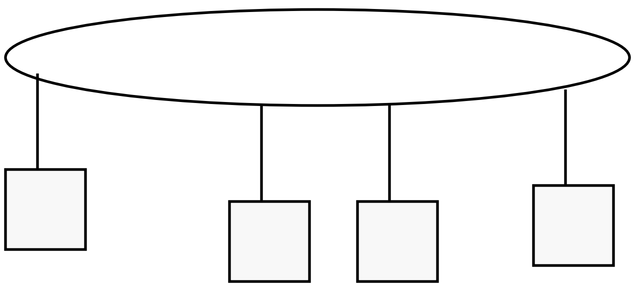
Simple Network Scheduling

- In distributed systems, signals are captured at one place, transmitted over a network and then utilized by the destination node, for example, video data captured at one place, sent over the network and displayed at another node.
- The simplest form of a deterministic network is a token ring such as FDDI, a 100 M Bit/sec LAN.
- Under FDDI, the messages queued at each node can be queued in either FIFO or priority order. We should use priority queueing and assign messages with GRMS priorities.



Network Scheduling: FDDI

- FDDI can be configured into synchronous mode for real time applications. We shall assume that all messages are periodic. (If not, we will use sporadic servers, so that the resulting aperiodic data streams can be analyzed as if they are periodic.)

  - The source node is responsible to remove its own packets when they come around the ring.
  - U_i is the percentage of ring bandwidth that will be used by a node to transmit data on the ring. $U_i = U_{i1} + U_{i2} + \ldots + U_{in}$, U_{ik}

  - For example: at node 2, there are two periodic messages
  - Message 1 with transmission time 1 and period 10
  - Message 2 with transmission time 2 and period 40
  - $U_2 = U_{21} + U_{22} = 0.1 + 0.05 = 0.15$

FDDI (cont.)

- Walk time, W, the time a token circulates the ring once when there is no transmission.

- Node i's hold time, H_i, is maximum time that Node i can hold the token and transmit data.

- <u>In synchronous mode</u>, the target token rotation time, TTRT, is the sum of walk time and hold times around the ring.

- TTRT is a design parameter and should be no longer than the shortest period of all the periodic messages.

- In token ring, hold times are used for transmission. Walk time represents the sum of each node's waiting time for the token. Walk time is overhead.

Proportional Allocation of Bandwidth

- How can we determine the hold time for each node?

- Proportional allocation is a simple algorithm that makes hold time at each node proportion to its bandwidth requirements:

  - Initial holding time: H_i = (TTRT - W)* (U_i /U), where Ui is the total utilization of node i and U is the total bandwidth of the ring
  - If a particular node cannot schedule all its messages while others can, adjustments to Hi’s can be made.

Example of Proportional Allocation

- Suppose that there are two stations S_1 and S_2.
- Let S_1 have 1 message stream (C_{11} = 5 and T_{11} = 100) and let station S_2 have 2 message streams (C_{21} = 20, T_{21} = 200) and (C_{22} = 30, T_{22} = 300) , where the C's and T's are transmission time and periods respectively.
- Let the walk time be 1 msec.
- U_1 = 5/100 = 0.05 and U_2 = 20/200 + 30/300 = 0.2.
- Let walk time (propagation delay) be 1.
- Let TTRT be the shortest message period 100.
- H_1 = (100 - 1)* 0.05/(0.05 + 0.2) = 19.8
- H_2 = (100 -1)*0.2/0.25 = 79.2.

Scheduling the Ring

- In CPU scheduling, we have periodic (or sporadic) tasks sharing the CPU. Tasks are assigned rate monotonic priority. Ready tasks with higher priority get the CPU first.
- We can map the periodic tasks to periodic messages. Both are given rate monotonic priority.
- At each node, when the node holds the token, high priority messages go first.
- CPU will always be given to the highest priority ready task. But across the nodes, the token goes in a round-robin way. The node with the highest priority message still have to wait until the token arrives.
- The TOKEN DOES <u>NOT</u> UNDERSTANDING PRIORITY

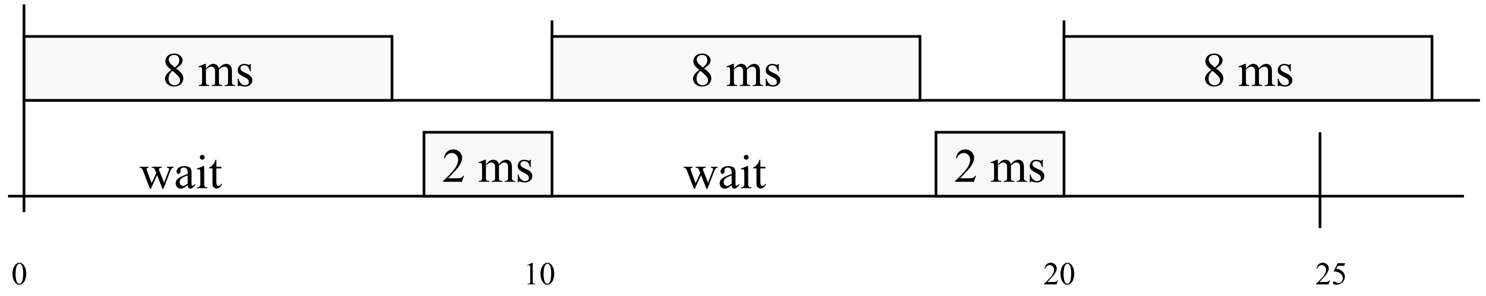
Are GRMS Analysis Doomed?

- GRMS is a priority scheduling theory, but token does not recognized priority.
- How can we approach this problem at all?
- Shall we give up?

Think Creatively

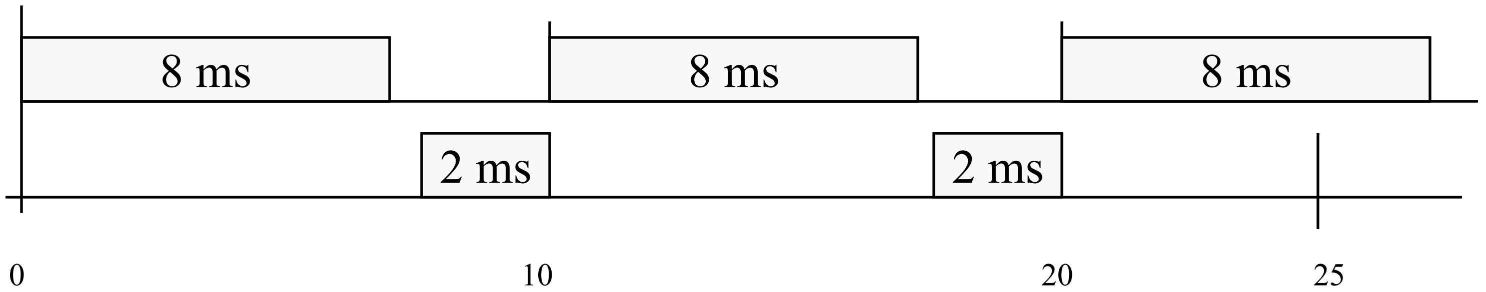
- Let's look at the actual transmission pattern. For example:
  - Suppose TTRT (token target rotation time) is 10 ms and a node has
    - hold time 2 ms
    - a message with transmission time 4 ms and period 25 ms

  - The worst case is that the node waits for 8 ms.
  - It can send 2 ms.
  - Have you seen a pattern like this before?



What is the Equivalent Task Set

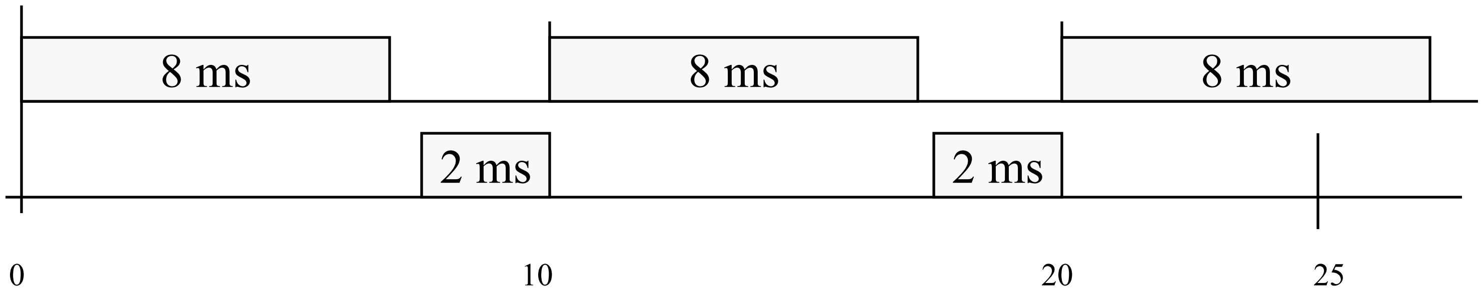
•



Class Exercise: Look at this picture. Create a CPU task set that will give the identical timeline. (10 min)

Equivalent Task Set

•



Task 1: C1 = 8 ms, T1 = 10

Task 2: C2 = 4 ms, T2 = 25

At each node, the "Task 1" is known as the TTRT Task, its execution time is (TTRT – HOLD TIME of THIS NODE) and the period is just TTRT.

Think about this: from the viewpoint of THIS node, every 10 ms, the rings is taken away for (10 – 2) = 8 ms, as if there were a high priority task with period 10 ms and executive time 8 ms.

Sample Problem: FDDI

- Node 1 has two periodic message, {(c1 = 20, T1 = 200) and (c2 = 30, T2 = 300)}
- The TTRT is 100 and the hold for Node 1 is 80

- What is the equivalent task set for Node 1

- Answer: {(20, 100), (20, 200) and {30, 300)}.