CHALMERS UNIVERSITY OF TECHNOLOGY | UNIVERSITY OF GOTHENBURG

Real-Time Systems

Exercise #6

Professor Jan Jonsson

Department of Computer Science and Engineering
Chalmers University of Technology

Pseudo-parallel execution

"For a task set to be schedulable with DM there cannot exist an instance of a task execution in the schedule where the worst-case response time of the task exceeds its deadline."

"For a task set to be schedulable with EDF there cannot exist an interval of length L in the schedule where the processor demand in that interval exceeds the length L."

The following example is based on an old exam problem.

Example: EDF + DM scheduling

Problem:

Consider a real-time system with five tasks, whose timing properties are listed in the table below. Tasks τ_3 and τ_4 cooperate in the sense that τ_4 uses the result produced by τ_3, and they must both complete within a common end-to-end deadline of $D_{global} = 75$. The global deadline should be divided between these two tasks, and X represents the part of the deadline that is assigned to τ_3. To guarantee that valid data is available at the beginning of its execution, the offset for τ_4 is set to X. The same offset is also used for task τ_5. All other tasks are supposed to be independent, with an offset of 0.

	C_i	O_i	D_i	T_i
τ_1	5	0	6	15
τ_2	6	0	11	25
τ_3	9	0	X	75
τ_4	15	X	$75 - X$	75
τ_5	15	X	25	25

Example: EDF + DM scheduling

Problem (cont'd):

The tasks are assigned to two separate processors that are connected through a shared-bus network. Processor 1 executes tasks τ_1, τ_2 and τ_3, and employs preemptive earliest-deadline-first (EDF) scheduling (that is, dynamic priorities.) Processor 2 executes tasks τ_4 and τ_5, and employs preemptive deadline-monotonic (DM) scheduling (that is, static priorities.) The overhead for sending data between the processors over the network is assumed to be negligible.

Derive the range of allowed (maximum and minimum) values of X for which all tasks will meet their deadlines.

Example: EDF + DM scheduling

Processor demand analysis for EDF:

- The processor demand for a task τ_i in a given time interval $[0, L]$ is the amount of processor time that the task needs in the interval in order to meet the deadlines that fall within the interval.
- Let N_i^L represent the number of instances of τ_i that must complete execution before L .

$$C_P(0, L) = \sum_{i=1}^{n} N_i^L C_i$$

Example: EDF + DM scheduling

- The total processor demand is

$$C_P(0,L) = \sum_{i=1}^{n} \left(\left\lfloor \frac{L - D_i}{T_i} \right\rfloor + 1 \right) C_i$$

A sufficient and necessary condition for earliest-deadline-first scheduling, for which $D_i \leq T_i$, is

$$\forall L \in K : C_P(0,L) \leq L$$

$$K = \left\{ D_i^k \mid D_i^k = kT_i + D_i, D_i^k \leq \mathrm{LCM}\left\{ T_1,\ldots,T_n \right\}, 1 \leq i \leq n, k \geq 0 \right\}$$

Example: EDF + DM scheduling

The easiest approach to solving this problem is to take advantage of the fact that the two processors are "isolated" from each other with respect to time. Since τ_4 and τ_5 have a common arrival time at $t = X$, this could be seen as the origin of these tasks' life cycles. And since τ_3 is assumed to have completed its execution no later than $t = X$, and the time to transfer its data to the other processor is assumed to be negligible, τ_3 and τ_4 need not synchronize its executions.

Processor 1 (EDF scheduling):

Since the task deadlines are shorter than the periods, we apply processor-demand analysis. We first derive the LCM for the tasks: $\text{LCM}\{\tau_1, \tau_2, \tau_3\} = \text{LCM}\{15, 25, 75\} = 75$.

We then calculate the set of control points K: $K_1 = \{6, 21, 36, 51, 66\}$, $K_2 = \{11, 36, 61\}$ och $K_3 = \{X\}$ which gives us $K = K_1 \cup K_2 \cup K_3 = \{6, 11, 21, 36, 51, 61, 66, X\}$.

Example: EDF + DM scheduling

Processor-demand analysis, including unknown control point X:

L	$N_1^L \cdot C_1$	$N_2^L \cdot C_2$	$N_3^L \cdot C_3$	$C_P(0,L)$	$C_P(0,L) \leq L$
6	$(\lfloor \frac{(6-6)}{15} \rfloor + 1) \cdot 5 = 5$	$(\lfloor \frac{(6-11)}{25} \rfloor + 1) \cdot 6 = 0$	$(\lfloor \frac{(6-X)}{75} \rfloor + 1) \cdot 9 =?$	$5 + ?$	OK if $X > 6$
11	$(\lfloor \frac{(11-6)}{15} \rfloor + 1) \cdot 5 = 5$	$(\lfloor \frac{(11-11)}{25} \rfloor + 1) \cdot 6 = 6$	$(\lfloor \frac{(11-X)}{75} \rfloor + 1) \cdot 9 =?$	$11 + ?$	OK if $X > 11$
21	$(\lfloor \frac{(21-6)}{15} \rfloor + 1) \cdot 5 = 10$	$(\lfloor \frac{(21-11)}{25} \rfloor + 1) \cdot 6 = 6$	$(\lfloor \frac{(21-X)}{75} \rfloor + 1) \cdot 9 =?$	$16 + ?$	OK if $X > 21$
36	$(\lfloor \frac{(36-6)}{15} \rfloor + 1) \cdot 5 = 15$	$(\lfloor \frac{(36-11)}{25} \rfloor + 1) \cdot 6 = 12$	$(\lfloor \frac{(36-X)}{75} \rfloor + 1) \cdot 9 =?$	$27 + ?$	OK
51	$(\lfloor \frac{(51-6)}{15} \rfloor + 1) \cdot 5 = 20$	$(\lfloor \frac{(51-11)}{25} \rfloor + 1) \cdot 6 = 12$	$(\lfloor \frac{(51-X)}{75} \rfloor + 1) \cdot 9 =?$	$32 + ?$	OK
61	$(\lfloor \frac{(61-6)}{15} \rfloor + 1) \cdot 5 = 20$	$(\lfloor \frac{(61-11)}{25} \rfloor + 1) \cdot 6 = 18$	$(\lfloor \frac{(61-X)}{75} \rfloor + 1) \cdot 9 =?$	$38 + ?$	OK
66	$(\lfloor \frac{(66-6)}{15} \rfloor + 1) \cdot 5 = 25$	$(\lfloor \frac{(66-11)}{25} \rfloor + 1) \cdot 6 = 18$	$(\lfloor \frac{(66-X)}{75} \rfloor + 1) \cdot 9 =?$	$43 + ?$	OK

As can be seen from the table, the processor demand never exceeds the length of the interval for the known control points if $X > 21$. While $X = 36$ is a safe choice, it is possible to find a smaller value of X by observing that the processor demand $C_P(0,L) = 25$ for $21 < X < 36$. This means that we can add $D_3 = X \geq 25$ as the last control point. Hence, tasks τ_1, τ_2 and τ_3 will meet their deadlines if $X \geq 25$.

Example: EDF + DM scheduling

Processor 2 (DM scheduling):

We first observe that $t = X$ is the critical instant for tasks τ_4 and τ_5. We can therefore apply response-time analysis to determine a suitable value for X.

We first check whether $D_4 < 25$, which would mean that τ_4 has the highest priority. However, the response time for τ_5 would then become $R_5 = C_4 + C_5 = 15 + 15 = 30 > D_5$, causing τ_5 to miss its deadline.

Consequently, $D_4 \geq 25$, and we use response time analysis to find the smallest value of D_4. Assume $R_4^0 = C_4 = 15$.

$R_4^1 = 15 + \lceil \frac{15}{25} \rceil \cdot 15 = 15 + 1 \cdot 15 = 30$

$R_4^2 = 15 + \lceil \frac{30}{25} \rceil \cdot 15 = 15 + 2 \cdot 15 = 45$

$R_4^3 = 15 + \lceil \frac{45}{25} \rceil \cdot 15 = 15 + 2 \cdot 15 = 45$ (convergence)

Therefore, we have that $D_4 \geq 45$.

Since $D_4 = 75 - X$, we can now see that $X \leq 75 - 45 = 30$. Hence, tasks τ_4 and τ_5 will meet their deadlines if $X \leq 30$.

Conclusion: By combining the results from the analysis of processors 1 and 2, we see that all tasks will meet their deadlines if $25 \leq X \leq 30$