Dynamic Priorities

Real Time Operating Systems and Middleware

Luca Abeni
abeni@dit.unitn.it

Dynamic Priorities

- RM and DM are optimal *fixed priority* assignments
- Maybe we can improve schedulability by using *dynamic priorities*?
  - Fixed priority scheduling: a task τ always has the same priority
  - Dynamic priority scheduling: τ's priority can change during time...
  - Assumption: priorities change from job to job (a job $J_{i,j}$ always has the same priority $p_{h,k}$)

Earliest Deadline First

- Dynamic task priority / fixed job priority
  - Task τ_i's priority can change $\rightarrow$ p_i is not constant
  - Job $J_{i,j}$'s priority does not change $\rightarrow$ $p_{i,j}$ is constant
- Simplest idea: give priority to tasks with the earliest absolute deadline: $d_{i,j} < d_{h,k} \Rightarrow p_{i,j} > p_{h,k}$
- WARNING: absolute deadline, not relative deadline!
  - Earliest Deadline First (EDF)
  - DM $\rightarrow$ *relative* deadlines; EDF $\rightarrow$ *absolute* deadlines

Can We Do any Better than RM?

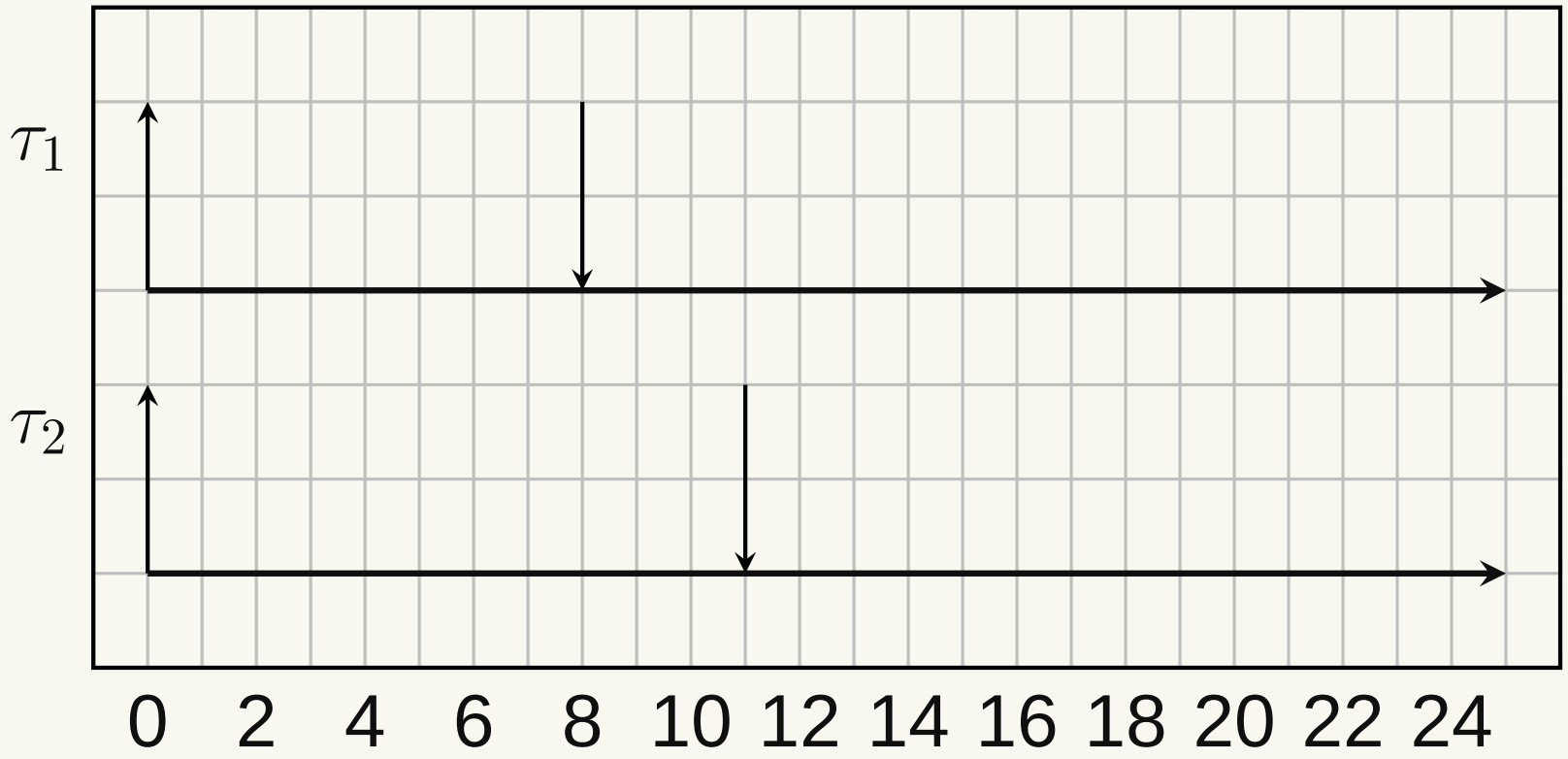
- Yes we can! (of course)
  - Consider a system of periodic tasks with relative deadline equal to the period.
  - The system is schedulable with EDF if and only if

$$\sum_i \frac{C_i}{T_i} \leq 1$$

  - $U_{lub} = 1$!!! Optimal algorithm!!!
- If $D_i \neq T_i$:
  - TDA or RTA can be used... But can be complex!

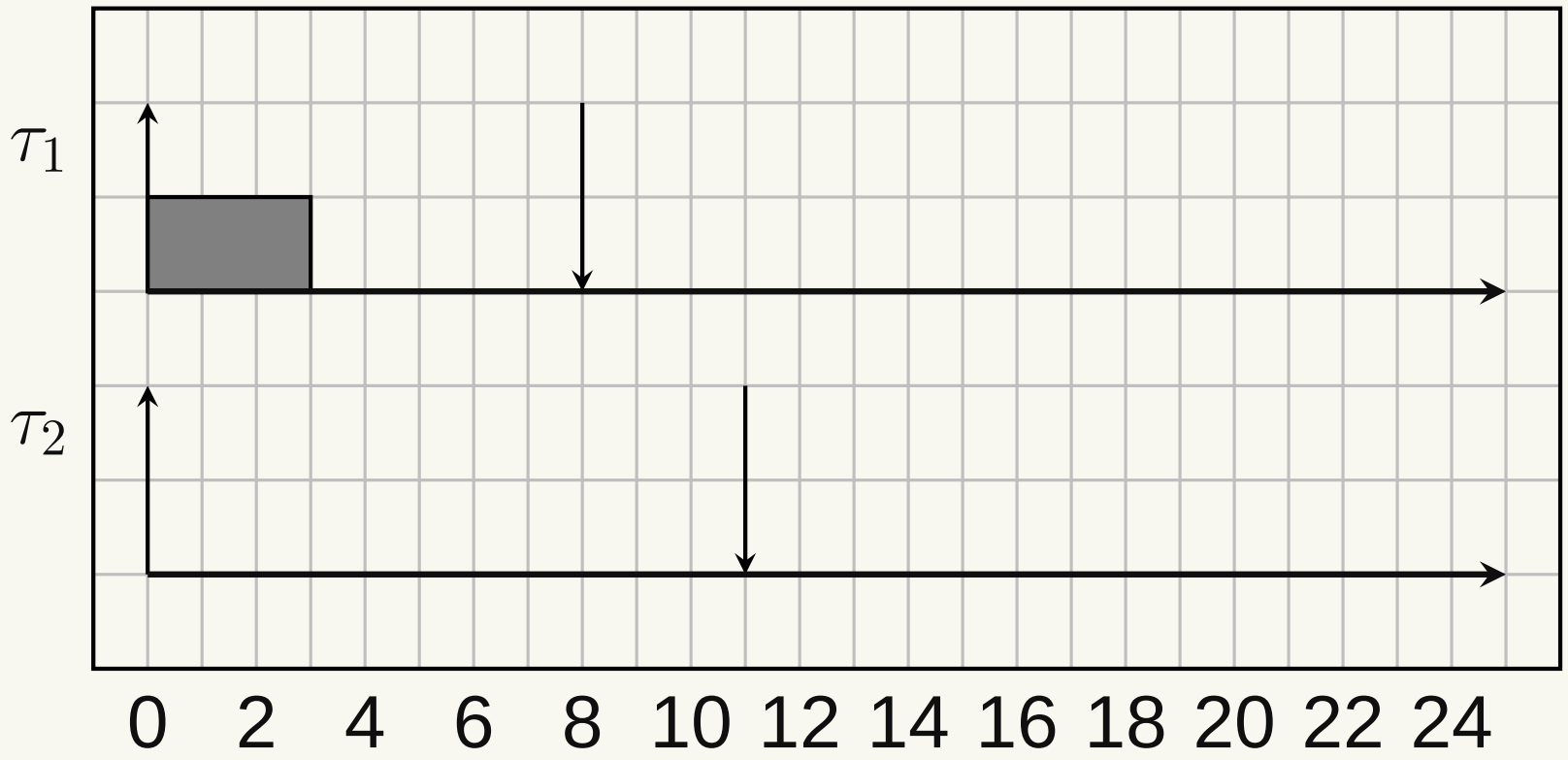
An Example – RM

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



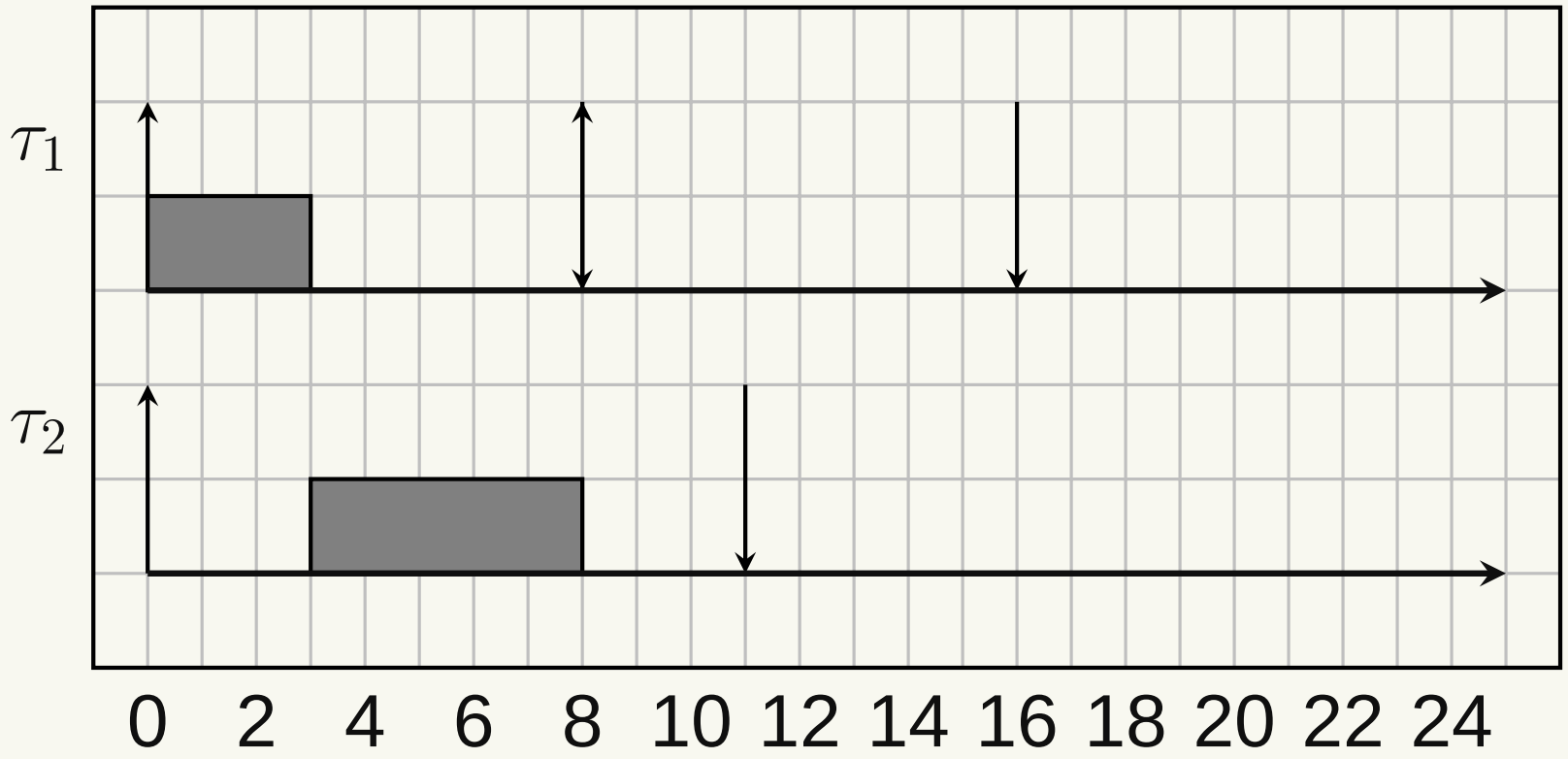
An Example – RM

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



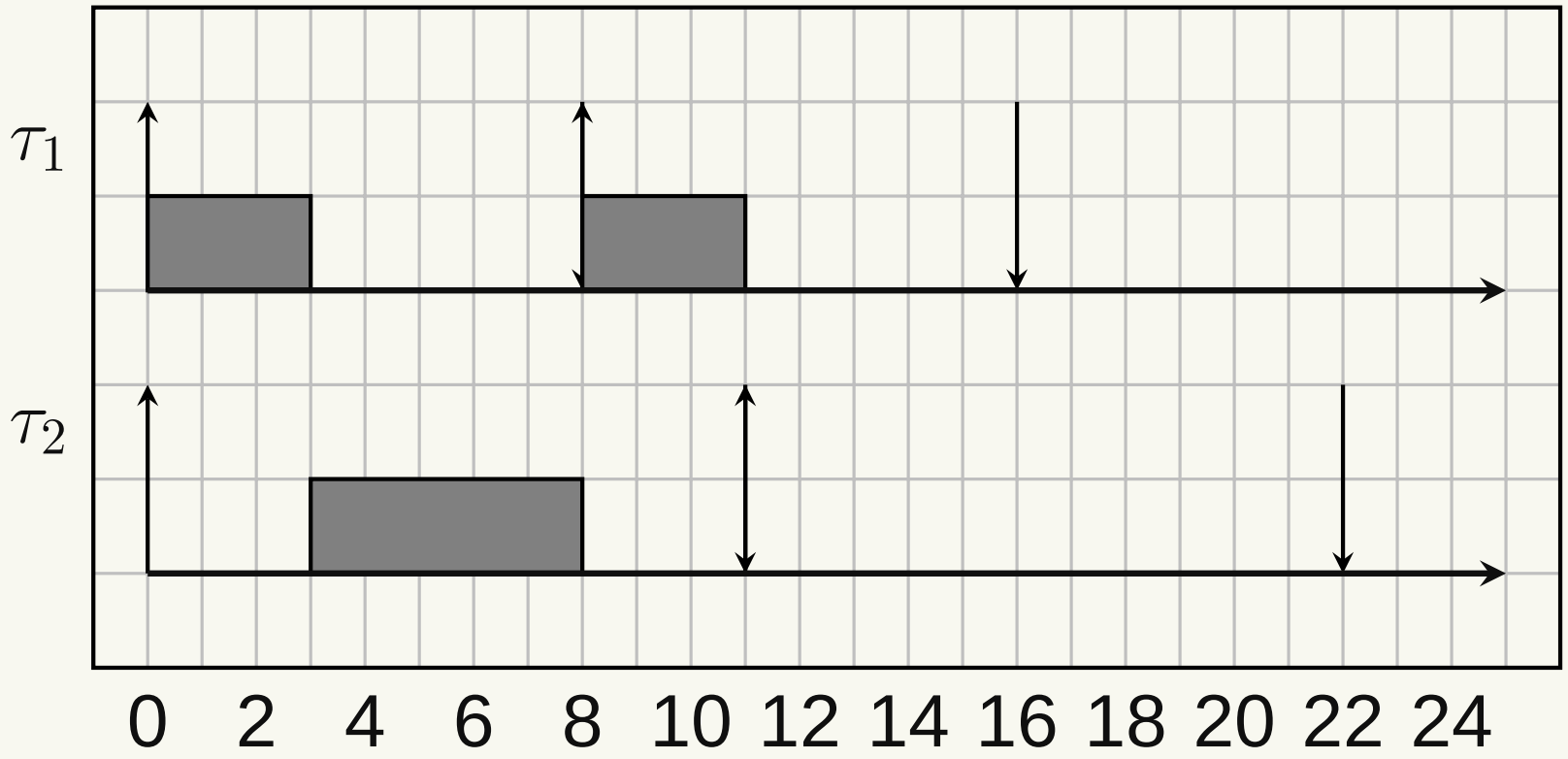
An Example – RM

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



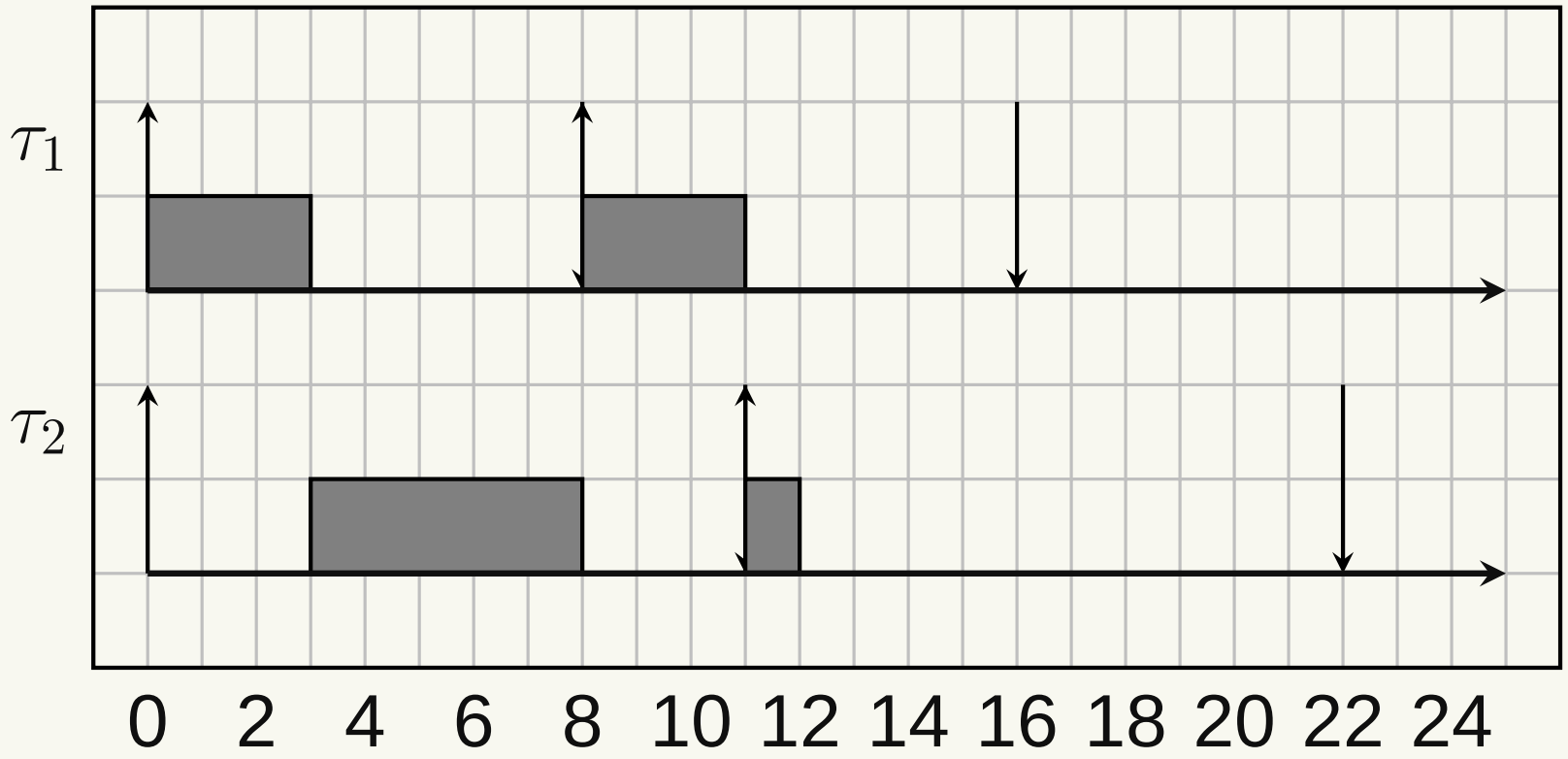
An Example – RM

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



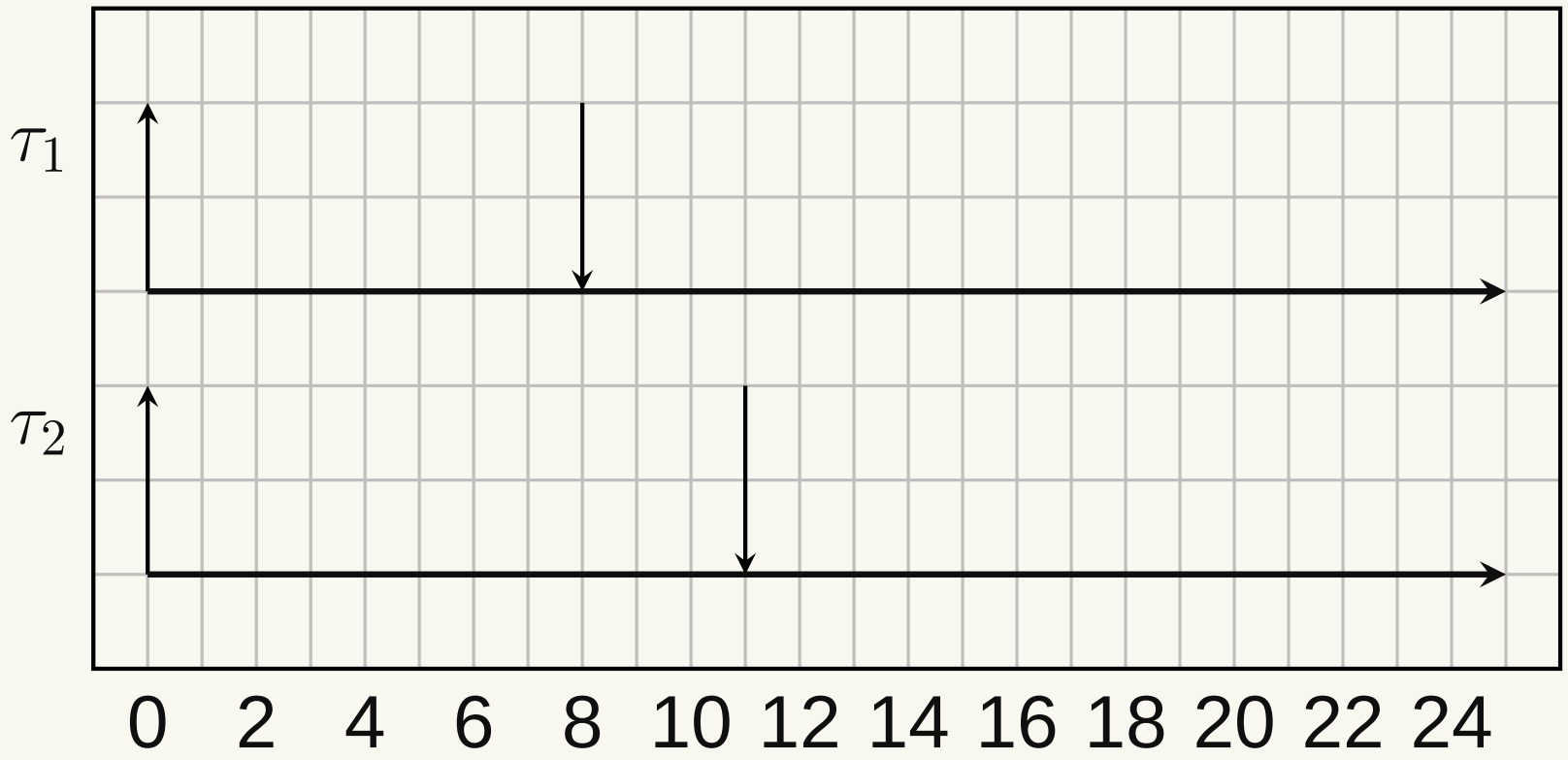
An Example – RM

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



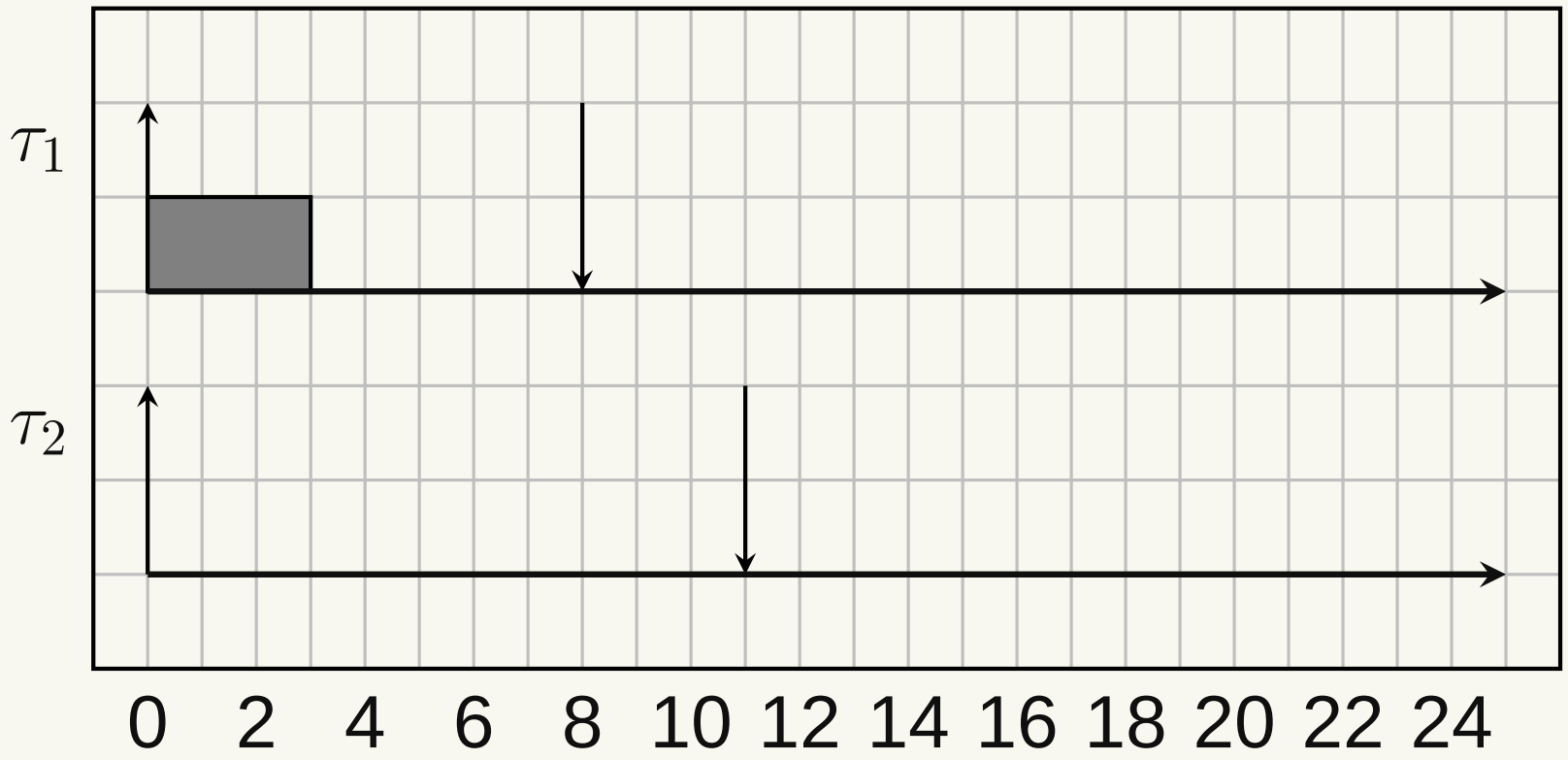
The Same Example – EDF

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



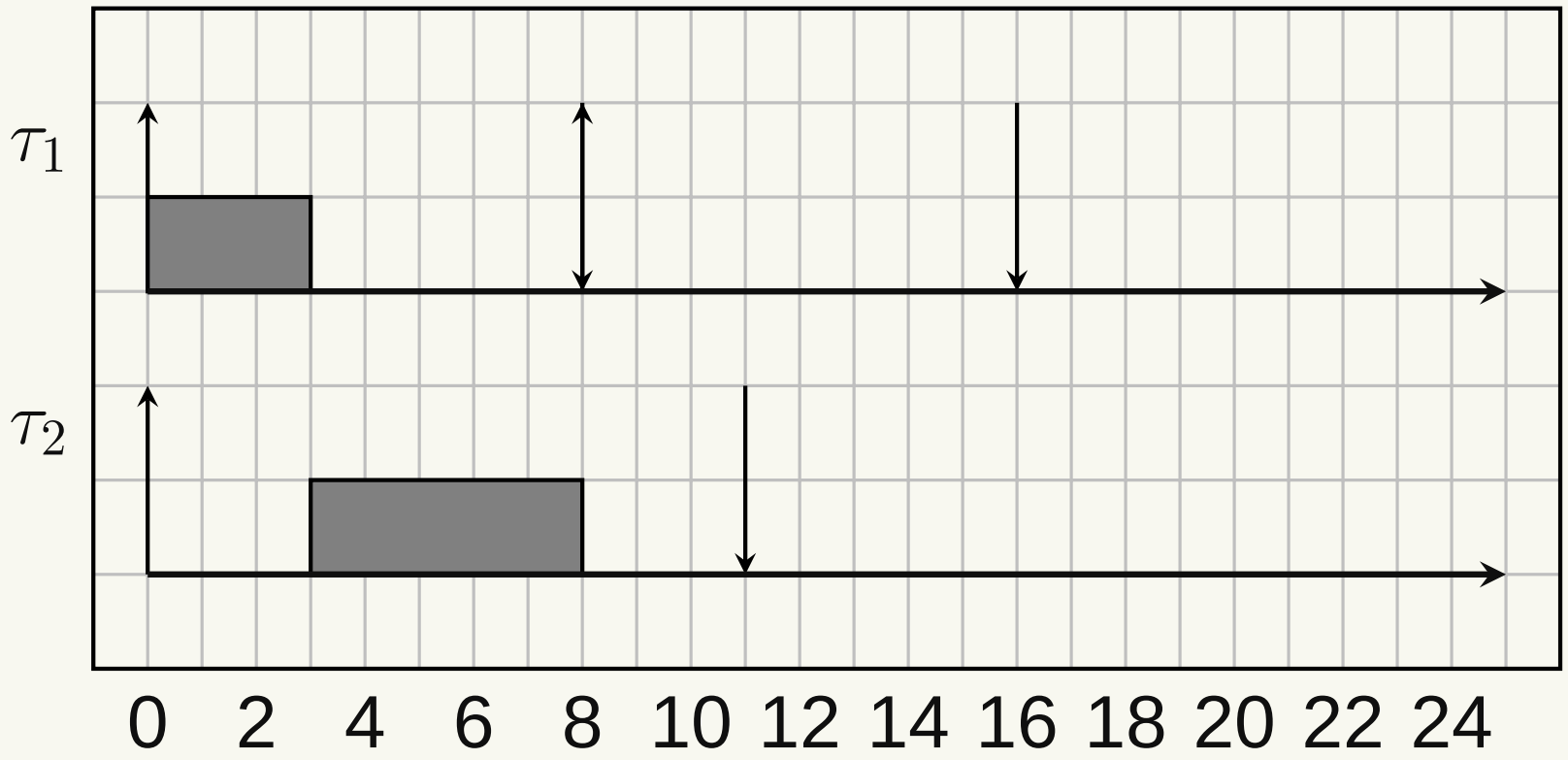
The Same Example – EDF

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



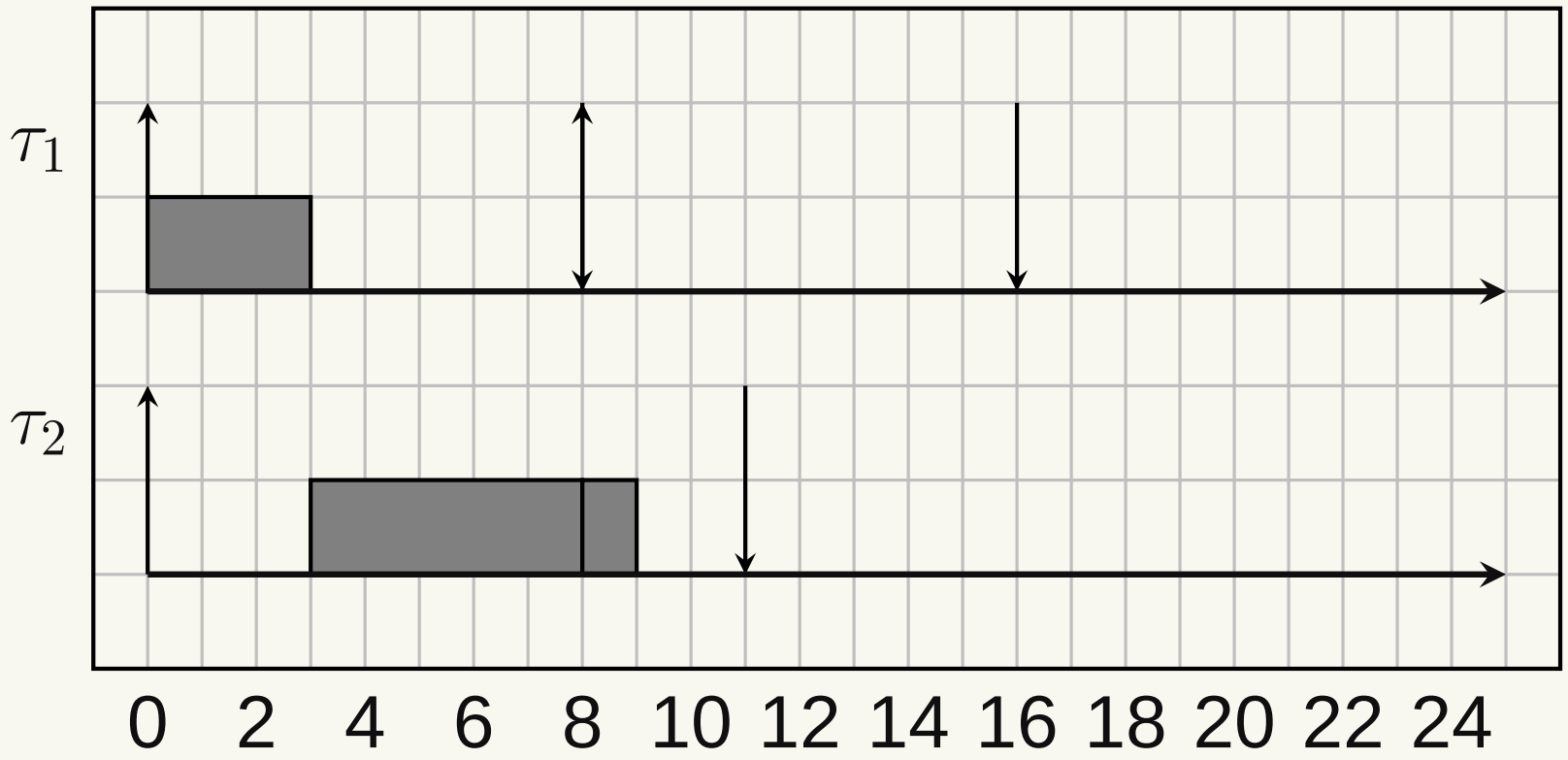
The Same Example – EDF

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



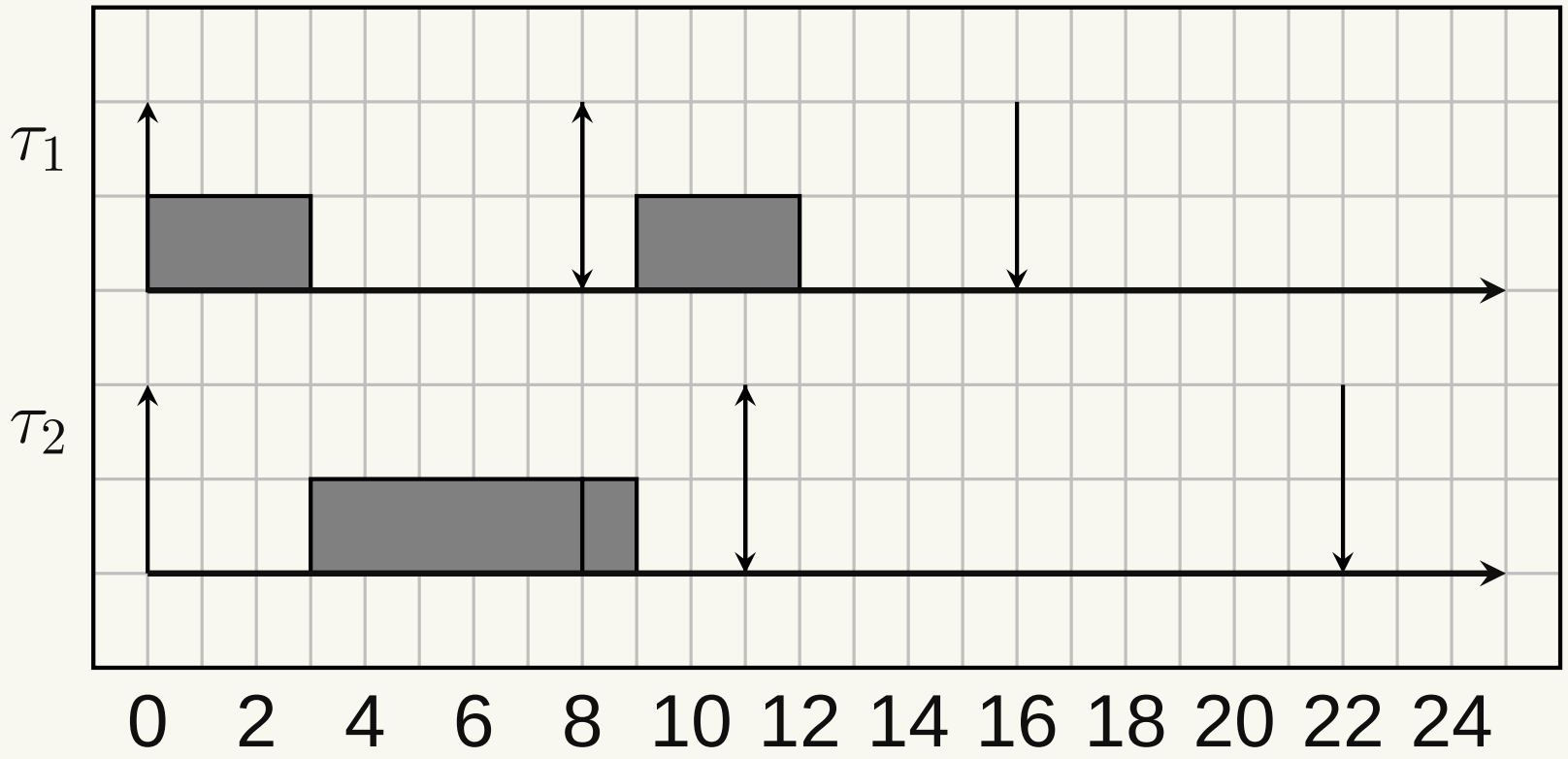
The Same Example – EDF

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



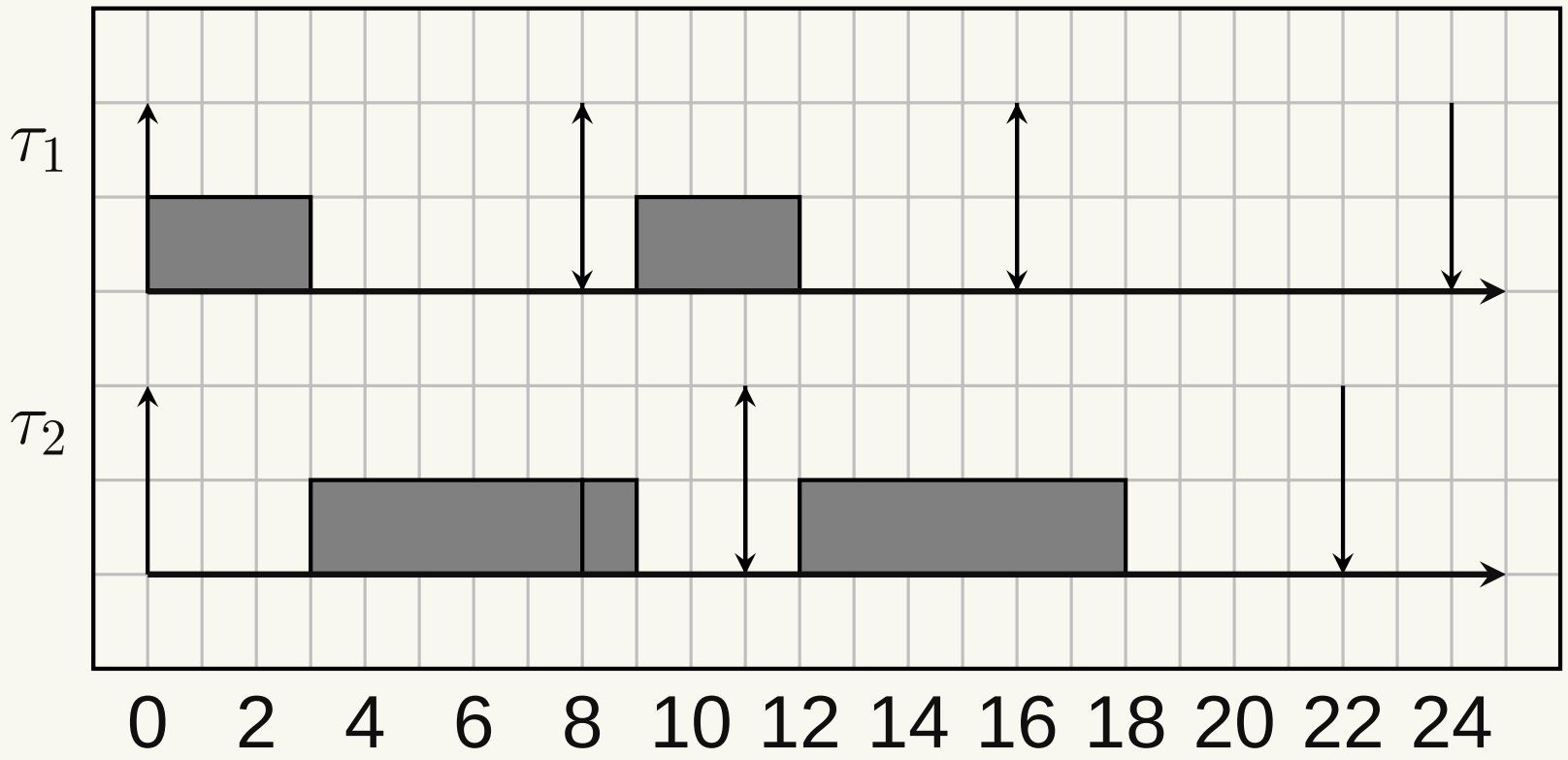
The Same Example – EDF

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



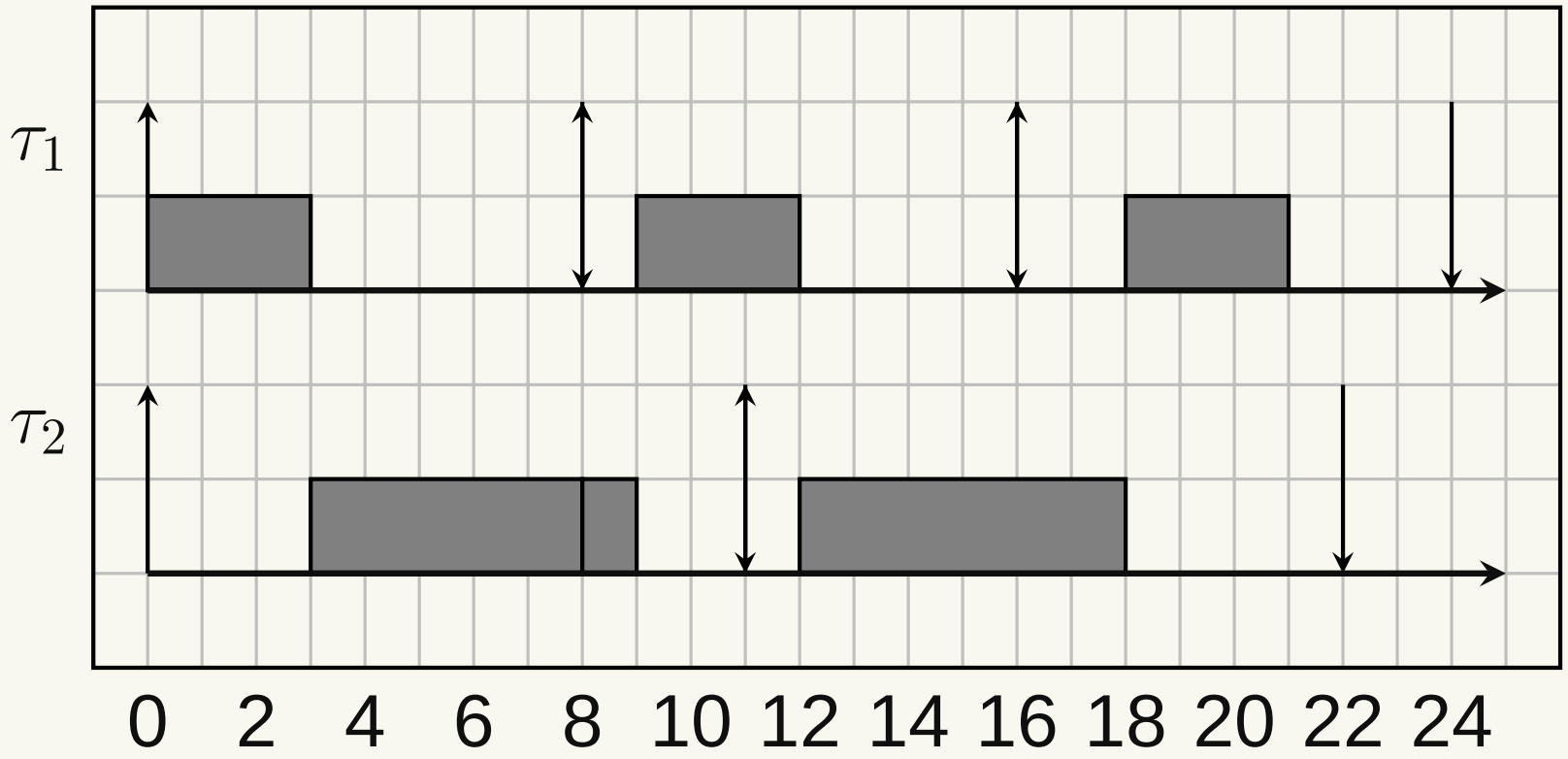
The Same Example – EDF

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



The Same Example – EDF

- $\tau_1 = (3, 8, 8)$, $\tau_2 = (6, 11, 11) \Rightarrow U = 0.92$



Is EDF so Wonderful?

- First answer would be “yes”
- But it is not so well supported by mainline OS (or even RTOS)...
  - Why???
- Up to some time ago, no widely used RTOS provided EDF
  - But things are rapidly changing!
  - A scheduling policy based on EDF is in mainline Linux since 3.14!!!