Scheduling Algorithm and Analysis

Rate Monotonic
(Module 29)

Yann-Hang Lee
Arizona State University
yhlee@asu.edu
(480) 727-7507

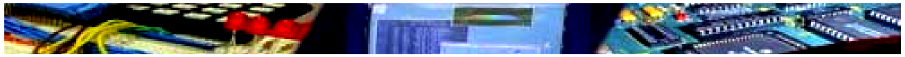
Summer 2014

Rate-Monotonic Scheduling Algorithm

- **A base case: no additional overhead, simple periodic tasks with $p_i = D_i$**
- **Assign priorities according their periods**
  - T_i has a higher priority that T_k if $i < k$ ($p_i < p_k$)
  - Is RM optimal? $\Rightarrow$ if there is a feasible fixed-priority schedule, then RM is feasible
  - How do we know RM is feasible $\Rightarrow$ schedulability test
- **Results:**
  - RM is optimal if $p_i \geq D_i$
  - sufficient condition $\Rightarrow$ utilization test

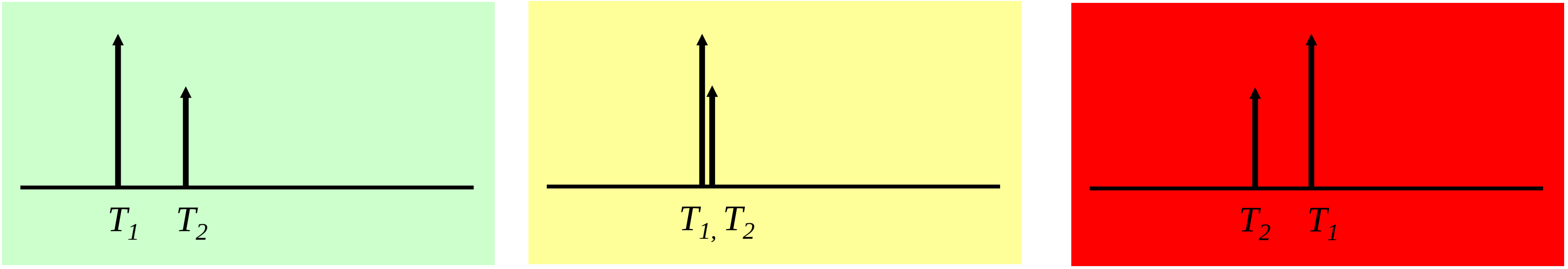
$$U = \sum_{i=1}^{n} \frac{e_i}{p_i} \leq n(2^{1/n} - 1)$$

  - a complete test $\Rightarrow$ what is the worst response time given all possible arrivals and preemptions



Critical Instant

- **Critical instant of T_i: a job of T_i arriving at the instant has a maximum response time**
- **If we can find the critical instant of T_i, then**
  - check whether all jobs of T_i meet their deadlines
  - let's increase ei until the maximum response time = D_i
  - $\Rightarrow$ schedulable utilization
- **In-phase instant is critical: all higher priority tasks are released at the same instant (assume all jobs are completed before the next job in the tasme task is released.)**
  - which T_2 has the maximum response time



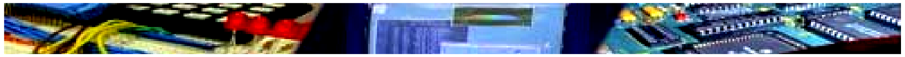
Schedulability: UB Test

- **Utilization bound (UB) test: a set of *n* independent periodic tasks scheduled by the rate monotonic algorithm will always meet its deadlines, for all task phasings, if**

$$\frac{C_1}{T_1} + \ldots\ldots + \frac{C_n}{T_n} <= U(n) = n(2^{1/n} - 1)$$

$U(1) = 1.0$	$U(4) = 0.756$	$U(7) = 0.728$
$U(2) = 0.828$	$U(5) = 0.743$	$U(8) = 0.724$
$U(3) = 0.779$	$U(6) = 0.734$	$U(9) = 0.720$

- **For harmonic task sets, the utilization bound is *U(n)=1.00* for all *n*.**



Schedulability Test: Time-Demand Analysis

- **Consider in-phase instant only**
- **If J_i is done at t, then the total work must be done in [0,t] is (from J_i and all higher priority tasks)**

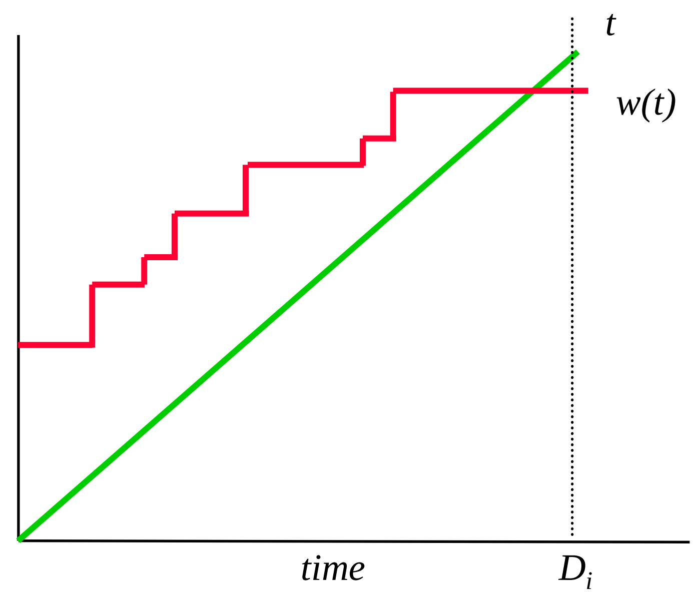
$$w_i(t) = e_i + \sum_{k=1}^{i-1} \left\lceil \frac{t}{p_k} \right\rceil e_k$$

- **Can we find a $t \le D_i$ such that**

  $$w_i(t) \le t$$

  - cannot check all $t \in [0, D_i]$
  - check all arrival instants and D_i
- **The completion time of J_i satisfies**

$$t = e_i + \sum_{k=1}^{i-1} \left\lceil \frac{t}{p_k} \right\rceil e_k$$



Supplementary Slides