Embedded Systems

Exercise 3: Scheduling Periodic and Mixed Task Sets

5. / 12. April 2017

Roman Trüb, roman.trueb@tik.ee.ethz.ch
Overview

- Clicker Quiz
- Summary of the Lecture
- Start solving the Exercise
- Discussion of the Solution
Clicker Quiz
Clicker Quiz – Question 1

Is the given set of periodic tasks schedulable with EDF?

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_i$</td>
<td>6</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>$T_i$</td>
<td>9</td>
<td>15</td>
<td>5</td>
</tr>
<tr>
<td>$D_i$</td>
<td>9</td>
<td>15</td>
<td>5</td>
</tr>
</tbody>
</table>

(a) Yes.

(b) No.

(c) No statement can be made using the sufficient and necessary test.

$$U = \sum_{i=1}^{n} \frac{C_i}{T_i} \leq 1 \iff \text{Schedulable with EDF}$$

$$\frac{6}{9} + \frac{5}{15} + \frac{1}{5} = 1.2 \ngeq 1 \quad \text{Test failed}$$

$\Rightarrow \quad \text{Task set is not schedulable with EDF}$
Clicker Quiz – Question 2

The following set of periodic tasks is scheduled using RM. Select all correct statements.

<table>
<thead>
<tr>
<th>Task</th>
<th>( \tau_1 )</th>
<th>( \tau_2 )</th>
<th>( \tau_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_i )</td>
<td>4</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>( T_i )</td>
<td>11</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>( D_i )</td>
<td>11</td>
<td>9</td>
<td>6</td>
</tr>
</tbody>
</table>

(a) The task set passes the sufficient (but not necessary) schedulability test.
(b) The task set fails the sufficient (but not necessary) schedulability test.
(c) According to the sufficient (but not necessary) test, the task set is not schedulable with RM.

\[
\sum_{i=1}^{n} \frac{C_i}{T_i} \leq n\left(\frac{1}{2^n} - 1\right) \implies \text{Schedulable with RM}
\]

\[
\frac{4}{11} + \frac{1}{9} + \frac{2}{6} = 0.86672 \leq 0.7798 \implies \text{Test failed}
\]

With this test result, no statement can be made since schedulability test condition is not necessary.
\( \Rightarrow \) Task set could still be schedulable with RM.
Clicker Quiz – Question 3

The given set of periodic tasks is executed together with a Polling Server with period $T_s = 12$.

<table>
<thead>
<tr>
<th>$C_i$</th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>$T_i$</td>
<td>14</td>
<td>11</td>
<td>7</td>
</tr>
<tr>
<td>$D_i$</td>
<td>14</td>
<td>11</td>
<td>7</td>
</tr>
</tbody>
</table>

Considering the sufficient (but not necessary) schedulability test, what is the maximum (integer) execution time $C_s$ of the Polling Server such that the set of periodic tasks together with the Polling Server is schedulable with RM?

(a) $C_s = 1$
(b) $C_s = 2$
(c) $C_s = 3$
(d) $C_s = 4$

\[
\frac{C_s}{T_s} \leq (n + 1)(2^{\frac{1}{n+1}} - 1) - \sum_{i=1}^{n} \frac{C_i}{T_i}
\]

\[
C_s \leq 12(0.7568 - 0.5584) = 2.3808
\]

$C_s = \lfloor 2.3808 \rfloor = 2$
Clicker Quiz – Question 4

The following two periodic tasks are executed with EDF.

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_i$</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$T_i$</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$D_i$</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

We would like to schedule an aperiodic job $\tau_a$ with execution time $C_a = 1$ and release time $r_a = 0$ using a Total Bandwidth Server (TBS) with utilization $U_s = 0.25$. What is the smallest guaranteed absolute deadline $d_a$ of $\tau_a$ such that all jobs meet their deadlines?

(a) $d_a = 2$
(b) $d_a = 3$
(c) $d_a = 4$
(d) $d_a = 5$

\[
d_a = \max(r_a, d_{k-1}) + \frac{C_a}{U_s}
\]

\[
d_a = \max(0, 0) + \frac{1}{0.25} = 4
\]
Lecture Summary
## Scheduling Algorithms

<table>
<thead>
<tr>
<th></th>
<th>Periodic with $D = T$</th>
<th>Periodic with $D &lt; T$</th>
<th>Mixed Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Priority</strong></td>
<td>RM</td>
<td>DM</td>
<td>Polling Server</td>
</tr>
<tr>
<td><strong>Dynamic Priority</strong></td>
<td>EDF</td>
<td>EDF</td>
<td>Total Bandwidth Server</td>
</tr>
</tbody>
</table>
### Scheduling Algorithms

<table>
<thead>
<tr>
<th></th>
<th>Periodic with $D = T$</th>
<th>Periodic with $D &lt; T$</th>
<th>Mixed Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Priority</strong></td>
<td><strong>RM</strong></td>
<td><strong>DM</strong></td>
<td>Polling Server</td>
</tr>
<tr>
<td><strong>Dynamic Priority</strong></td>
<td><strong>EDF</strong></td>
<td><strong>EDF</strong></td>
<td>Total Bandwidth Server</td>
</tr>
</tbody>
</table>
Rate Monotonic (RM)

Static priority assignment:
Tasks with shorter period get higher priority

T1

T2

T3

\( t \)
RM – Schedulability Test

\[ \sum_{i=1}^{n} \frac{C_i}{T_i} \leq n \left(2^{\frac{1}{n}} - 1\right) \]
Sufficient vs. Necessary

\[ X \Rightarrow Y \quad \text{X is sufficient for Y} \]

\[ \text{not}(X) \Rightarrow \text{not}(Y) \]

\[ \iff \]

\[ X \iff Y \quad \text{X is necessary for Y} \]

\[ X \iff Y \quad \text{X is sufficient and necessary for Y} \]

Example:

\[ \sum_{i=1}^{n} \frac{C_i}{T_i} \leq n(2^{\frac{1}{n}} - 1) \iff \text{Task set is schedulable RM} \]

\[ \text{Sufficient (but not necessary)} \]
## Scheduling Algorithms

<table>
<thead>
<tr>
<th></th>
<th>Periodic with $D = T$</th>
<th>Periodic with $D &lt; T$</th>
<th>Mixed Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static Priority</td>
<td>RM</td>
<td>DM</td>
<td>Polling Server</td>
</tr>
<tr>
<td>Dynamic Priority</td>
<td>EDF</td>
<td>EDF</td>
<td>Total Bandwidth Server</td>
</tr>
</tbody>
</table>
Deadline Monotonic (DM)

Static priority assignment:
Tasks with smaller relative deadline get higher priority
DM – Schedulability Test

\[ \sum_{i=1}^{n} \frac{C_i}{D_i} \leq n \left( 2^{\frac{1}{n}} - 1 \right) \]
RM & DM – Schedulability Test

Algorithm: DM_guarantee (Γ)
{    for (each τᵢ ∈ Γ){
        I = 0;
        do {
            R = I + Cᵢ;
            if (R > Dᵢ) return(UNSCHEDULABLE);
            I = Σⱼ=1,...,(i-1)\[R/Tⱼ\] Cⱼ;
        } while (I + Cᵢ > R);
    }
    return(SCHEDULABLE);
}

Longest Response Time $R_i$
(computed iteratively)

\[
I_i = \sum_{j=1}^{i-1} \left[ \frac{R_i}{T_j} \right] C_j + C_i
\]
# Scheduling Algorithms

<table>
<thead>
<tr>
<th></th>
<th>Periodic with $D = T$</th>
<th>Periodic with $D &lt; T$</th>
<th>Mixed Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Priority</strong></td>
<td>RM</td>
<td>DM</td>
<td>Polling Server</td>
</tr>
<tr>
<td><strong>Dynamic Priority</strong></td>
<td>EDF</td>
<td>EDF</td>
<td>Total Bandwidth Server</td>
</tr>
</tbody>
</table>

Roman Trüb | 5.4.2017 / 12.4.2017 | 18
Earliest Deadline First (EDF)

Dynamic Priority Assignment:
Always execute the task with the currently closest deadline

T1

T2

T3
EDF – Schedulability Tests

\[ D_i = T_i \]  
\[ D_i < T_i \]

**Necessary & Sufficient**

\[ \sum_{i=1}^{n} \frac{C_i}{T_i} \leq 1 \]

**Sufficient (but not necessary)**

\[ \sum_{i=1}^{n} \frac{C_i}{D_i} \leq 1 \]

**Utilization:**

\[ U = \sum_{i=1}^{n} \frac{C_i}{T_i} \]
Mixed Task Scheduling

- Mixed (or hybrid) task set contains both periodic and aperiodic tasks

- Basic idea for scheduling:
  - Schedule the periodic tasks as usual
  - Serve the aperiodic tasks via a "server" that behaves like a periodic task but serves the aperiodic tasks
## Scheduling Algorithms

<table>
<thead>
<tr>
<th></th>
<th>Periodic with $D = T$</th>
<th>Periodic with $D &lt; T$</th>
<th>Mixed Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Priority</strong></td>
<td>RM</td>
<td>DM</td>
<td>Polling Server</td>
</tr>
<tr>
<td><strong>Dynamic Priority</strong></td>
<td>EDF</td>
<td>EDF</td>
<td>Total Bandwidth Server</td>
</tr>
</tbody>
</table>
RM – Polling Server

- **Idea:** Introduce an artificial periodic task \((C_s, T_s)\) which serves the aperiodic requests

**Schedulability test for mixed task set:**

\[
\frac{C_s}{T_s} + \sum_{i=1}^{n} \frac{C_i}{T_i} \leq (n + 1)(2^{\frac{1}{n+1}} - 1)
\]

**Aperiodic guarantee:**

\[
(1 + \left\lceil \frac{C_a}{C_s} \right\rceil)T_s \leq D_a
\]

**Assumption:** aperiodic task finishes before new aperiodic request arrives

**Sufficient (but not necessary)**
### Scheduling Algorithms

<table>
<thead>
<tr>
<th></th>
<th>Periodic with $D = T$</th>
<th>Periodic with $D &lt; T$</th>
<th>Mixed Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Priority</strong></td>
<td>RM</td>
<td>DM</td>
<td>Polling Server</td>
</tr>
<tr>
<td><strong>Dynamic Priority</strong></td>
<td>EDF</td>
<td>EDF</td>
<td>Total Bandwidth Server</td>
</tr>
</tbody>
</table>
EDF – Total Bandwidth Server

- **Idea:** For every aperiodic request a deadline which is based on the server’s parameters \((C_s, T_s)\) is assigned. The aperiodic requests are then scheduled with EDF as any other periodic instance.

**Assignment of the deadlines to the aperiodic requests:**

\[
d_k = \max(r_k, d_{k-1}) + \frac{C_k}{U_s}
\]

\[
U_s = \frac{C_s}{T_s}
\]

**NOTE:** recursive computation!
Aperiodic requests are assumed to be ordered by increasing release time \(r_i\).
EDF – Total Bandwidth Server

Schedulability test for mixed task set:

\[ U_p + U_s \leq 1 \iff \]

Set of periodic tasks with utilization \( U_p \) and a Total Bandwidth Server with utilization \( U_s \) is schedulable with EDF

Utilization of Total Bandwidth Server:

\[ U_s = \frac{C_s}{T_s} \]

Utilization of the periodic tasks:

\[ U_p = \sum_{i=1}^{n} \frac{C_i}{T_i} \]
Exercise 3
Task 1: Earlyest Deadline First (EDF) and Total Bandwidth Server (TBS)

Consider the following set of periodic tasks:

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_i$</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$T_i$</td>
<td>3</td>
<td>5</td>
<td>13</td>
</tr>
</tbody>
</table>

A Total Bandwidth Server (TBS) executes along with the periodic tasks above.

1. What can be the maximum value of $U_s$ such that the whole set (i.e., periodic tasks and the TBS) is schedulable with EDF?

2. Now assume $U_s = 0.25$. Construct the EDF schedule (in Figure 1) in the case in which three aperiodic requests $J_4(r_4 = 0, C_4 = 2)$, $J_5(r_5 = 15, C_5 = 1)$ and $J_6(r_6 = 10, C_6 = 1)$ are served by TBS. Assume that the arrival time of the first instance/job of each periodic task is 0.
Task 2: Schedulability Test for Fixed Priorities - Rate Monotonic (RM)

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_i$</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>$T_i$</td>
<td>3</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

1. Test if the given task-set is schedulable under RM, using the sufficient test.
2. Test if the given task-set is schedulable under RM, using the necessary test.
3. Assume that the first job of each task arrives at time 0. Construct the schedule for the interval [0, 20] and illustrate it graphically. In case they exist, identify deadline misses.
### Task 3: Scheduling with Polling Server

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_i$</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$D_i$</td>
<td>6</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>$T_i$</td>
<td>6</td>
<td>8</td>
<td>16</td>
</tr>
</tbody>
</table>

In addition to the above periodic tasks, we have an aperiodic job $J_a$ with computation time $C_a = 1$, and relative deadline $D_a$. The scheduling policy is RM. The aperiodic job is scheduled through a Polling Server (PS).

1. Let the period and computing time of the polling server be $T_s = 25$ and $C_s = 1$, respectively. Compute the aperiodic guarantee available to $J_a$, i.e., compute the minimum relative deadline of $J_a$ which is guaranteed not to be missed.
2. Using the sufficient test of RM, test if the polling server of 1.) is schedulable along with the periodic task-set?
Exercise 3 - Solution
Task 1: Earliest Deadline First (EDF) and Total Bandwidth Server (TBS)

Consider the following set of periodic tasks:

<table>
<thead>
<tr>
<th></th>
<th>( \tau_1 )</th>
<th>( \tau_2 )</th>
<th>( \tau_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_i )</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>( T_i )</td>
<td>3</td>
<td>5</td>
<td>13</td>
</tr>
</tbody>
</table>

A Total Bandwidth Server (TBS) executes along with the periodic tasks above.

1. What can be the maximum value of \( U_s \) such that the whole set (i.e., periodic tasks and the TBS) is schedulable with EDF?

2. Now assume \( U_s = 0.25 \). Construct the EDF schedule (in Figure 1) in the case in which three aperiodic requests \( J_4(r_4 = 0, C_4 = 2), J_5(r_5 = 15, C_5 = 1) \) and \( J_6(r_6 = 10, C_6 = 1) \) are served by TBS. Assume that the arrival time of the first instance/job of each periodic task is 0.
Task 1 – Solution

1. Maximum utilization of the server:
   \[ U_{s,\text{max}} = 1 - U_p = 1 - \left( \frac{1}{3} + \frac{1}{5} + \frac{2}{13} \right) \approx 0.3128 \]

2. Order tasks by increasing release time: J₄, J₆, J₅
   
   Then, calculate new deadlines recursively:
   \[ d_k = \max(r_k, d_{k-1}) + \frac{C_k}{U_s} \]
   
   \[ d_4 = \max(r_4, d_0) + 2/0.25 = 0 + 8 = 8 \]
   \[ d_6 = \max(r_6, d_4) + 1/0.25 = 10 + 4 = 14 \]
   \[ d_5 = \max(r_5, d_6) + 1/0.25 = 15 + 4 = 19 \]
2. EDF schedule with aperiodic tasks:
Task 2: Schedulability Test for Fixed Priorities - Rate Monotonic (RM)

<table>
<thead>
<tr>
<th></th>
<th>$T_1$</th>
<th>$T_2$</th>
<th>$T_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_i$</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>$T_i$</td>
<td>3</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

1. Test if the given task-set is schedulable under RM, using the sufficient test.
2. Test if the given task-set is schedulable under RM, using the necessary test.
3. Assume that the first job of each task arrives at time 0. Construct the schedule for the interval [0, 20] and illustrate it graphically. In case they exist, identify deadline misses.
Task 2 – Solution

1. Sufficient test:

\[
\sum_{i=1}^{n} \frac{C_i}{T_i} \leq n\left(2^{\frac{1}{n}} - 1\right)
\]

\[
\frac{1}{3} + \frac{3}{8} + \frac{2}{9} = 0.93 \leq 3\left(2^{\frac{1}{3}} - 1\right) = 0.78
\]

Test failed! Since it is not necessary, we still don’t know if the task set is schedulable with RM or not.
Task 2 – Solution

2. Necessary test: Iterative algorithm
   1) Order tasks by their priority (already ok in this case)
   2) Start with task which has lowest priority:

\[ R_i = \sum_{k=1}^{i-1} \left\lfloor \frac{R_i}{T_k} \right\rfloor C_k + C_i \]

\[ \tau_3: \]
\[ R_3^0 = C_3 = 2 \quad I_3^0 = \left\lfloor \frac{1}{3} \right\rfloor 1 + \left\lfloor \frac{2}{3} \right\rfloor 3 = 1 + 3 = 4 \quad 4 + 2 \neq 2 \]
\[ R_3^1 = 4 + 2 = 6 \quad I_3^1 = \left\lfloor \frac{1}{3} \right\rfloor 1 + \left\lfloor \frac{2}{3} \right\rfloor 3 = 2 + 3 = 5 \quad 5 + 2 \neq 6 \]
\[ R_3^2 = 5 + 2 = 7 \quad I_3^2 = \left\lfloor \frac{1}{3} \right\rfloor 1 + \left\lfloor \frac{2}{3} \right\rfloor 3 = 3 + 3 = 6 \quad 6 + 2 \neq 7 \]
\[ R_3^3 = 6 + 2 = 8 \quad I_3^3 = \left\lfloor \frac{1}{3} \right\rfloor 1 + \left\lfloor \frac{2}{3} \right\rfloor 3 = 3 + 3 = 6 \quad 6 + 2 = 8 \ldots \text{OK} \]

(since \( R_3 = 8 \leq T_3 = 9 \))

\[ \tau_2: \]
\[ R_2^0 = C_2 = 3 \quad I_2^0 = \left\lfloor \frac{3}{2} \right\rfloor 1 = 1 \quad 1 + 3 \neq 3 \]
\[ R_2^1 = 1 + 3 = 4 \quad I_2^1 = \left\lfloor \frac{3}{2} \right\rfloor 1 = 2 \quad 2 + 3 \neq 4 \]
\[ R_2^2 = 2 + 3 = 5 \quad I_2^2 = \left\lfloor \frac{3}{2} \right\rfloor 1 = 2 \quad 2 + 3 = 5 \ldots \text{OK} \] (since \( R_2 = 5 \leq T_2 = 8 \))

\[ \tau_1: \]
\[ R_1^0 = C_1 = 1 \quad I_1^0 = 0 \quad 0 + 1 = 1 \ldots \text{OK} \] (since \( R_1 = 1 \leq T_1 = 1 \))

Necessary test succeeds \( \Rightarrow \) task set is schedulable with RM
Task 2 – Solution

3. RM schedule
Task 3: Scheduling with Polling Server

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_i$</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$D_i$</td>
<td>6</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>$T_i$</td>
<td>6</td>
<td>8</td>
<td>16</td>
</tr>
</tbody>
</table>

In addition to the above periodic tasks, we have an aperiodic job $J_a$ with computation time $C_a = 1$, and relative deadline $D_a$. The scheduling policy is RM. The aperiodic job is scheduled through a Polling Server (PS).

1. Let the period and computing time of the polling server be $T_s = 25$ and $C_s = 1$, respectively. Compute the aperiodic guarantee available to $J_a$, i.e., compute the minimum relative deadline of $J_a$ which is guaranteed not to be missed.

2. Using the sufficient test of RM, test if the polling server of 1.) is schedulable along with the periodic task-set?
Task 3 – Solution

1. Aperiodic guarantee:

\[(1 + \left[ \frac{C_a}{C_s} \right])T_s \leq D_a\]

Substituting values & considering minimum \( => D_a = 50\)

2. Sufficient schedulability test for RM Polling Server:

\[
\frac{C_s}{T_s} + \sum_{i=1}^{n} \frac{C_i}{T_i} \leq (n + 1)(2^{\frac{1}{n+1}} - 1)
\]

\[0.748 \leq 0.76 \quad \text{Test succeeded!}\]

\( => \text{Task set is schedulable with RM}\)