{1}{10} + \frac{3}{20} = 0.90$$

We have 4 tasks so we are going to calculate U_{RM} with the parameter n=4

$$U_{RM}(4) = 4(2^{\frac{1}{4}}-1) = 0.756$$

With the results obtained we can assume that there is a feasible monotonic schedule but we will have to simulate the system execution to be sure.

Question 3

First of all task must be ordered trough their period so tasks with shorter period will have the higher priority. Tasks will be ordered as shown next:

$$T_1 > T_2 > T_3$$

We have all tasks ordered so right now we have to fill a table with all the execution times like next one:

Time	Ready to run	Running	Time	Ready to run	Running
0	$J_{2,1}$ $J_{3,1}$	$J_{1,1}$	6	$J_{3,1}$ $J_{3,2}$	$J_{1,3}$
1	$J_{3,1}$	$J_{2,1}$	7	$J_{3,2}$	$J_{3,1}$
2	$J_{3,1}$	$J_{2,1}$	8	$J_{3,1}$	$J_{2,3}$
3	$J_{3,1}$	$J_{1,2}$	9	$J_{2,3}$ $J_{3,2}$	$J_{1,4}$
4	$J_{3,1}$	$J_{2,2}$	10	$J_{3,2}$	$J_{2,3}$
5	$J_{3,1}$	$J_{2,2}$	11		$J_{3,2}$

As we can see in table above during the 6th step we can see that 2 executions of task 3 are ready to run so this tasks **won't be schedulable**.