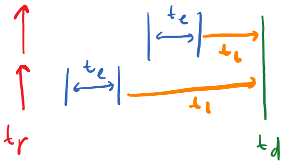
Least Laxicity First

2 Ekim 2017 Pazartesi 11:41

Dynamic. Check laxicity rather than deadline



$t_e < t_d - t_r$

$t_l = t_d - t_s - t_e$

If laxity is negative, deadline is already missed

Laxicity time: Time left untill deadline from end of execution

Task with shortest laxicity time runs first

Aperiodic task: To serve A.S.A.P.

You have periodic tasks happens regulary with any algorithm.

And aperiodic tasks comes to play :)

Approaches

X Highest prior to aperiodic tasks.

Not a good solution => periodic tasks might miss deadline

✓ Utilisation of IDLE times or:

Not Best but good

Lowest prior (Background)

Not A.S.A.P but a solution considering tight schelude

we have unutilized time

✓ Poller: Ta(P,e)

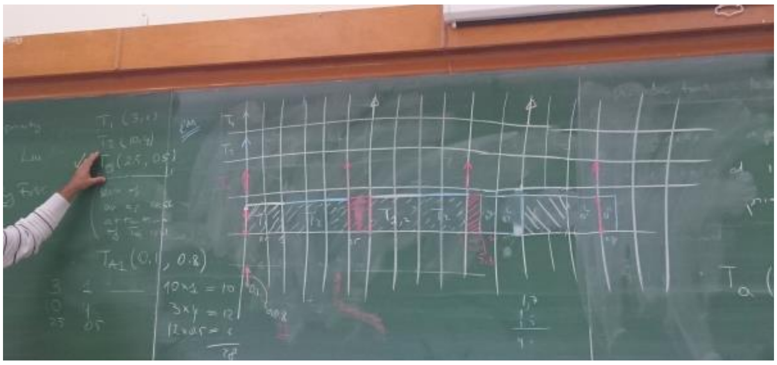
Think a process (as a black box) that runs aperiodic tasks (in the box)

Assign processing time in hyperperiod.

Not sure of aperiodic task exists at the time of Ta instance

If there is no aperiodic task avalible at the time of launch Ta will left CPU immideatly (ignored).

If algorithm is staeic, aperiodic task that starts after Ta start time will not be able to run.



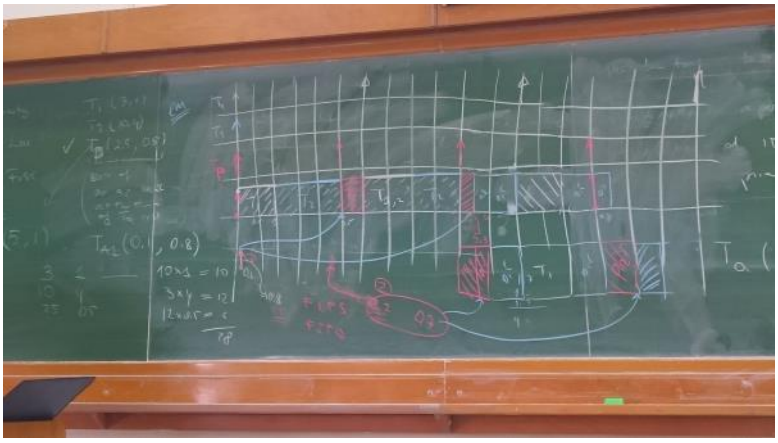
First response time would be shorter if period is smaller even though CPU time is same

$\frac{0{,}25}{1{,}25}$, $\frac{0{,}5}{2{,}5}$, $\frac{1}{5}$

↳ better (more frequent check)

↳ better (more frequent check)

Ta serves aperiodic tasks as FCFS (Queue)



✓ Servers

↳ Consumption Rule

↳ Replenehsment rule (To fill up to its max)

Deferable Server (Pg: 197)

CR: Consume one unit of resoursces per unit of time

RR: For every period top the budget up

introduced with servers

Do it for ELF

Also check

Simple Sporadic Server 207,218

Sp SL server 212,213