Advanced Real-Time Systems

Lecture 2/6

Enrico Bini

October 26, 2012

Outline

Motivation

- The drawback of the response-time test is its "black-box" nature: all task parameters are specified and we get a yes/no answer.
- In reality, it is often more desirable to have a margin on the system parameters that guarantee schedulability: *sensitivity analysis*.

Scheduling points tests

An alternate exact test is the following one.

Theorem (Lehoczky et al. [?])

A constrained deadline (with $D_i \leq T_i$) task set is schedulable by FP if and only if

$$\forall i \in \mathcal{N},\ \exists t \in [0, D_i], \quad C_i + I_i(t) \leq t$$

Interesting, but not so practical (how do we check if it exists a point in a real interval?)

First simple reduction

If we remember the expression of $I_i(t)$ we realize that the previous condition is equivalent to the following one, which can be better managed

$$\forall i \in \mathcal{N},\ \exists t \in \mathcal{S}_i, \quad C_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j \leq t$$

with

$$\mathcal{S}_i = \{kT_j : k \in \mathbb{N},\ 0 < kT_j < D_i,\ j < i\} \cup \{D_i\}$$

This is the set of *scheduling points*.

However the points in $\mathcal{S}_i$ can still be many and period dependent, especially when the periods of the higher priority tasks are significantly smaller than T_i.

Second sophisticated reduction

- Can we remove points from $\mathcal{S}_i$ to reduce the complexity?
- By removing arbitrarily points we may lose necessity.

Theorem

A constrained deadline (with $D_i \leq T_i$) task set is schedulable by FP if and only if

$$\forall i \in \mathcal{N},\ \exists t \in \mathcal{P}_{i-1}(D_i), \quad C_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j \leq t \qquad (1)$$

with $\mathcal{P}_j(t)$ being a set recursively defined as

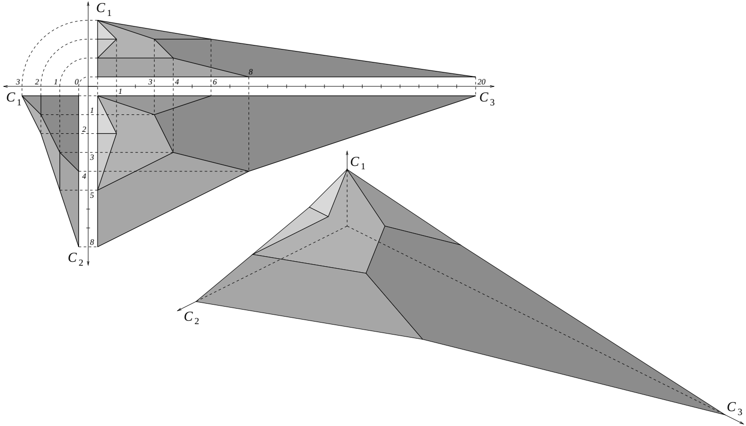
$$\begin{cases} \mathcal{P}_0(t) = \{t\} \\ \mathcal{P}_j(t) = \mathcal{P}_{j-1}\left(\left\lfloor \frac{t}{T_j} \right\rfloor T_j\right) \cup \mathcal{P}_{j-1}(t). \end{cases} \qquad (2)$$

Show how $\mathcal{P}_{i-1}(D_i)$ is computed, for two tasks and U-plane.

Visualization of the test

$$\forall i \in \mathcal{N},\ \exists t \in \mathcal{P}_{i-1}(D_i), \quad C_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j \leq t \qquad (3)$$

If $T_1 = 3, T_2 = 8, T_3 = 20$, and $D_i = T_i$ the schedulable C_i are



Outline

Min schedulable speed

- If the processor runs at speed r, then all computation times become C_i/r
- From the scheduling point condition it is not difficult to find the smallest speed that guarantee FP-schedulability

$$\forall i \in \mathcal{N},\ \exists t \in \mathcal{P}_{i-1}(D_i), \quad \frac{C_i}{r} + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil \frac{C_j}{r} \leq t$$

$$\forall i \in \mathcal{N},\ \exists t \in \mathcal{P}_{i-1}(D_i), \quad r \geq \frac{C_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j}{t}$$

$$r \geq \max_{i \in \mathcal{N}} \min_{t \in \mathcal{P}_{i-1}(D_i)} \frac{C_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j}{t}$$

- (there is no direct way to find it using the response time)
- Example of calculation for two tasks in the plot.

Max schedulable C_k

What is the maximum schedulable C_k?

- all tasks τ_i with $i < k$ are unaffected by C_k
- to ensure the schedulability of τ_k it must be

$$C_k \leq \max_{t \in \mathcal{P}_{k-1}(D_k)} (t - \sum_{j=1}^{k-1} \left\lceil \frac{t}{T_j} \right\rceil C_j)$$

  The RHS is the amount of *level-*$(k-1)$ *idle time* in $[0, D_k]$

- to ensure the schedulability of all tasks with lower priority $i > k$ it must be

$$C_k \leq \min_{i=k+1,\ldots,n} \max_{t \in \mathcal{P}_{i-1}(D_i)} \frac{t - (C_i + \sum_{\substack{j=1 \\ j \neq k}}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j)}{\left\lceil \frac{t}{T_k} \right\rceil}$$

- hence C_k^{max} is the minimum of the two RHS

Min schedulable D_k

Since R_k is independent of the deadline D_k, the minimum schedulable deadline is just $D_k = R_k$.

Min schedulable T_k

- Task periods appear in the ceiling operator. Not trivial to extract them
- More sophisticated technique needed.

Let $T_k^{(i)}$ denote the minimum period of τ_k such that τ_i is schedulable.

- $i < k$, meaningless
- then
$$T_k^{\min} = \max_{i \geq k} T_k^{(i)}, \tag{4}$$
- The schedulability of τ_k is not affected by T_k
  - if constrained deadline ($D_k \leq T_k$) then $T_k^{(k)} = D_k$;
  - if implicit deadline ($D_k = T_k$) then $T_k^{(k)} = R_k$.

Computing $T_k^{(i)}$

- It requires some effort

Definition

Given the subset of tasks $\mathcal{M} \subseteq \mathcal{N}$, we define *level-$\mathcal{M}$ idle time* in $[0, D]$, denoted by $Y(\mathcal{M}, D)$, the amount of time in $[0, D]$ in which no task in $\mathcal{M}$ is executing, under the worst-case scenario for activations.

Examples:

- $Y(\varnothing, D) = D$
- $Y(\{\tau_1\}, T_1) = T_1 - C_1$

To evaluate $T_k^{(i)}$ we need $Y(\{1, \ldots, i\} \setminus \{k\}, D_i)$, because it is the time that can be consumed by τ_k, keeping τ_i schedulable.

Computing $T_k^{(i)}$

- How do we compute $Y(\mathcal{M}, D)$?

$$Y(\mathcal{M}, D) = \max_{t \in \mathcal{P}_{|\mathcal{M}|}(D)} \left\{ t - \sum_{j \in \mathcal{M}} \left\lceil \frac{t}{T_j} \right\rceil C_j \right\}$$

- It is scaring! Let's check.
- Then the maximum number $n_k^{(i)}$ of τ_k (associated to the minimum period $T_k^{(i)}$) jobs which can preserve the schedulability of τ_i then is

$$n_k^{(i)} = \left\lfloor \frac{Y(\{1, \ldots, i\} \setminus \{k\}, D_i)}{C_k} \right\rfloor$$

Computing $T_k^{(i)}$

- Given the maximum number of jobs $n_k^{(i)}$, what is the minimum period $T_k^{(i)}$ which preserves the schedulability of τ_i?
- $T_k^{(i)}$ is such that τ_k has $n_k^{(i)}$ jobs interfering on τ_i, and by increasing $T_k^{(i)}$ by any small amount the number of interfering jobs would increase. Hence

$$T_k^{(i)} = \frac{R_i}{n_k^{(i)}}$$

with R_i response time of τ_i with $n_k^{(i)}$ interfering jobs by τ_k

$$R_i = C_i + n_k^{(i)} C_k + \sum_{\substack{j=1 \\ j \neq k}}^{i-1} \left\lceil \frac{R_i}{T_j} \right\rceil C_j$$

Computing $T_k^{(i)}$

Summarizing

1. For all $i > k$ we compute

$$n_k^{(i)} = \left\lfloor \frac{Y(\{1,\ldots,i\} \setminus \{k\}, D_i)}{C_k} \right\rfloor$$

$$R_i = C_i + n_k^{(i)} C_k + \sum_{\substack{j=1 \\ j \neq k}}^{i-1} \left\lceil \frac{R_i}{T_j} \right\rceil C_j$$

$$T_k^{(i)} = \frac{R_i}{n_k^{(i)}}$$

$$T_k^{\text{min}} = \max\{D_k, T_k^{(k+1)}, T_k^{(k+2)}, \ldots, T_k^{(n)}\} \text{ if constr dl}$$

$$T_k^{\text{min}} = \max\{R_k, T_k^{(k+1)}, T_k^{(k+2)}, \ldots, T_k^{(n)}\} \text{ if impl dl}$$

Using sufficient tests for sensitivity

If some pessimism is acceptable, we can gain in simplicity by using sufficient tests for the sensitivity

- Using LL bound $U_{\mathsf{LL}} = n(\sqrt[n]{2} - 1)$:
  - Min speed: $r^{\mathsf{min}} = \frac{\sum_i U_i}{U_{\mathsf{LL}}}$
  - Max C_k: $C_k^{\mathsf{max}} = T_k(U_{\mathsf{LL}} - \sum_{j \neq k} U_j)$
  - Min T_k: $T_k^{\mathsf{min}} = \frac{C_k}{U_{\mathsf{LL}} - \sum_{j \neq k} U_j}$

Outline

Sufficient tests

Let us have the following task set (write them down on the whiteboard)

$T_i = D_i$	C_i	U_i
3	1	$1/3 \approx 0.333$
8	2	$2/8 = 0.25$
20	5	$5/20 = 0.25$

LL test ($\sqrt[3]{2} = 1.2599$):

Sufficient tests

Let us have the following task set (write them down on the whiteboard)

$T_i = D_i$	C_i	U_i
3	1	$1/3 \approx 0.333$
8	2	$2/8 = 0.25$
20	5	$5/20 = 0.25$

LL test ($\sqrt[3]{2} = 1.2599$):

$$0.8333 \leq 3(\sqrt[3]{2} - 1) = 0.7798$$

HB test:

Sufficient tests

Let us have the following task set (write them down on the whiteboard)

$T_i = D_i$	C_i	U_i
3	1	$1/3 \approx 0.333$
8	2	$2/8 = 0.25$
20	5	$5/20 = 0.25$

LL test ($\sqrt[3]{2} = 1.2599$):

$$0.8333 \leq 3(\sqrt[3]{2} - 1) = 0.7798$$

HB test:

$$(1 + U_1)(1 + U_2)(1 + U_3) = 2.083 \leq 2$$

Response time

$$R_1^{(0)} = 1$$
$$R_2^{(0)} = 2$$
$$R_2^{(1)} = 2 + 1 = 3$$
$$R_3^{(0)} = 5$$
$$R_3^{(1)} = 5 + 2 \times 1 + 2 = 9$$
$$R_3^{(2)} = 5 + 3 + 4 = 12$$
$$R_3^{(3)} = 5 + 4 + 4 = 13$$
$$R_3^{(4)} = 5 + 5 + 4 = 14$$

Since $\forall i = 1, 2, 3$ we have $R_i \le D_i$, then the task set is FP-schdulable.

Response time up bound

$$\overline{R}_i = \frac{C_i + \sum_{j=1}^{i-1} C_j(1 - U_j)}{1 - \sum_{j=1}^{i-1} U_j}$$

$$\overline{R}_1 = \frac{C_1 + 0}{1} = C_1 = 1 \quad (= R_1)$$

$$\overline{R}_2 = \frac{C_2 + C_1(1 - U_1)}{1 - U_1} = \frac{2 + \frac{2}{3}}{\frac{2}{3}} = 4 \quad (R_2 = 3)$$

$$\overline{R}_3 = \frac{5 + \frac{2}{3} + \frac{3}{2}}{\frac{5}{12}} = 17.2 \quad (R_3 = 14)$$

Since $\forall i = 1, 2, 3$ we have $\overline{R}_i \leq D_i$, then the task set is FP-schdulable (even using the response time upper bound).

Sched points test

$$\forall i \in \mathcal{N},\ \exists t \in \mathcal{P}_{i-1}(D_i), \quad C_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j \leq t \tag{5}$$

$i = 1$

$\mathcal{P}_0(T_1) = \{T_1\} = \{3\}$

$C_1 \leq T_1 \quad 1 \leq 3$

$i = 2$

$\mathcal{P}_1(T_2) = \{6, 8\}$

$2C_1 + C_2 \leq 6$ or $3C_1 + C_2 \leq 8$

$i = 3$

$\mathcal{P}_2(T_3) = \{15, 16, 18, 20\}$

$5C_1 + 2C_2 + C_3 \leq 15$ or $6C_1 + 2C_2 + C_3 \leq 16$ or

$6C_1 + 3C_2 + C_3 \leq 18$ or $7C_1 + 3C_2 + C_3 \leq 20$

Min schedulable speed

Should be ≤ 1 because already schedulable

$$r^{\text{min}} = \max_{i \in \mathcal{N}} \min_{t \in \mathcal{P}_{i-1}(D_i)} \frac{C_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil C_j}{t}$$

$$i = 1,\ \mathcal{P}_0(T_1) = \{3\}$$

$$\frac{C_1}{T_1} = \frac{1}{3}$$

$$i = 2,\ \mathcal{P}_1(T_2) = \{6, 8\}$$

$$\frac{2C_1 + C_2}{6} = \frac{4}{6} \approx 0.667,\ \frac{3C_1 + C_2}{8} = \frac{5}{8} = 0.625$$

$$i = 3,\ \mathcal{P}_2(T_3) = \{15, 16, 18, 20\}$$

$$\frac{14}{15} \approx 0.933,\ \frac{15}{16} \approx 0.938,\ \frac{17}{18} \approx 0.944,\ \frac{18}{20} = 0.9$$

$$r^{\text{min}} = \max\{0.333, 0.625, 0.9\} = 0.9$$

$$r^{\text{min}}(\text{LL}) = \sum_i U_i / U_{\text{LL}} = 1.069$$

Max schedulable C_1

should be ≥ 1 because already schedulable

$$i = 1,\ \mathcal{P}_0(T_1) = \{3\}$$

$$C_1 \leq 3$$

$$i = 2, \mathcal{P}_1(T_2) = \{6, 8\}$$

$$C_1 \leq \max\left\{\frac{6-2}{2}, \frac{6-2}{3}\right\} = 2$$

$$i = 3,\ \mathcal{P}_2(T_3) = \{15, 16, 18, 20\}$$

$$C_1 \leq \max\left\{\frac{15-9}{5}, \frac{16-9}{6}, \frac{18-11}{6}, \frac{20-11}{7}\right\} = \frac{9}{7} \approx 1.286$$

$$C_1^{\mathsf{max}}(\mathsf{LL}) = T_1(U_{\mathsf{LL}} - \sum_{j\in\{2,3\}} U_i) = 3(0.780 - 0.5) = 0.839$$

Min schedulable T_1

Implicit deadline

$$i = 1,\ T_1 \geq R_1 = 1$$

$$i = 2,\ Y(\{2\}, D_2) = T_2 - C_2 = 8 - 2 = 6$$

$$n_1^{(2)} = \lfloor 6/1 \rfloor = 6$$

$$\text{if } n_1^{(2)} = 6,\ R_2 = 2 + 6 = 8$$

$$T_1^{(2)} = 8/6 = 4/3 = 1.333$$

$$i = 3,\ Y(\{2,3\}, 20) = \max_{t \in \{16,20\}} \{t - \lceil t/T_2 \rceil C_2 - \lceil t/T_3 \rceil C_3\}$$

$$= \max\{16 - 4 - 5, 20 - 6 - 5\} = 9$$

$$n_1^{(3)} = \lfloor 9/1 \rfloor = 9, \text{if } n_1^{(3)} = 9,\ R_3 = 20$$

$$T_1^{(3)} = 20/9 = 2.222$$

$$T_1^{\text{min}} = \max\{1, 1.333, 2.222\} = 2.222$$

$$T_1^{\text{min}}(\text{LL}) = 3.575$$