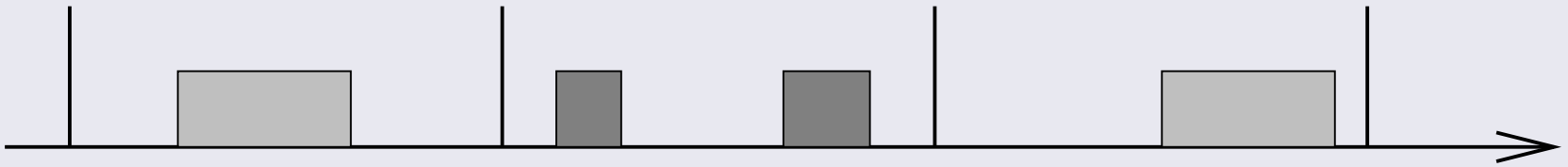
Supplied Time for an Periodic Server

Luca Abeni
luca.abeni@santannapisa.it

May 8, 2018

Periodic Servers

- Periodic Server $\mathcal{S} = (Q, P)$: guarantees Q units of time every period P
  - Can be implemented in different ways (example: CBS)
- Different from static allocation: we do not know where in the period the Q time units are allocated
  - Execution inside a period can even be preempted!

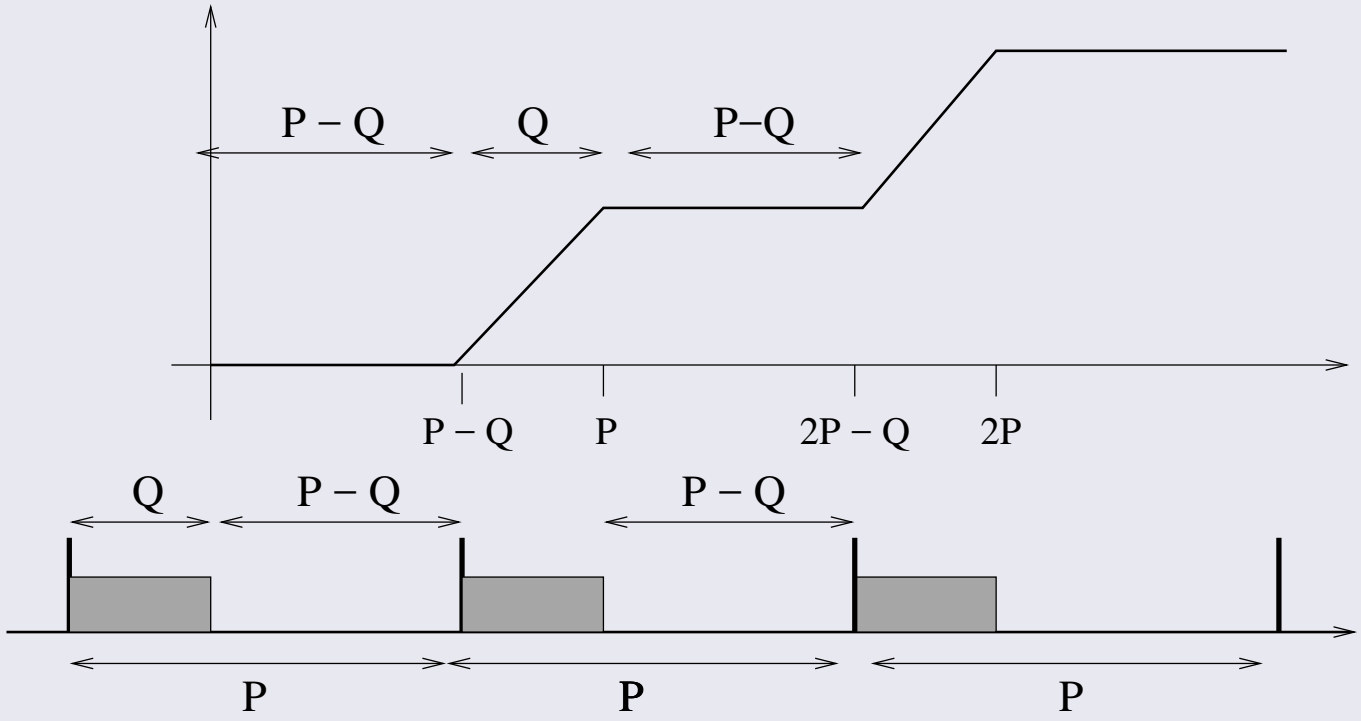


Periodic Servers — Supplied Time

- $sbf(t)$: minimum amount of time that a VM is guaranteed to receive in a time interval of size t
  - Consider all the possible intervals of size t...
    - As already seen for static time partitioning
  - ...And all the possible "legal CPU allocations" generated by the periodic server!
- Big difference with static time partitioning: consider all the possible allocations of Q in the period

The Wrong Solution

- Immagine Q is allocated at the beginning of the period
  - Worst case allocation: $t0$ immediately after Q
  - The time interval starts when the root scheduler deschedules the component



The Wrong Solution — 2

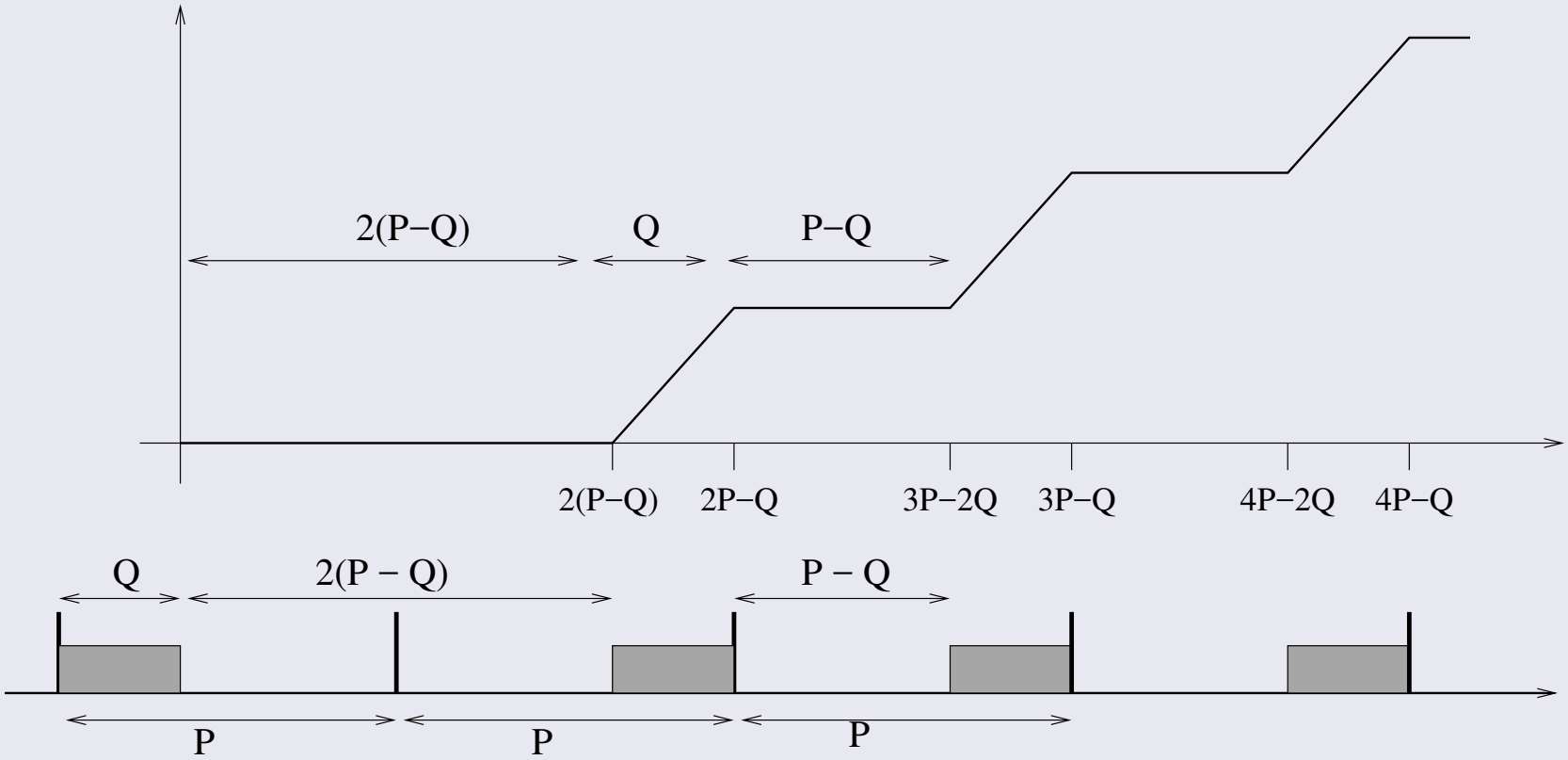
- Supplied time: 0 until $P - Q$...
- ...Then increases with slope 1 until P...
- ...Then flat again until $2P - Q$...
- ...

$$sbf(t) = \begin{cases} 0 & \text{if } t < (P - Q) \\ (n-1)Q & \text{if } (n-1)P \leq t < nP - Q \\ t + nQ - (n-1)P & \text{if } nP - Q \leq t < nP \end{cases}$$

Why Wrong?

- The previous computation assumed Q always at the beginning of a period...
- ...But this is not the worst case!
  - Think about the second period...
  - ...What happens if the root scheduler delays the allocation?
  - The initial "0 allocation period" increases!!!
- Worst-case schedule: Q at the beginning of the first period and at the end of the second one
  - See the difference with static time partitioning?

Considering the Worst-Case Situation



$$sbf(t) = \begin{cases} 0 & \text{if } t < 2(P-Q) \\ (n-1)Q & \text{if } nP - Q \leq t < (n+1)P - 2Q \\ t - (n+1)(P-Q) & \text{if } (n+1)P - 2Q \leq t < (n+1)P - Q \end{cases}$$