TU/e

2IMN20 - Real-Time Systems

Response-time analysis for fixed-priority preemptive scheduling (2/2)

Geoffrey Nelissen

2023-2024

Disclaimer:
Some slides were provided by Dr. Reinder Bril

IRiS

Agenda

- **Worst-case response-time analysis** under **preemptive Fixed priority** scheduling
  - Reminder
  - Generalization (arrival curves)

- **Best-case response time analysis** for preemptive **Fixed priority** scheduling

- **Worst-case schedulability analysis** for periodic or sporadic tasks under **preemptive EDF** scheduling


QUIZ TIME

Quiz

For fixed priority preemptive scheduling. How do you call the execution scenario leading to the WCRT of all tasks?

- The optimal release
- The worst-case execution time
- The worst arrival
- The critical instant

Quiz

For fixed priority preemptive scheduling. How do call the execution scenario leading to the WCRT of all tasks?

- The optimal release
- The worst-case execution time
- The worst arrival
- **The critical instant**

Quiz

For fixed priority preemptive scheduling.
Which of the following statements is/are true for a critical instant?

- All tasks have their first job arriving at the same time
- All jobs execute for their worst-case execution time
- All tasks release a first job at the same time
- All jobs are released with their maximum jitter
- The worst-case response time is experienced by the first job of each task
- Jobs are released as slow as possible

Quiz

For fixed priority preemptive scheduling.
Which of the following statements are true for a critical instant?

- All tasks have their first job arriving at the same time
- **All jobs execute for their worst-case execution time**
- **All tasks release a first job at the same time**
- All jobs are released with their maximum jitter
- The worst-case response time is experienced by the first job of each task
- Jobs are released as slow as possible

Quiz

Which test(s) is/are a **necessary schedulability test** for fixed priority preemptive scheduling of independent sporadic tasks with **constrained deadlines** and no release jitter?

1. $\sum_{k=1}^{n} U_k \leq 1$
2. $\prod_{i=1}^{n} (U_i + 1) \leq 2$
3. $\forall \tau_i,\ R_i \leq D_i$ with $R_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{R_i}{T_k} \right\rceil \cdot C_k$
4. $\forall \tau_i,\quad C_i + \sum_{k=1}^{i-1} \left\lceil \frac{D_i}{T_k} \right\rceil \cdot C_k \leq D_i$

5 $\forall \tau_i,\ R_i \leq D_i$ with $R_i = \max_j \{X_{i,j} + \sigma_i - (j-1) \times T_i\}$

with $X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_{i,j} + \sigma_k}{T_k} \right\rceil \cdot C_k$

Quiz

Which test(s) is/are a **sufficient schedulability test** for fixed priority preemptive scheduling of independent sporadic tasks with **constrained deadlines** and no release jitter?

- $\sum_{k=1}^{n} U_k \leq 1$
- $\prod_{i=1}^{n} (U_i + 1) \leq 2$
- $\forall \tau_i,\ R_i \leq D_i$ with $R_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{R_i}{T_k} \right\rceil \cdot C_k$
- $\forall \tau_i,\quad C_i + \sum_{k=1}^{i-1} \left\lceil \frac{D_i}{T_k} \right\rceil \cdot C_k \leq D_i$
- $\forall \tau_i,\ R_i \leq D_i$ with $R_i = \max_j \{X_{i,j} + \sigma_i - (j-1) \times T_i\}$

  with $X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_{i,j} + \sigma_k}{T_k} \right\rceil \cdot C_k$

Quiz

Which test(s) is/are an **exact schedulability test** for fixed priority preemptive scheduling of independent sporadic tasks with **constrained deadlines** and no release jitter?

- $\sum_{k=1}^{n} U_k \leq 1$
- $\prod_{i=1}^{n} (U_i + 1) \leq 2$
- $\forall \tau_i,\ R_i \leq D_i$ with $R_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{R_i}{T_k} \right\rceil \cdot C_k$
- $\forall \tau_i,\quad C_i + \sum_{k=1}^{i-1} \left\lceil \frac{D_i}{T_k} \right\rceil \cdot C_k \leq D_i$
- $\forall \tau_i,\ R_i \leq D_i$ with $R_i = \max_j \{X_{i,j} + \sigma_i - (j-1) \times T_i\}$

  with $X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_{i,j} + \sigma_k}{T_k} \right\rceil \cdot C_k$

Quiz

Which test is a **necessary/sufficient/exact** **schedulability test** for fixed priority preemptive scheduling of independent sporadic tasks with **constrained deadlines** and no release jitter?

- $\sum_{k=1}^{n} U_k \leq 1$ — **Necessary**
- $\prod_{i=1}^{n} (U_i + 1) \leq 2$ — **None**
- $\forall \tau_i,\ R_i \leq D_i$ with $R_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{R_i}{T_k} \right\rceil \cdot C_k$ — **Necessary, sufficient, exact**
- $\forall \tau_i,\quad C_i + \sum_{k=1}^{i-1} \left\lceil \frac{D_i}{T_k} \right\rceil \cdot C_k \leq D_i$ — **Sufficient**
- $\forall \tau_i,\ R_i \leq D_i$ with $R_i = \max_j \{X_{i,j} + \sigma_i - (j-1) \times T_i\}$

  with $X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_{i,j} + \sigma_k}{T_k} \right\rceil \cdot C_k$ — **Necessary, sufficient, exact**

Quiz

What is the WCRT of task τ_1 when scheduled by rate monotonic priorities?

τ_i	C_i	T_i	D_i	U_i
τ_1	2	5	5	0.4
τ_2	4	10	10	0.4
τ_3	1	25	25	0.04

$U = 0.84$

Reminder:

$$R_i = \sigma_i + X_i$$

where

$$X_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_i + \sigma_k}{T_k} \right\rceil \cdot C_k$$

- 2
- 4
- 5
- 8
- 9
- 10

Quiz

What is the WCRT of task τ_2 when scheduled by rate monotonic priorities?

τ_i	C_i	T_i	D_i	U_i
τ_1	2	5	5	0.4
τ_2	4	10	10	0.4
τ_3	1	25	25	0.04

$U = 0.84$

Reminder:

$$R_i = \sigma_i + X_i$$

where

$$X_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_i + \sigma_k}{T_k} \right\rceil \cdot C_k$$

- 2
- 4
- 5
- 8
- 9
- 10

Exercise

- find the WCRT of tasks τ_1 and τ_2 when they are scheduled by rate monotonic priorities

τ_i	C_i	T_i	D_i	U_i
τ_1	2	5	5	0.4
τ_2	4	10	10	0.4
τ_3	1	25	25	0.04

Reminder:

$$R_i = \sigma_i + X_i$$

where

$$X_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_i + \sigma_k}{T_k} \right\rceil \cdot C_k$$

$$R_1^{(0)} = 2$$

$$R_1^{(1)} = 2 + \sum_{k=1}^{1-1} \left\lceil \frac{R_1^{(0)}}{T_k} \right\rceil \cdot C_k = 2$$

We stop here since $R_1^{(n)} \leq R_1^{(n-1)}$ ➡ **WCRT of τ_1 is 2**

Do the same for τ_3

Exercise

- find the WCRT of tasks τ_1 and τ_2 when they are scheduled by rate monotonic priorities

τ_i	C_i	T_i	D_i	U_i
τ_1	2	5	5	0.4
τ_2	4	10	10	0.4
τ_3	1	25	25	0.04

Reminder:

$$R_i = \sigma_i + X_i$$

where

$$X_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_i + \sigma_k}{T_k} \right\rceil \cdot C_k$$

$$R_2^{(0)} = 4$$

$$R_2^{(1)} = 4 + \sum_{k=1}^{2-1} \left\lceil \frac{R_2^{(0)}}{T_k} \right\rceil \cdot C_k = 4 + \left\lceil \frac{4}{5} \right\rceil \cdot 2 = 6$$

$R_2^{(1)} \leq R_2^{(0)}$? **NO, so continue**

$$R_2^{(2)} = 4 + \sum_{k=1}^{2-1} \left\lceil \frac{R_2^{(1)}}{T_k} \right\rceil \cdot C_k = 4 + \left\lceil \frac{6}{5} \right\rceil \cdot 2 = 8$$

$R_2^{(2)} \leq R_2^{(1)}$? **NO, so continue**

$$R_2^{(3)} = 4 + \sum_{k=1}^{2-1} \left\lceil \frac{R_2^{(2)}}{T_k} \right\rceil \cdot C_k = 4 + \left\lceil \frac{8}{5} \right\rceil \cdot 2 = 8$$

$R_2^{(3)} \leq R_2^{(2)}$? **Yes, so stop**

$\Rightarrow$ **WCRT of τ_2 is 8**

Exercise

- find the WCRT of tasks τ_1 and τ_2 when they are scheduled by rate monotonic priorities

τ_i	C_i	T_i	D_i	U_i
τ_1	2	5	5	0.4
τ_2	4	10	10	0.4
τ_3	1	25	25	0.04

Reminder:

$$R_i \quad = \sigma_i + X_i$$

where

$$X_i \quad = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_i \quad + \sigma_k}{T_k} \right\rceil \cdot C_k$$

$$R_3^{(0)} = 1$$

$$R_3^{(1)} = 1 + \sum_{k=1}^{3-1} \left\lceil \frac{R_3^{(0)}}{T_k} \right\rceil \cdot C_k = 1 + \left\lceil \frac{1}{5} \right\rceil \cdot 2 + \left\lceil \frac{1}{10} \right\rceil \cdot 4 = 7$$

$R_3^{(1)} \leq R_3^{(0)}$? **NO, so continue**

$$R_3^{(2)} = 1 + \sum_{k=1}^{2-1} \left\lceil \frac{R_3^{(1)}}{T_k} \right\rceil \cdot C_k = 1 + \left\lceil \frac{7}{5} \right\rceil \cdot 2 + \left\lceil \frac{7}{10} \right\rceil \cdot 4 = 9$$

$R_3^{(2)} \leq R_3^{(1)}$? **NO, so continue**

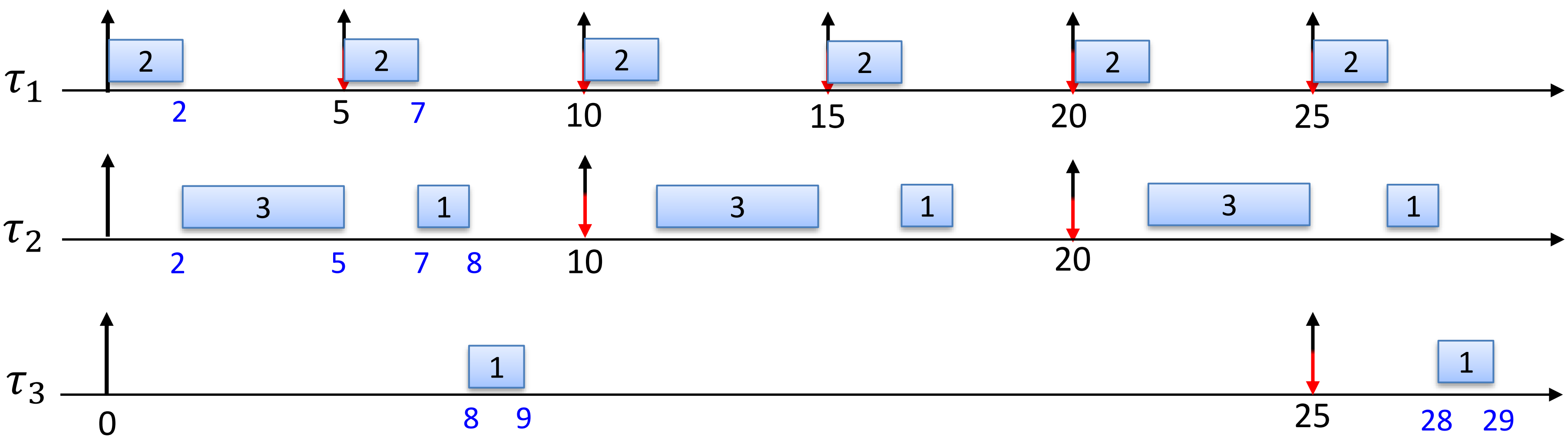
$$R_3^{(3)} = 1 + \sum_{k=1}^{2-1} \left\lceil \frac{R_3^{(2)}}{T_k} \right\rceil \cdot C_k = 1 + \left\lceil \frac{9}{5} \right\rceil \cdot 2 + \left\lceil \frac{9}{10} \right\rceil \cdot 4 = 9$$

$R_3^{(3)} \leq R_3^{(2)}$? **Yes, so stop**

➡ **WCRT of τ_3 is 9**

Visualizing the WCRT: the critical instant

τ_i	C_i	T_i	D_i	U_i	R_i
τ_1	2	5	5	0.4	2
τ_2	4	10	10	0.4	8
τ_3	1	25	25	0.04	9




QUIZ TIME

Agenda

- **Worst-case response-time analysis** under **preemptive Fixed priority** scheduling
  - Reminder
  - **Generalization (arrival curves)**

- **Best-case response time analysis** for periodic tasks under preemptive **Fixed priority** scheduling

- **Worst-case schedulability analysis** for periodic or sporadic tasks under **preemptive EDF** scheduling

The most generic WCRT test we learned so far

Level-i busy window

$$L_i = \sum_{k=1}^{i} \left\lceil \frac{L_i + \sigma_k}{T_k} \right\rceil \cdot C_k$$

$$N_i = \left\lceil \frac{L_i + \sigma_i}{T_i} \right\rceil$$

Jobs finishing times relative to the first job release time

$$X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_{i,j} + \sigma_k}{T_k} \right\rceil \cdot C_k$$

Valid for periodic/sporadic independent tasks with release jitter and arbitrary deadlines

Worst-case response time

$$R_{i,j} = X_{i,j} + \sigma_i - (j-1) \times T_i$$

$$R_i = \max_{1 \le j \le N_i} \{R_{i,j}\}$$

WCRT of τ_i

IRiS

The most generic WCRT test we learned so far

Level-i busy window

$L_i = \sum^{i} \left\lceil \frac{L_i + \sigma_k}{\ } \right\rceil \cdot C_k$ $\quad N_i = \left\lceil \frac{L_i + \sigma_i}{\ } \right\rceil$

Job

What if the tasks are not periodic or sporadic?

Is there a way to extend the existing test?

Wo

What tools do we know to model any arbitrary release patterns?

$R_{i,j} = X_{i,j} + \sigma_i - (j-1) \times T_i$ $\quad R_i = \max_{1 \le j}$

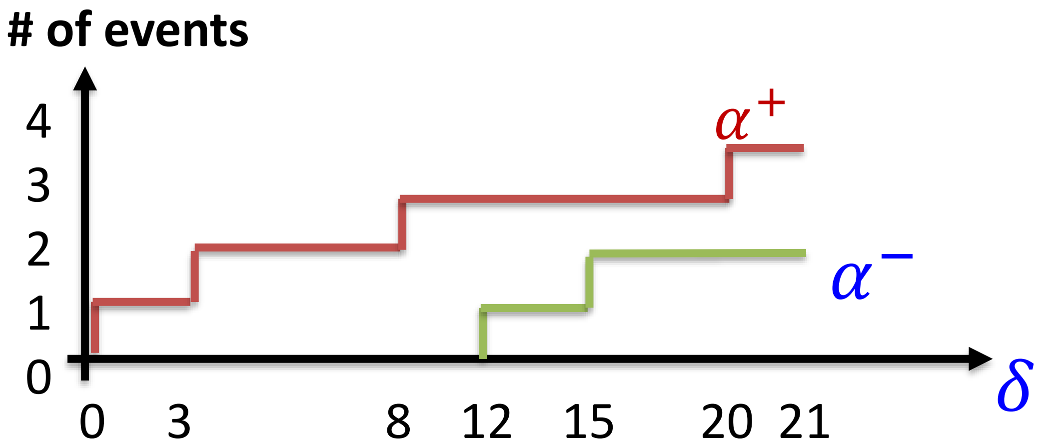
Arrival curves

Modelling complex arrival patterns with arrival curves

- An arrival curve represents the **lower bound** and **upper bound** on the **number of events in any time interval**.

α^+ = maximum number of events in any interval of duration δ

α^- = minimum number of events in any interval of duration δ

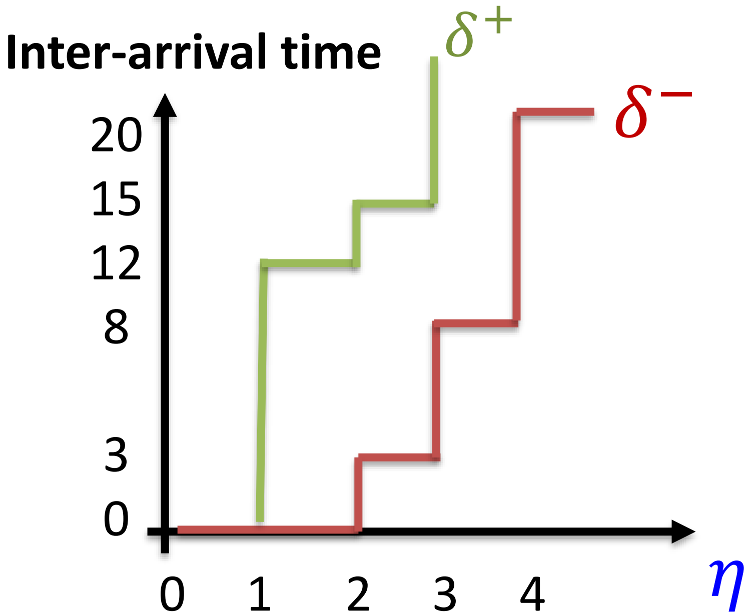


Inverse function of arrival curves

- The inverse of arrival curves provide a **lower bound** and **upper bound** on the **inter-arrival time between consecutive events in a system**.

δ^+ = maximum inter-arrival time between η events

δ^- = minimum inter-arrival time between η events



Understanding the terms

Level-i busy window

$$L_i = \sum_{k=1}^{i} \left\lceil \frac{L_i + \sigma_k}{T_k} \right\rceil \cdot C_k$$

$$N_i = \left\lceil \frac{L_i + \sigma_i}{T_i} \right\rceil$$

Jobs finishing times relative to the first job release time

$$X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_{i,j} + \sigma_k}{T_k} \right\rceil \cdot C_k$$

Worst-case response time

$$R_{i,j} = X_{i,j} + \sigma_i - (j-1) \times T_i$$

$$R_i = \max_{1 \le j \le N_i} \{R_{i,j}\}$$

WCRT of τ_i

IRiS

… the terms

For all higher or equal-priority tasks than τ_i

Count the maximum number of jobs released by task τ_k in the interval $[0, X_{i,j})$

Equivalent to $\alpha_i^+(L_i)$

$$L_i = \sum_{k=1}^{i} \left\lceil \frac{L_i + \sigma_k}{T_k} \right\rceil \cdot C_k$$

Equivalent to $\alpha_k^+(L_i)$

$$N_i = \left\lceil \frac{L_i + \sigma_i}{T_i} \right\rceil$$

Number of jobs of τ_i release in the level-i busy window

j jobs of τ_i must execute until the finish time of the j^{th} jobs of τ_i

…es

For all higher-priority tasks

$$X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \left\lceil \frac{X_{i,j} + \sigma_k}{T_k} \right\rceil \cdot C_k$$

Equivalent to $\alpha_k^+(X_{i,j})$

Equivalent to $\alpha_i^+(X_{i,j})$

Count the maximum number of jobs released by task τ_k in the interval $[0, X_{i,j})$

…ponse time

$$R_{i,j} = X_{i,j} + \sigma_i - (j-1) \times T_i \qquad R_i = \max_{1 \le j \le N_i} \{R_{i,j}\}$$

Understanding the terms

Level-i busy window

$$L_i = \sum_{k=1}^{i} \alpha_k^+(L_i) \cdot C_k$$

$$N_i = \alpha_i^+(L_i)$$

Jobs finishing times relative to the first job release time

$$X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \alpha_k^+(X_{i,j}) \cdot C_k$$

Worst-case response time

$$R_{i,j} = X_{i,j} + \sigma_i - (j-1) \times T_i$$

$$R_i = \max_{1 \le j \le N_i} \{R_{i,j}\}$$

Understanding the terms

Level-i busy window

$$L_i = \sum_{k=1}^{i} \alpha_k^+(L_i) \cdot C_k$$

$$N_i = \alpha_i^+(L_i)$$

Jobs finishing times relative to the first job release time

$$X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \alpha_k^+(X_{i,j}) \cdot C_k$$

Worst-case response time

$$R_{i,j} = X_{i,j} + \sigma_i - (j-1) \times T_i$$

$$R_i = \max_{1 \le j \le N_i} \{R_{i,j}\}$$

Time from the arrival of the first job τ_i to the finish time the j^{th} jobs of τ_i

Earliest arrival time of the j^{th} jobs of τ_i

Equivalent to $\delta_i^-(j)$

g the terms

For all higher or equal-priority tasks than τ_i

Count the maximum number of jobs released by task τ_k in the interval $[0, L_i\ )$

$$L_i = \sum_{k=1}^{i} \alpha_k^+(L_i) \cdot C_k$$

$$N_i = \alpha_i^+(L_i)$$

Number of jobs of τ_i release in the level-i busy window

Jobs fi

For all higher or equal-priority tasks than τ_i

$$X_{i,j} = j \times C_i + \sum_{k=1}^{i-1} \alpha_k^+(X_{i,j}) \cdot C_k$$

Latest finishing time of the j^{th} jobs of τ_i

Count the maximum number of jobs released by task τ_k in the interval $[0, X_{i,j})$

nse time

$$R_{i,j} = X_{i,j} + \sigma_i - \delta_i^-(j)$$

$$R_i = \max_{1 \le j \le N_i} \{R_{i,j}\}$$

Worst-case response time of the j^{th} jobs of τ_i

Earliest arrival of the j^{th} jobs of τ_i

WCRT of τ_i

Agenda

- **Worst-case response-time analysis** under **preemptive Fixed priority** scheduling
  - Reminder
  - **Generalization (arrival curves)**

- **Best-case response time analysis** for preemptive **Fixed priority** scheduling

- **Worst-case schedulability analysis** for periodic or sporadic tasks under **preemptive EDF** scheduling

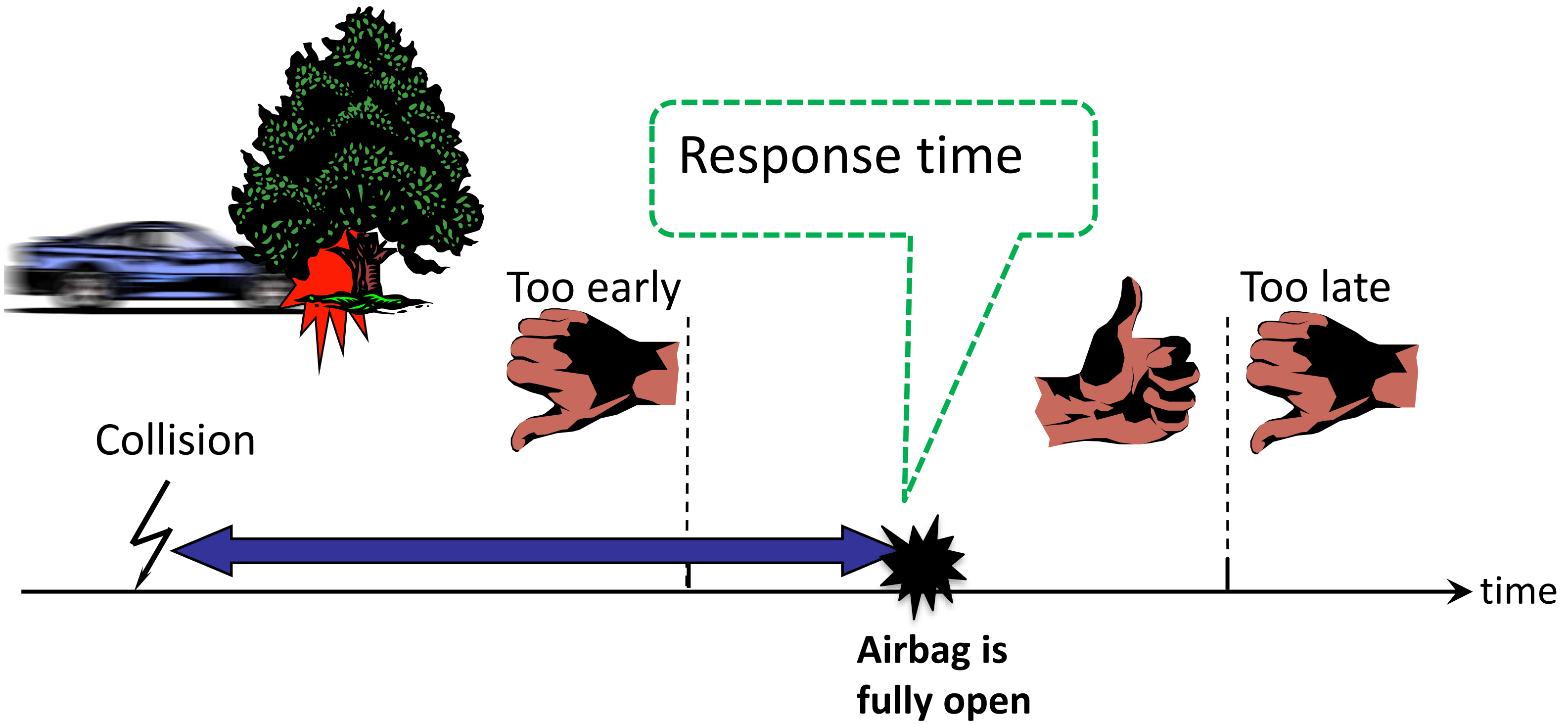
Motivation

Why would we be interested in the best-case?

Motivation 1

- Too early may be as bad as too late!

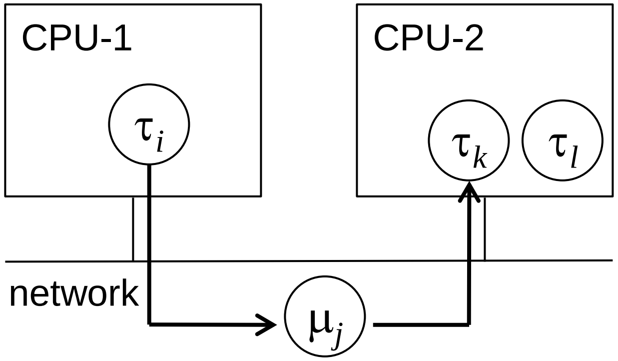
Example: airbag system



Motivation 2

Used for activation jitter of tasks in multicore/distributed systems

Example: distributed system



1. a strictly **periodic** *system* event activates a **task** $\boldsymbol{\tau_i}$
2. Task τ_i sends message μ_j on the bus/network

   ➔ **Response time jitter of $\boldsymbol{\tau_i}$ causes release jitter of $\boldsymbol{\mu_j}$**

3. Message μ_j triggers task τ_k:

   ➔ **Transmission time jitter of $\boldsymbol{\mu_j}$ causes release jitter of $\boldsymbol{\tau_k}$**

4. Task τ_k generates a *system* response

Release **jitter of $\boldsymbol{\tau_k}$ influences the *system* response** and **response times of task $\boldsymbol{\tau_l}$** with a lower priority

Best-case vs worst-case

- The best-case response time is the **dual of** the worst-case response time

Worst-case response time	**Best-case response time**
$R_i = \sup\{R_{i,j} \mid \forall j, 1 \leq j \leq \infty\}$	$BR_i = \inf\{R_{i,j} \mid \forall j, 1 \leq j \leq \infty\}$
Assume jobs execute for their **WCET**	Assume jobs execute for their **BCET**
Assume **latest** possible release	Assume **earliest** possible release
Assume **largest** possible interference by higher priority tasks	Assume **smallest** possible interference by higher priority tasks
Assume **largest** possible blocking	Assume **smallest** possible blocking

See lecture on shared resource access protocols

Best-case response time: general case

- The best-case response time is the **dual of** the worst-case response time

Worst-case response time

$$R_i = \sup\{R_{i,j} \mid \forall j, 1 \leq j \leq \infty\}$$

If $R_i \leq T_i$, R_i is the ***smallest*** positive solution of

$$X_i = C_i + \sum_{k=1}^{i-1} \alpha_k^{+}(X_i) \cdot C_k$$

$$R_i = X_i + \sigma_i$$

Best-case response time

$$BR_i = \inf\{R_{i,j} \mid \forall j, 1 \leq j \leq \infty\}$$

If $BR_i \leq T_i$, BR_i is the ***largest*** positive solution of

$$BR_i = BCET_i + \sum_{k=1}^{i-1} \alpha_k^{-}(BR_i) \cdot BCET_k$$

What does α_k^{-} look like for periodic or sporadic tasks with release jitter?

IRiS

Optimal instant for periodic tasks

(opposite of critical instant)

- An *optimal* instant of task τ_i is when τ_i "assumes" its BR_i.

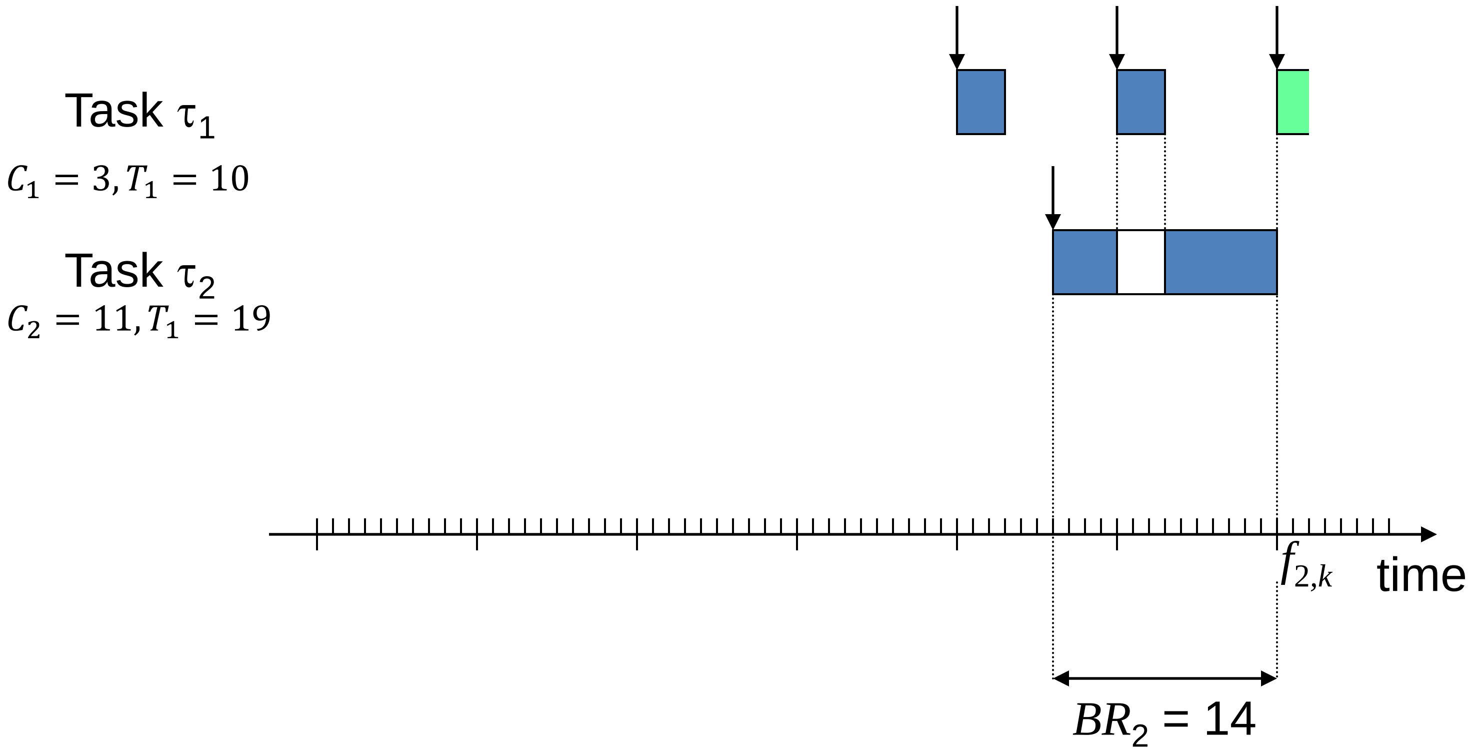
- An optimal instant requires to **minimize the number of higher priority jobs** executing during τ_i's response time
  - Job $\tau_{i,k}$ ***ends* simultaneously with the *release*** of all tasks with a higher priority, and $\tau_{i,k}$'s release time is equal to its start time.

  - ➔ Optimal instant different for each task !

35

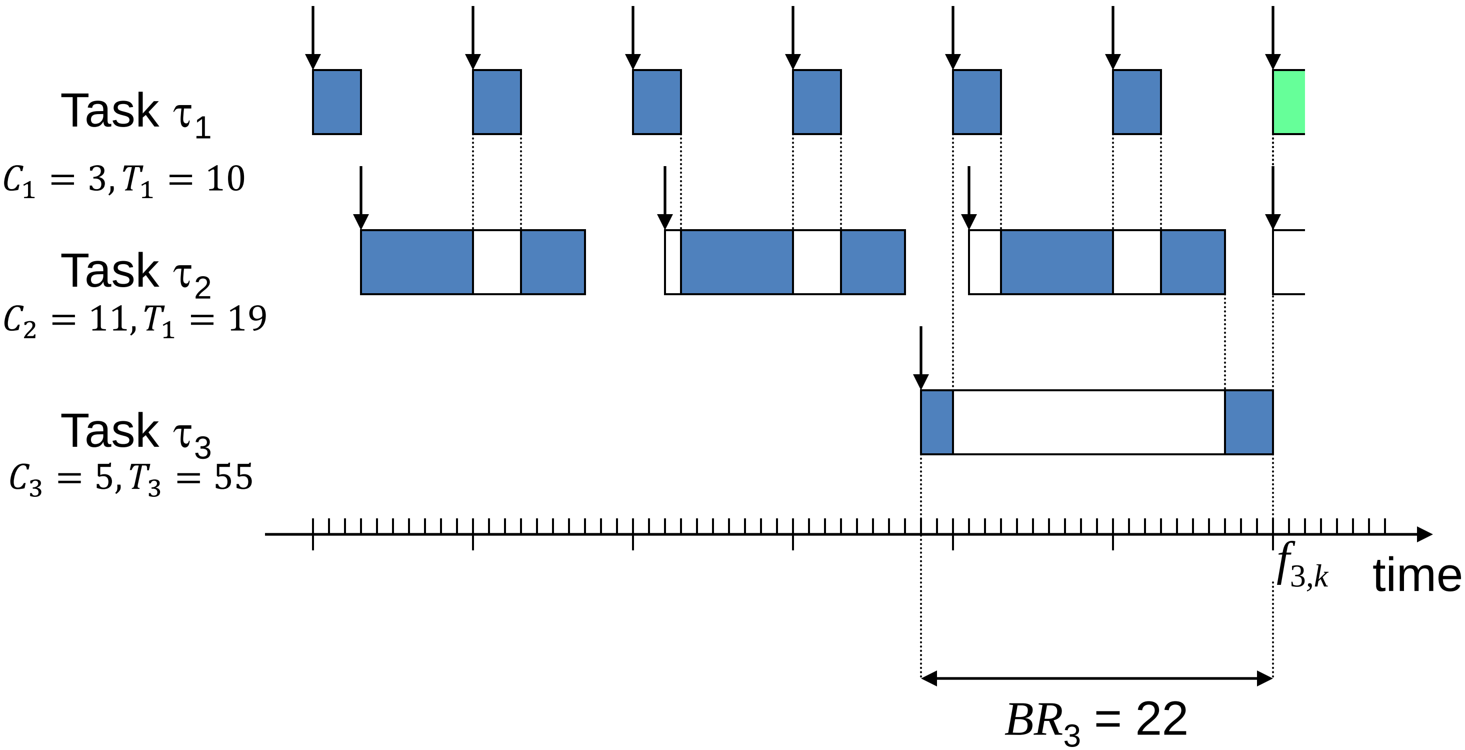
Optimal instant for periodic tasks - visualisation

- **Timeline:** optimal instant for τ_2



Optimal instant for periodic tasks - visualisation

- **Timeline:** optimal instant for τ_3



Optimal instant for periodic tasks - calculation

- BR_i is the ***largest*** positive solution of the recursive equation

$$x = BCET_i + \sum_{k<i} \left(\left\lceil \frac{x}{T_k} \right\rceil - 1 \right) BCET_k$$

Minimum number of jobs released by task τ_k in the interval [0, x)

- Solved with an iterative procedure:

Initialize to the WCRT because BCRT≤WCRT by definition

$$BR_i^{(0)} = R_i$$

$$BR_i^{(l+1)} = BCET_i + \sum_{k<i} \left(\left\lceil \frac{BR_i^{(l)}}{T_k} \right\rceil - 1 \right) BCET_k$$

Stop iterating when the same value is found for two successive iterations.

Techniques

- **Example for task τ_3:** $C_1 = 3, T_1 = 10$ $C_2 = 11, T_1 = 19$ $C_3 = 5, T_3 = 100$

39

Techniques

- **Example for task τ_3:** $C_1 = 3, T_1 = 10$ $C_2 = 11, T_1 = 19$ $C_3 = 5, T_3 = 100$
  - $BR_3^{(0)} = R_3 = 56$
  - $BR_3^{(1)} = C_3 + \Sigma_{j<3} (\lceil BR_3^{(0)}/T_j \rceil - 1)C_j =$
    $5 + (\lceil 56/10 \rceil - 1)\cdot 3 + (\lceil 56/19 \rceil - 1)\cdot 11 = 5 + 5\cdot 3 + 2\cdot 11 = 42$
  - $BR_3^{(2)} = 5 + (\lceil 42/10 \rceil - 1)\cdot 3 + (\lceil 42/19 \rceil - 1)\cdot 11 = 5 + 4\cdot 3 + 2\cdot 11 = 39$
  - $BR_3^{(3)} = 5 + (\lceil 39/10 \rceil - 1)\cdot 3 + (\lceil 39/19 \rceil - 1)\cdot 11 = 5 + 3\cdot 3 + 2\cdot 11 = 36$
  - $BR_3^{(4)} = 5 + (\lceil 36/10 \rceil - 1)\cdot 3 + (\lceil 36/19 \rceil - 1)\cdot 11 = 5 + 3\cdot 3 + 1\cdot 11 = 25$
  - $BR_3^{(5)} = 5 + (\lceil 25/10 \rceil - 1)\cdot 3 + (\lceil 25/19 \rceil - 1)\cdot 11 = 5 + 2\cdot 3 + 1\cdot 11 = 22$
  - $BR_3^{(6)} = 5 + (\lceil 22/10 \rceil - 1)\cdot 3 + (\lceil 22/19 \rceil - 1)\cdot 11 = 5 + 2\cdot 3 + 1\cdot 11 = 22$
  - Because $BR_3^{(5)} = BR_3^{(6)} = 22$, $BR_3 = 22$.

40

Jitter analysis

- Types of jitter:
  - *Activation or release jitter* σ_i: variation in release times (e.g. output of one task triggers a next task or delay induced by OS to treat an event and make the task ready)
  - *Response jitter*: variations in response times

- Response jitter RJ_i of a task τ_i:

$$RJ_i \stackrel{\text{def}}{=} \sup_{\phi,k,l}(R_{i,k}(\phi) - R_{i,l}(\phi))$$

- A bound on response jitter: $RJ_i \leq R_i - BR_i$

Note: this is only an upper bound (hence the $\leq$) because R_i and BR_i may be obtained for different phasings