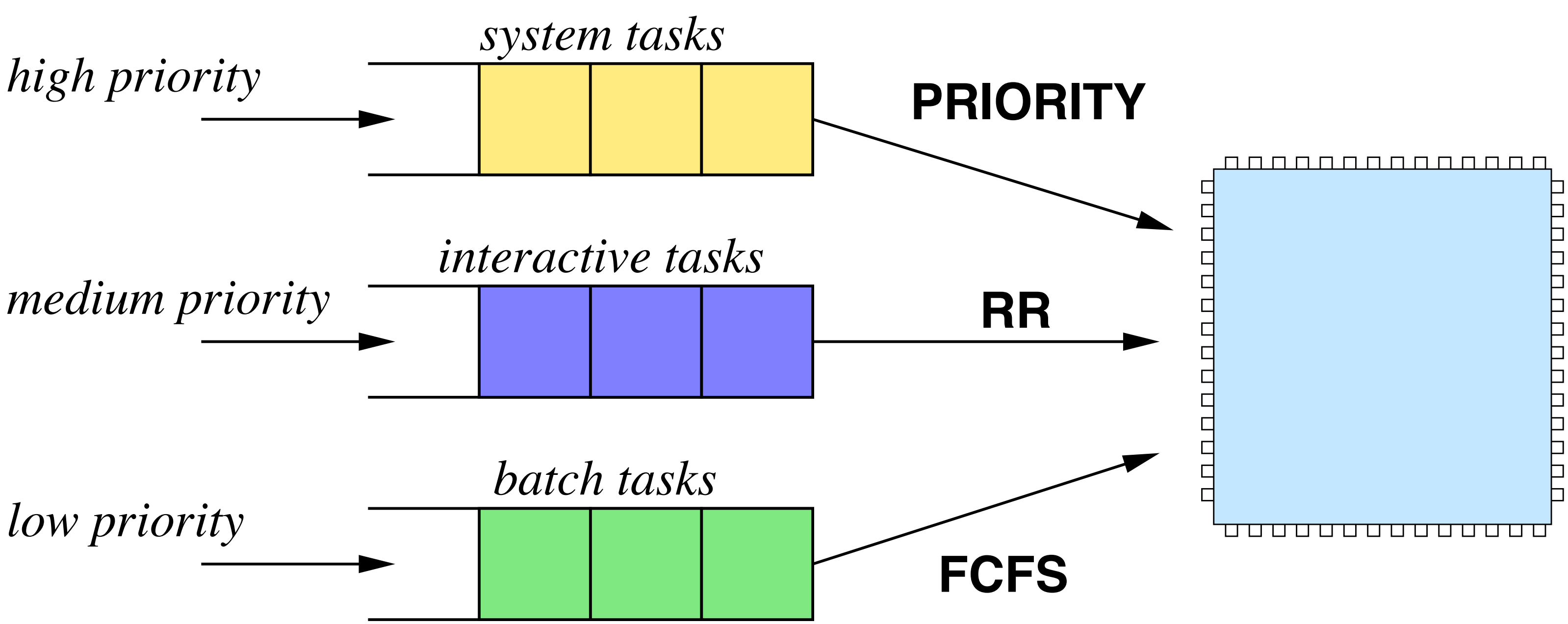
ECE 3140 / CS 3420
EMBEDDED SYSTEMS
LECTURE 16

Prof. José F. Martínez

TR 1:25-2:40pm in 150 Olin

Multi-Level Scheduling

Mix and match different scheduling algorithms

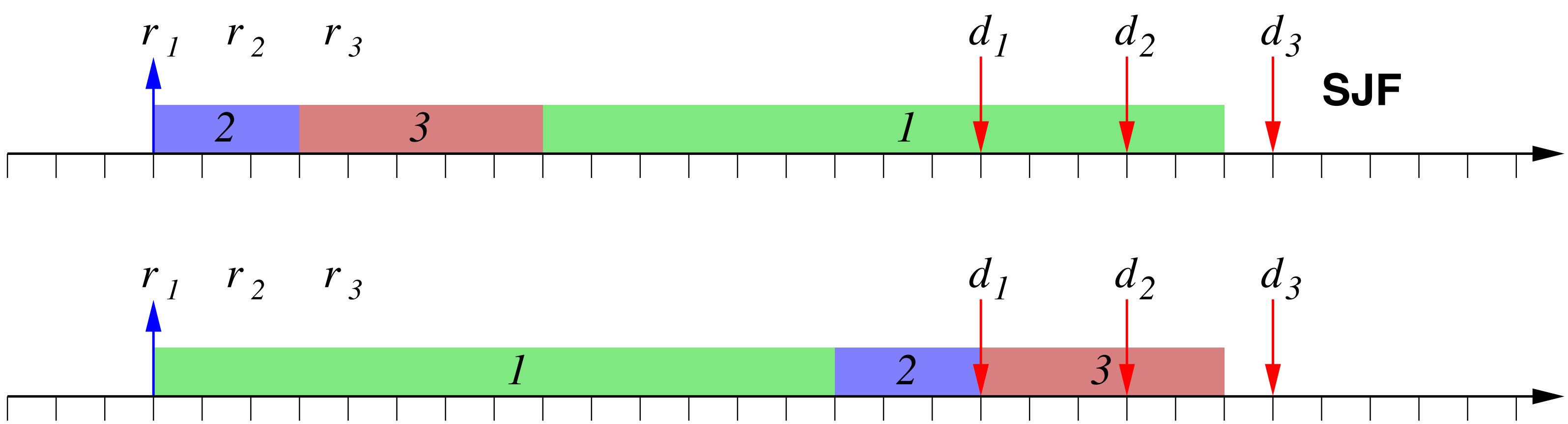


Real-Time Algorithms

Tasks can be scheduled by:

- relative deadlines D_i—static
- absolute deadlines d_i—dynamic

Since the goal is to meet deadlines, we should be using knowledge of deadlines to determine the schedule.



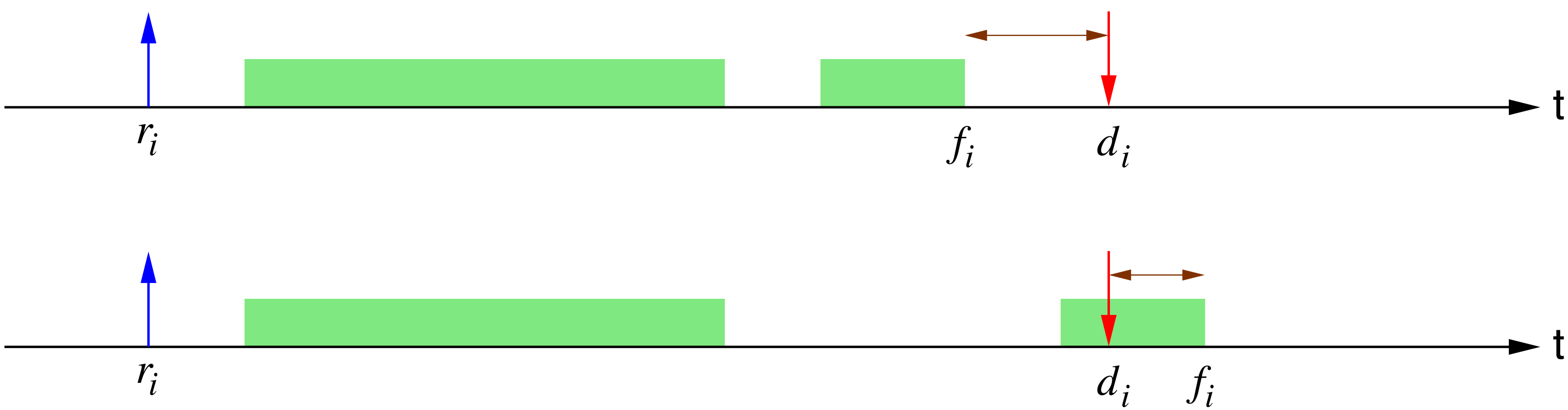
Earliest Due Date

Earliest Due Date (EDD): select the task with the earliest *relative* deadline.

- All tasks arrive simultaneously
- Fixed priority (D_i is known)
- Preemption is not an issue
- *It minimizes the maximum lateness* L_{max}

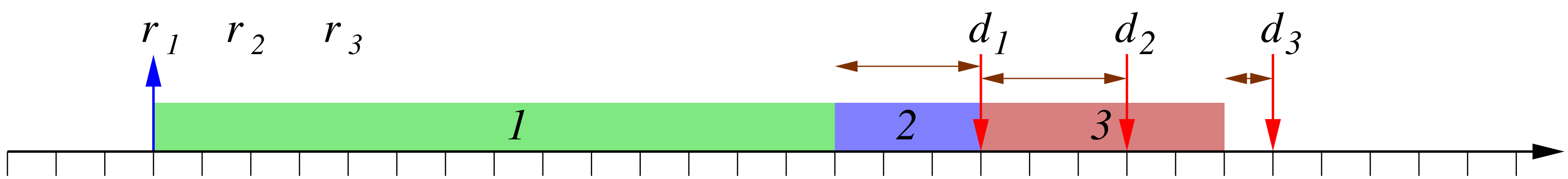
Earliest Due Date

$$L_i = f_i - d_i$$



Maximum Lateness

$$L_{max} = \max_i (L_i)$$



- $L_{max} < 0 \Rightarrow$ no task misses its deadline

Earliest Due Date

Jackson's Rule: *Given a set of n independent tasks, any algorithm that executes the tasks in order of nondecreasing deadlines is optimal with respect to maximum latenesss.*

Why does it minimize the maximum lateness? Proof?

- In particular, if $L_{max}(\sigma)$ is the maximum lateness of a schedule, then:

$$\forall \sigma : \qquad L_{max}(\sigma_{EDD}) \leq L_{max}(\sigma)$$

Earliest Due Date

How can we guarantee that a task set Γ is feasible?

- We can compute L_{max}!

A task set is feasible iff $\forall i : f_i \leq d_i$

If we sort the tasks using EDD and all tasks arrive simultaneously, then

$$f_i = \sum_{k=1}^{i} C_i$$

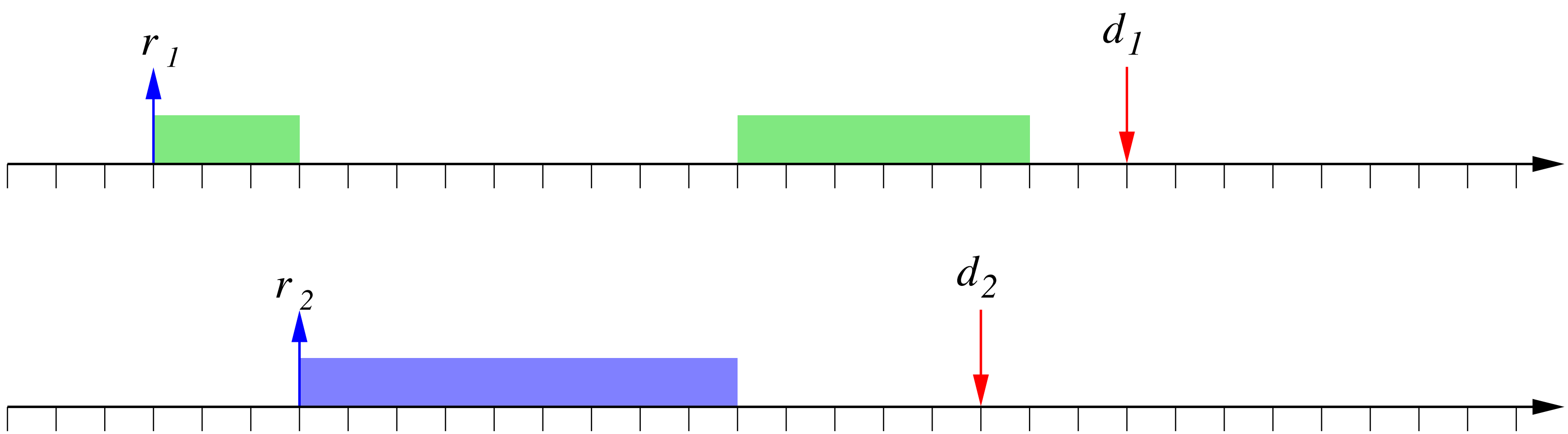
Earliest Deadline First

Earliest Deadline First (EDF): select the task with the earliest *absolute* deadline.

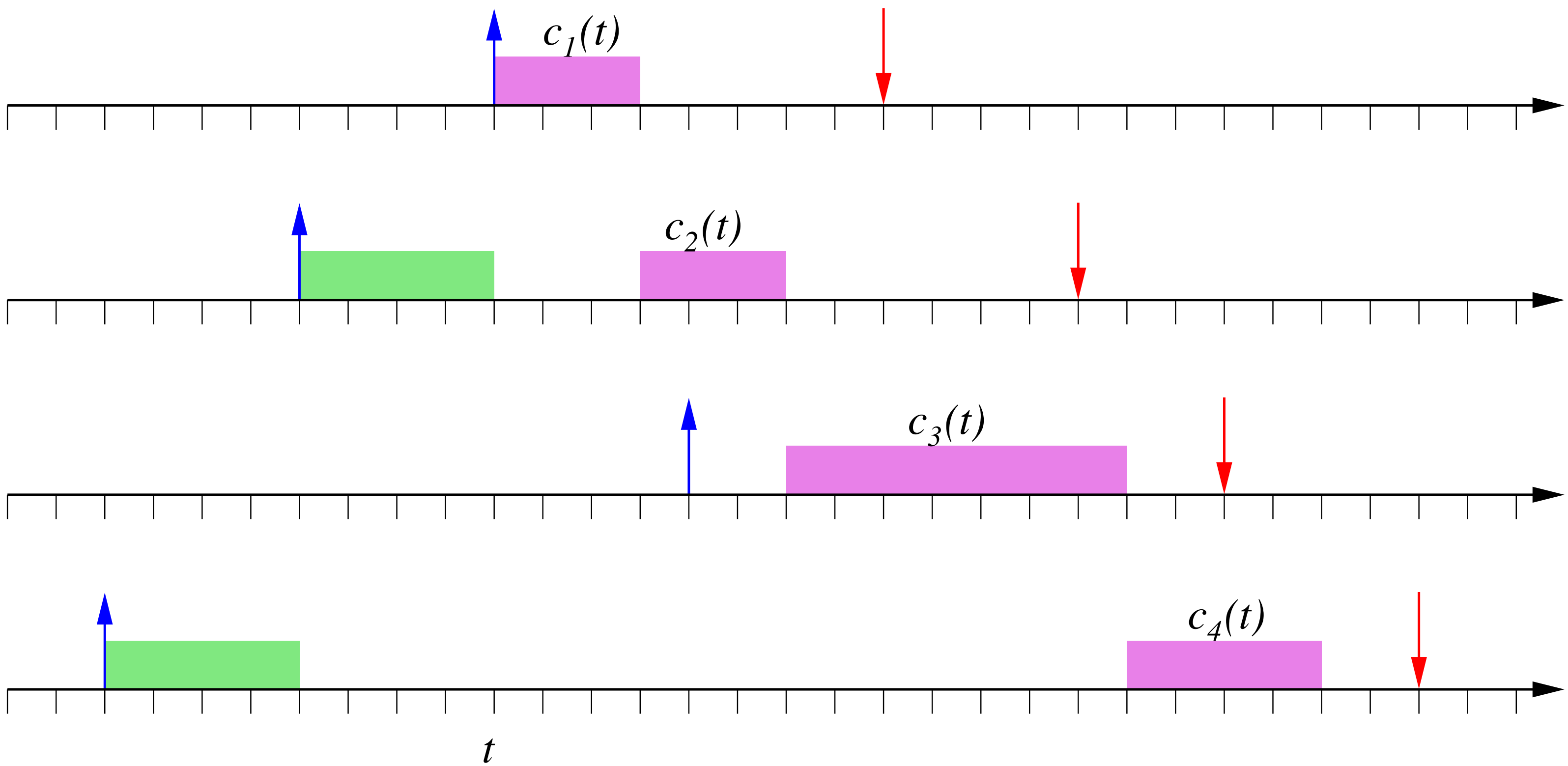
- Tasks may arrive at any time
- Dynamic priority (d_i depends on when the tasks arrive)
- Preemptive tasks
- *It minimizes the maximum lateness* L_{max}

Earliest Deadline First

Tasks that arrive with earlier deadlines pre-empt tasks with later deadlines.



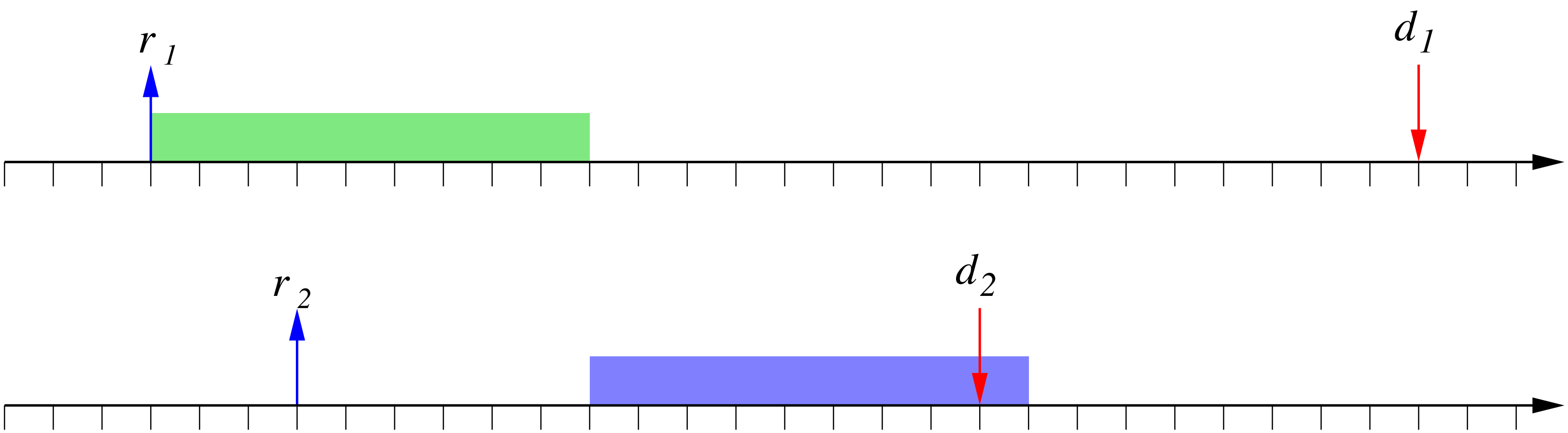
Earliest Deadline First: Feasibility



$$\forall i : \sum_{k=1}^{i} c_k(t) \leq d_i - t$$

Earliest Deadline First

Under non-preemptive scheduling, EDF is not optimal



Earliest Deadline First

… unless the algorithm has knowledge of the future!