COLLEGE OF MANAGEMENT & INFORMATION TECHNOLOGY

BACHELOR IN INFORMATION TECHNOLOGY

Real Time System
BIT

Submitted by:
Name: Amir Shrestha
Year/ Semester: III Year /V Semester
LCID: LC00017000502

Submitted by:
Saroj Ghimire

Date: 5/28/2021

QUESTION No 1:

State whether the following statements are TRUE or FALSE. Justify your answer.

Answer:

a. A hard real time application consists of only hard real-time tasks.

   FALSE, in hard real time application, it may also consist of other several non-real time tasks such as logging activities.

b. Every safety critical real time system contains a fail-safe state.

   FALSE, because the failure of safety critical system can lead the lives of people. Example: navigation system on board an aircraft.

c. A deadline constraint between two stimuli is a behavioral constraint on the environment of the system.

   TRUE, deadline constraint is behavioral constraint where it says that since the constraint is imposed on the basis of 2nd stimulus event.

d. A good algorithm for scheduling of hard real time tasks tries to

   FALSE, because good scheduling algorithm for hard real time tasks is only concerned with completing the tasks before the deadlines.

e. All hard real time systems usually are safety critical in nature.

   FALSE, because in real time system it says that not all hard real time system is safety critical in nature. (e.g.: computer games)

f. Soft real time tasks do not have any associated time bounds.

   FALSE, because soft real time task also has time bounds associated with them.

QUESTION No 2:

What is the difference between a performance constraint and a behavioral constraint in a real-time system?

Answer:

The difference between a performance constraint and a behavioral constraint in a real - time system are:

S.N.	Performance Constraint	S.N.	Behavioral Constraint
1.	Constraints that are imposed on the response of the system.	**1.**	Constraints that are imposed on the stimuli generated by the system.
2.	It basically describes the overall performance of the system.	**2.**	It basically describes the behavior of the environment
3.	It ensures that the computer system performs satisfactorily.	**3.**	It ensures that the environment of a system is well-behaved.

QUESTION No 3:

What is understood by jitter associated with a periodic task? Mention techniques by which jitter can be overcome.

Answer:

Jitter is the variation between the expected timing and the actual timing for a task. It the derivation of task given to a period and from its strict periodic behavior.

Techniques by which jitter can be overcome are mentioned below:

- If only one or two actions (tasks) have tight jitter requirements, these actions are assigned very high priority. (Only work if fewer tasks)
- When it is used in an application in which the tasks are barely schedulable, which may lead to miss delaines for some tasks.
- If jitter must be minimized for an application that is barely schedulable, each task needs to be split into two: one which computes the output but does not pass it on, and one which passes the output on.
- This method involves setting the second tasks priority to very high values and its period to be the same as that of the first task.
- An action scheduled with this approach will run one cycle behind schedule, but the tasks will have tight completion time jitter.

(Reference: benchpartner.com)

QUESTION No 4:

Explain real time application with its working mechanism and use if any block diagram. (It must include one each of soft and hard real time system)

Answer:

Hard RTS: Miss Deadline ==> Failed/ No Performance/ Utilization = 0

- Entire System Process/Task Fails,
- No any kind of performance at all,
- No or zero utilization.

Example:

If given/defined time is 5 second and if task gets completed at or before 5 second then only task is considered successful. And utilization of the running task greater than zero. Or else if task gets completed after 5 second/ miss given deadline then task is failed. And utilization of the running task exactly equals to zero.

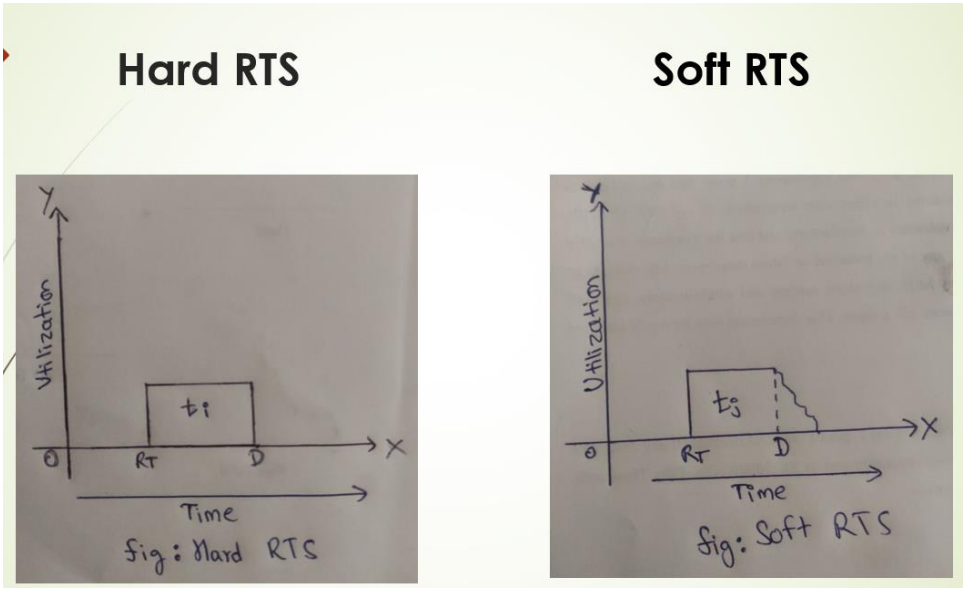
Soft RTS: Miss Deadline ==> Not Completely Failed/ Decrease Performance/ Util> 0

- System Process/Task is not completely failing.
- Decrease performance.
- Decrease utilization but greater than zero after deadline but will be zero after some additional time.

Example:

If given/defined time is 5 second and if task gets completed at or before 5 second then only task is considered successful. And utilization of the running task greater than zero. Also, if task gets completed after 5 second/ miss given deadline then task is not completely failing. And utilization of the running task decrease task greater than zero.

Graph:



Graph:

- Time at x-axis
- Utilization at y-axis
- Task as Bar
- Release time & Deadline

See after deadline utilization/performance is zero in Hard RTS.

See after deadline utilization is greater than zero but

Performance/ utilization decreases in Hard RTS.

QUESTION No 5:

Give two different explanations of why the periodic tasks (2,1), (4,1) and (8,2) are schedulable by the rate monotonic algorithm

Answer:

The rate monotonic algorithm is a procedure for assigning fixed priorities to tasks to maximize their "stimulability." A task set is considered schedulable if all tasks meet all deadlines all the time.

$$\sum_{k=1}^{n} \frac{C_k}{T_k} \leq U_{RM} = n(2^{1/n} - 1)$$

The term U is said to be processor utilization factor.

N is the number of tasks.

In our cases: 1/2 + 1/4 + 2/8 = 1 which is not less than 0.78.

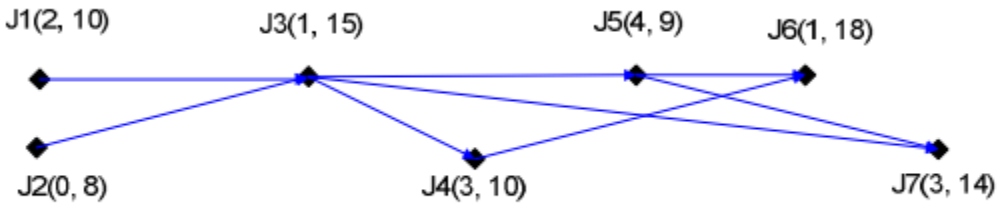
Therefore, 1 <= 0.78. Here, 1 is not less than 0.78 so the above task set is not schedulable under Liu and Layland's condition.

The above condition is not necessary, we can do a somewhat more involved sufficient and necessary condition test, as follows. We have to guarantee all the task that can be scheduled in possible instances. In particular if a task can be scheduled in its critical instances, then the stimulability guarantee condition holds.

(Reference: Dhanraj, 2021)

QUESTION No 6:

Consider a set of jobs J (r, d) with the release time and the deadline. The precedence constraints are shown as in the graph below. Derive the formula of calculating the effective release time and deadline. Calculate the effective release time and deadline for each job in the set.



Answer:

Effective Release time:

The effective release time of job without predecessors is equal to its given release time. The effective release time of job with predecessors is equal to the maximum value among its given release time and the effective release time of all of its predecessors. □

Effective Deadline:

The effective deadline of a job without a successor is equal to the given deadlines. The effective release time of all the jobs can be computed in one pass.

Calculation of effective release time:

- As J1 and J2 have no predecessors, their effective release times are their given
- release times i.e., 2 and 0 respectively.
- The given release time of J3 is 1, but the latest effective release time of its
- predecessors are 2 i.e., of J1. Hence, the effective release time of J3 is 2.
- Similarly, the effective release times of J4, J5, J6 and J7 are 3, 4, 4 and 4 respectively.

Calculation of effective deadline:

- As J6 and J7 have no successors, their effective deadlines are their given
- deadlines i.e., 18 and 14 respectively.
- Since the effective deadlines of the successor of J4 and J5 are larger than their
- given deadlines, so the effective deadlines of J4 and J5 are equal to their given
- deadlines i.e., 10 and 9 respectively.
- The given deadline of J3 i.e., 15 is larger than the minimum value of its
- successors i.e., 9, so the effective deadline of J3 is 9.
- Similarly, the effective deadline of J1 and J2 are 9 and 8 respectively.

Thank You!