Lab 3
Response Time Analysis using FpsCalc

Behnam Khodabandeloo

29 Nov. 2024

based on Duc Anh Nguyen’s slide from last year

Lab Instruction

- Find the assignment instruction file on Studium
- Read FpsCalc User Manual and try
- Focus is on the theory and concepts of Response time Analysis
- FpsCalc is just a helping tool to make things easier. Don't struggle with too much details of it.
- Lab sessions not mandatory, but recommended (TAs are there)

Key dates

Wed 27	Thu 28	Fri 29 (today) Lab 3 intro	Sat 30	Sun 1	Mon 2 Lab 2 demonstrati on	Tue 3

Wed 4	Thu 5 Lab 3 session	Fri 6 Lab 4 Intro	Sat 7	Sun 8	Mon 9 Lab 3 deadline	Tue 10

Wed 11	Thu 12 Lab 4 session	Fri 13	Sat 14	Sun 15	Mon 16	Tue 17

Lab Hand-in

- A pdf report containing the answers to the questions. When asked for, illustrate your answers by drawing timing schedules.
- All the fps files.
- A group has to upload one joint lab report/code before the deadline
- Late submission budget: 8 days
- A late submission subtracts corresponding days from all group member’s budget
- No late submission possible without enough budget!
- Special cases: contact the TA for extension

Lab goals

- Practice response time analysis
- Manual calculation, critical instant charts, tool FpsCalc
- Integrate blocking, jitter

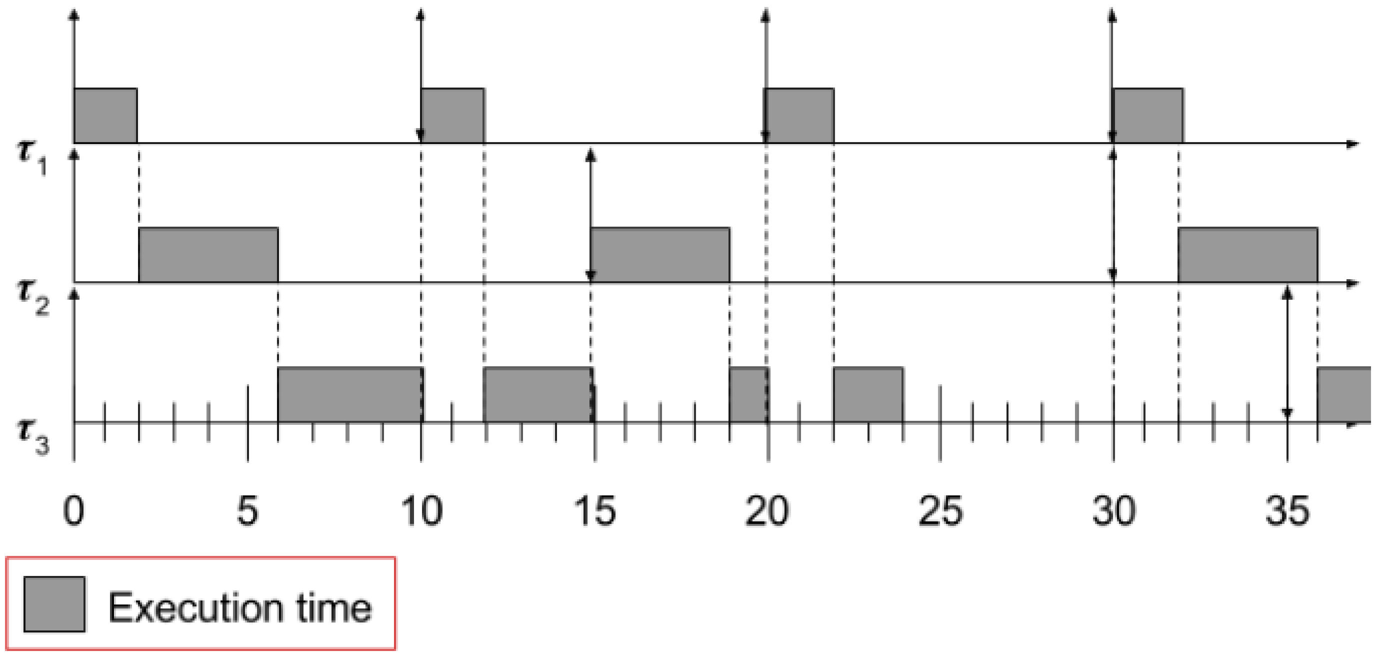
$$R_i = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil \cdot C_j$$

Theory review

Response time analysis

Task	C_i	T_i	D_i	P_i
τ_1	2 ms	10 ms	10 ms	1
τ_2	4 ms	15 ms	15 ms	2
τ_3	10 ms	35 ms	35 ms	3

Critical instant schedule:



Response time analysis for Fixed Priority Scheduling

Task	C_i	T_i	D_i	P_i
τ_1	2 ms	10 ms	10 ms	1
τ_2	4 ms	15 ms	15 ms	2
τ_3	10 ms	35 ms	35 ms	3

$$R_i^{n+1} = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^n}{T_j} \right\rceil C_j$$

How to calculate this recursive equation?

UPPSALA UNIVERSITET

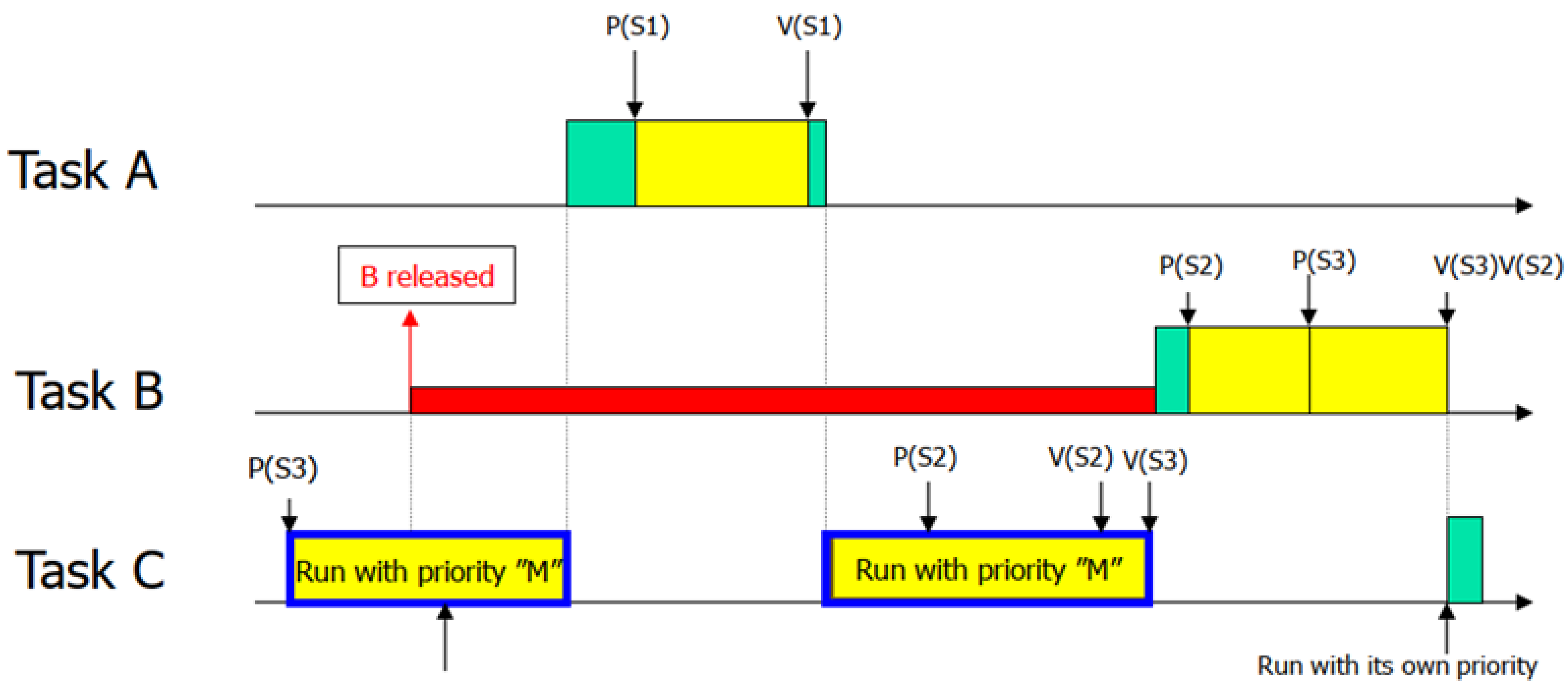
Blocking

$$R_i = C_i + B_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j$$

Task i can be blocked by lower priority tasks when resource sharing exists!

Blocking

$$R_i = C_i + B_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j$$



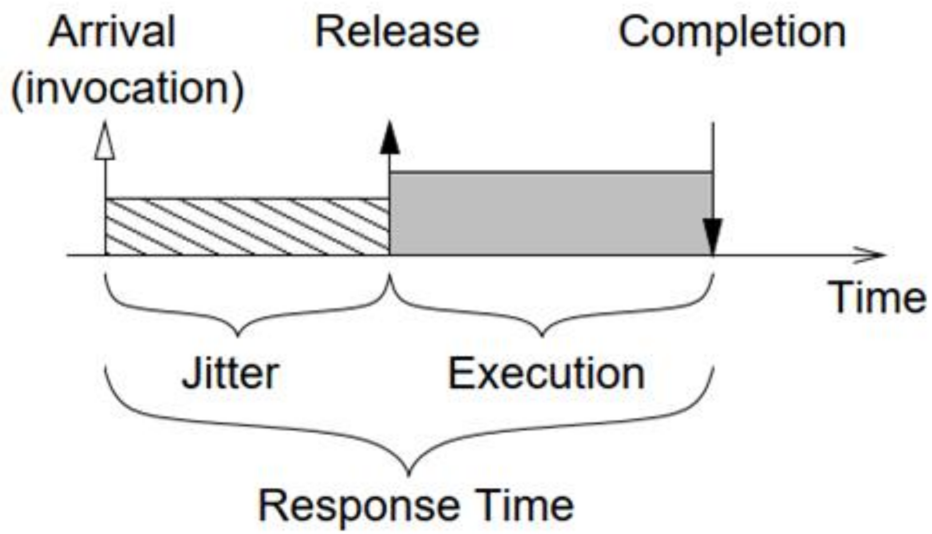
Blocking

$$R_i = C_i + B_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j$$

B_i has different ways to calculate depend on the resource sharing policy

The assignment pdf describes a simplified way to calculate it

Jitter



The jitter Ji for a task i is the difference between the maximal jitter and the minimal jitter.

$$J_i = J_i^{max} - J_i^{min}$$

Jitter

$$w_i = C_i + \sum_{j \in hp(i)} \left\lceil \frac{w_i + J_j}{T_j} \right\rceil C_j$$

$$= C_i + \sum_{j \in hp(i)} \left(1 + \left\lceil \frac{w_i - (T_j - J_j)}{T_j} \right\rceil \right) C_j$$

$$R_i = w_i + J_i$$

Utilization bound

- Based on system's utilization bound $U := \sum_{i \leqslant n} C_i / T_i$
- For EDF: $U \leqslant 1 \iff \tau$ schedulable (sufficient and necessary)
- For RM: $U \leqslant n(2^{1/n} - 1) \implies \tau$ schedulable (only sufficient!)

Task	C_i	T_i	D_i	P_i
τ_1	2 ms	10 ms	10 ms	1
τ_2	4 ms	15 ms	15 ms	2
τ_3	10 ms	35 ms	35 ms	3

$U = 2/10 + 4/15 + 10/35 = 0.752381 < 3*(2^{1/3}) - 1 = 0.7798$

UPPSALA UNIVERSITET

Utilization bound

- Based on system's utilization bound $U := \sum_{i \leqslant n} C_i / T_i$
- For EDF: $U \leqslant 1 \iff \tau$ schedulable (sufficient and necessary)
- For RM: $U \leqslant n(2^{1/n} - 1) \implies \tau$ schedulable (only sufficient!)

Task	C_i	T_i	D_i	P_i
τ_1	2 ms	10 ms	10 ms	1
τ_2	4 ms	15 ms	15 ms	2
τ_3	10 ms	35 ms	35 ms	3

U = 2/10 + 4/15 + 10/35 = 0.752381 < 3*($2^{1/3}$) – 1 = 0.7798

RM Schedulable!

Utilization bound

- Based on system's utilization bound $U := \sum_{i \leqslant n} C_i / T_i$
- For EDF: $U \leqslant 1 \iff \tau$ schedulable (sufficient and necessary)
- For RM: $U \leqslant n(2^{1/n} - 1) \implies \tau$ schedulable (only sufficient!)

Task	C_i	T_i	D_i	P_i
τ_1	2 ms	10 ms	10 ms	1
τ_2	4 ms	15 ms	15 ms	2
τ_3	10 ms	35 ms	35 ms	3

U = 0.752381 < 1

Utilization bound

- Based on system's utilization bound $U := \sum_{i \leqslant n} C_i / T_i$
- For EDF: $U \leqslant 1 \iff \tau$ schedulable (sufficient and necessary)
- For RM: $U \leqslant n(2^{1/n} - 1) \implies \tau$ schedulable (only sufficient!)

Task	C_i	T_i	D_i	P_i
τ_1	2 ms	10 ms	10 ms	1
τ_2	4 ms	15 ms	15 ms	2
τ_3	10 ms	35 ms	35 ms	3

U = 0.752381 < 1

EDF Schedulable!

dbf: Demand bound function

dbf(T,t) = Sum of all WCET of job instances from task set T that has the scheduling window (from arrival to deadline) that is fully inside *any* time interval of length t

$$dbf(\tau, t) = \sum_{i=1}^{n} \lfloor \frac{t + T_i - D_i}{T_i} \rfloor C_i = \sum_{i=1}^{n} \lfloor \frac{t - D_i}{T_i} + 1 \rfloor C_i$$

UPPSALA UNIVERSITET

The end
Good luck