Hold time lower bound

H.J. Rivera Verduzco 0977393

P.O. Box 513, 5600 MB Eindhoven, The Netherlands

Email: H.J.Rivera.Verduzco@student.tue.nl

April 12, 2017

1 BCHT for FPTS

Given a task-set $\mathcal{T}$, Algorithm 1 shows a procedure to derive a less pessimistic lower bound for the *best-case hold time* of a task τ_i. Table **??** explain the terminology for the variables used in the algorithm. Furthermore, function $BI_i(y, PA_d)$ returns the the largest positive solution satisfying

$$x = y + \sum_{h:\pi_h > \theta_i} \left(\left\lceil \frac{x}{T_h} \right\rceil - 1 \right)^+ BC_h + \sum_{d:\theta_i \geq \pi_d \pi_i} \left(\left\lceil \frac{x - PA_d}{T_d} \right\rceil - 1 \right)^+ BC_d, \tag{1}$$

where the notion w^+ stands for $\max(w, 0)$.

Table 1: Terminology.

Name	Descriptions
HLB_i^{init}	Initial lower bound. This is the *best-case hold time* of task τ_i when considering only preemptive tasks.
BH_i^{lb}	Lower bound of the *best-case hold time* of τ_i; $BH_i \geq BH_i^{lb}$.
UB_i	Upper bound in hold time when pushing the activation of delaying tasks.
PI_i	Push interval of delaying tasks relative to BH_i^{lb}.
DI_i	Delay interval of task τ_i is the time interval in which delaying tasks can affect the hold time.

Algorithm 1 Algorithm to derive a tighter lower bound for the *best-case hold time* of task τ_i.

1: **procedure** *bchtLowerBound*($\mathcal{T}, i$)
2: $HLB_i^{init} = BI_i(BC_i, \infty)$;
3: $BH_i^{lb} = HLB_i^{init}$;
4: $UB_i = BI_i(BC_i, BH_i^{lb})$;
5: **while** $UB_i > HLB_i^{init}$ **do**
6: $DI_i = UB_i - BH_i^{lb}$;
7: $PI_i = \min_{d:\theta_i \geq \pi_d > \pi_i} (DI_i \mod T_d)$;
8: $BH_i^{lb} = BH_i^{lb} + PI_i$;
9: $UB_i = BI_i(BC_i, BH_i^{lb})$;
end while
10: **return** BH_i^{lb};

In order to illustrate how the algorithm works, consider the task-set $\mathcal{T}_2$ described in Table 2. Figure 1 shows the notions after performing Algorithm 1 to calculate $BH_4^{lb} = 55$. Note that in Figure 1, the second job of τ_3 activated at time $t = 30$, namely $\iota_{3,2}$, does not start. This is to illustrate that the *best-case hold time* of τ_4 should be long enough in order to delay the start time of $\iota_{3,2}$, i.e. $BH_4 > BH_4^{lb}$. If this condition is not met, the job $\iota_{3,2}$ will start and delay the start time of τ_4. As a consequence, the job of τ_4 will experience preemptions by the higher priority tasks activated at time $t = 85$.

Table 2: Task set $\mathcal{T}_2$.

	T_i	$WC_i = BC_i$	π_i	θ_i	wl_i	BR_i	BH_i^{lb}	HLB_i^{init}
τ_1	80	14	4	4	1	14		
τ_2	80	6	3	3	1	6		
τ_3	30	15	2	2		15		
τ_4	240	50	1	2		56	55	50

The *least common multiple* of the periods is 240 and $U^{\mathcal{T}_2} \approx 0.96$.

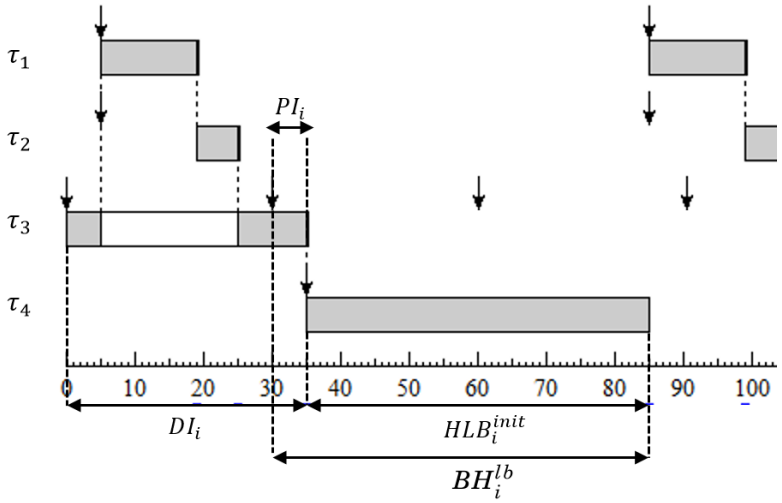

Figure 1: Schedule of task-set $\mathcal{T}_2$ with the notions described in Table **??**.

Note that the only way for the job of task τ_4 to have a hold time higher than $BH_4^{lb} = 55$ is by experiencing preemption by either task τ_1, τ_2 or both. Since τ_2 has a *best-case execution time* $BC_2 = 6$, we can choose this task to preempt τ_4 and the condition will be met because $HLB_4^{init} + BC_2 = 56$ which is higher than $BH_4^{lb} = 55$. This is indeed the *best-case hold time* because if we would choose τ_1 to preempt τ_4, the resulting hold time would be $HLB_4^{init} + BC_1 = 64$. Figure 2 shows the schedule where the *best-case hold time* of task τ_4 is assumed.

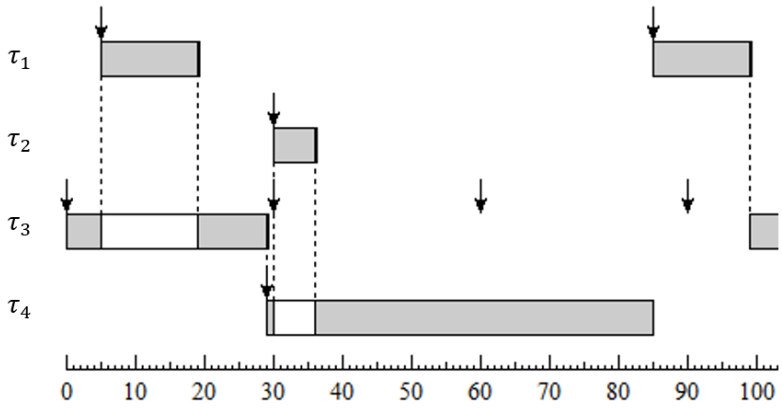

Figure 2: Schedule of task-set $\mathcal{T}_2$ where the *best-case hold time* of task τ_4 is assumed ($BH_4 = 56$).

References