

Sistemi Embedded e Real-time (M. Cesati)

Compito scritto del 22 settembre 2011

Esercizio 1. Si consideri il seguente sistema di task periodici : $T_1 = (6, 6, 2, 6)$, $T_2 = (0, 15, 3, 15)$, $T_3 = (24, 5, 2, 4)$.

(a) Nell'ipotesi che i job siano non interrompibili, verificare che non esiste alcuna dimensione ammissibile per il frame di una schedulazione ciclica strutturata.

(b) Nell'ipotesi che i job siano interrompibili, determinare una schedulazione ciclica strutturata che minimizza l'overhead dello scheduler ed il numero di frammenti dei job.

Esercizio 2. Si consideri un sistema di task periodici indipendenti, sempre interrompibili e che non si auto-sospendono: $T_1 = (3, 1)$, $T_2 = (4, 1)$, $T_3 = (6, 1)$, $T_4 = (12, 1)$.

(a) Supponendo che venga utilizzato uno scheduler EDF e che l'overhead dello scheduler e del cambio di contesto siano trascurabili, verificare analiticamente che il sistema è schedulabile su un singolo processore.

(b) Supponendo che lo scheduler EDF sia periodico con periodo $p_0 = 1/5$, che esso controlli la coda dei job pendenti in tempo $e_0 = 1/200$, e che il costo dell'attivazione di un job sia $CS_0 = 1/10$, verificare analiticamente che il sistema non è più schedulabile su un singolo processore.

(c) Con riferimento al punto (b): esiste una frequenza p_0 per lo scheduler che consente di garantire la schedulabilità del sistema? Giustificare la risposta.

Esercizio 3. Un sistema multiprocessore è dotato di $m = 16$ processori identici. Su di esso deve essere eseguito in ogni unità di tempo un lavoro totale stimato in 13 unità di tempo. Il carico è rappresentato da un certo numero di task periodici di frequenza minima pari a 1 schedulati per mezzo dell'algoritmo EDF-FF.

È possibile garantire la schedulabilità del sistema suddividendo in modo appropriato il carico tra 50 task periodici (non necessariamente identici)? Giustificare la risposta.

Sistemi Embedded e Real-time (M. Cesati)

Soluzioni del compito scritto del 22 settembre 2011

Esercizio 1. Si consideri il seguente sistema di task periodici : $T_1 = (6, 6, 2, 6)$, $T_2 = (0, 15, 3, 15)$, $T_3 = (24, 5, 2, 4)$.

(a) Nell'ipotesi che i job siano non interrompibili, verificare che non esiste alcuna dimensione ammissibile per il frame di una schedulazione ciclica strutturata.

Determiniamo le possibili dimensioni f del frame dello scheduler ciclico:

- I task hanno fasi 6, 0 e 24. Poiché f deve essere un divisore esatto di tutte le fasi si ha:

$$f \in \{1, 2, 3, 6\}.$$

- Vincolo sui tempi d'esecuzione dei job:

$$f \geq \max\{2, 3, 2\} = 3 \Rightarrow f \geq 3$$

- Vincolo sulla divisibilità della lunghezza dell'iperperiodo ($\text{mcm}\{6, 15, 5\} = 30$):

$$f \in \{1, 2, 3, 5, 6, 10, 15, 30\}.$$

Quindi considerando i vincoli precedenti: $f \in \{3, 6\}$.

- Vincolo sul task T_1 ($2f - \text{gcd}\{6, f\} \leq 6$):

$$\begin{aligned} 2 \cdot 3 - \text{gcd}\{6, 3\} = 3 &\leq 6 \Rightarrow f = 3 \text{ ok} \\ 2 \cdot 6 - \text{gcd}\{6, 6\} = 6 &\leq 6 \Rightarrow f = 6 \text{ ok} \end{aligned}$$

- Vincolo sul task T_2 ($2f - \text{gcd}\{15, f\} \leq 15$):

$$\begin{aligned} 2 \cdot 3 - \text{gcd}\{15, 3\} = 3 &\leq 15 \Rightarrow f = 3 \text{ ok} \\ 2 \cdot 6 - \text{gcd}\{15, 6\} = 9 &\leq 15 \Rightarrow f = 6 \text{ ok} \end{aligned}$$

- Vincolo sul task T_3 ($2f - \text{gcd}\{5, f\} \leq 4$):

$$\begin{aligned} 2 \cdot 3 - \text{gcd}\{5, 3\} = 5 &> 4 \Rightarrow f = 3 \text{ NO} \\ 2 \cdot 6 - \text{gcd}\{5, 6\} = 11 &> 4 \Rightarrow f = 6 \text{ NO} \end{aligned}$$

Si conclude che non esiste alcuna dimensione ammissibile per il frame.

(b) Nell'ipotesi che i job siano interrompibili, determinare una schedulazione ciclica strutturata che minimizza l'overhead dello scheduler ed il numero di frammenti dei job.

Avendo la possibilità di spezzare i job, il vincolo sulla dimensione minima del frame viene rilassato, e dunque è possibile prendere in considerazione anche i valori $f \in \{1, 2\}$ precedentemente scartati.

- Vincolo sul task T_1 ($2f - \gcd\{6, f\} \leq 6$):

$$2 \cdot 1 - \gcd\{6, 1\} = 1 \leq 6 \Rightarrow f = 1 \text{ ok}$$

$$2 \cdot 2 - \gcd\{6, 2\} = 2 \leq 6 \Rightarrow f = 2 \text{ ok}$$

- Vincolo sul task T_2 ($2f - \gcd\{15, f\} \leq 15$):

$$2 \cdot 1 - \gcd\{15, 1\} = 1 \leq 15 \Rightarrow f = 1 \text{ ok}$$

$$2 \cdot 2 - \gcd\{15, 2\} = 3 \leq 15 \Rightarrow f = 2 \text{ ok}$$

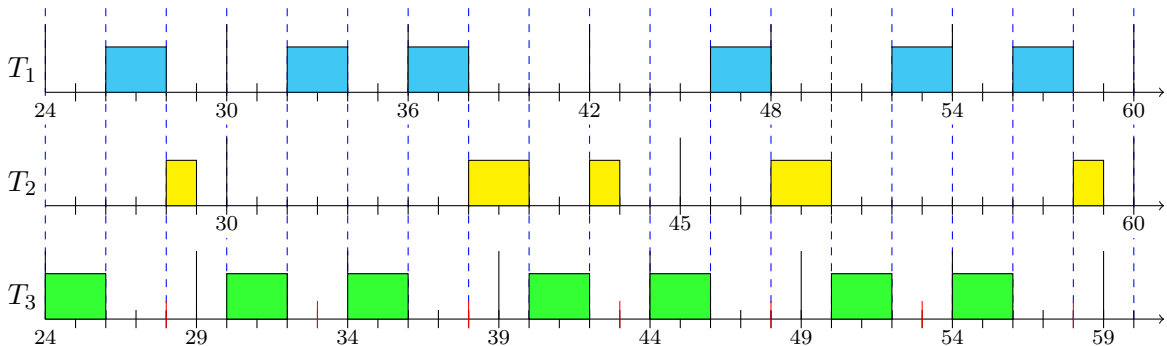
- Vincolo sul task T_3 ($2f - \gcd\{5, f\} \leq 4$):

$$2 \cdot 1 - \gcd\{5, 1\} = 1 \leq 4 \Rightarrow f = 1 \text{ ok}$$

$$2 \cdot 2 - \gcd\{5, 2\} = 3 \leq 4 \Rightarrow f = 2 \text{ ok}$$

Il valore $f = 2$ minimizza l'overhead dello scheduler, dunque consideriamo frame di dimensione 2. Per poter schedulare il job del task T_2 all'interno di un frame è necessario spezzarlo. Per avere un numero minimo di frammenti, imponiamo che ciascun job di T_2 sia suddiviso in un frammento di dimensione uguale a due ed in un frammento di dimensione uno.

L'esistenza di una schedulazione ciclica strutturata per $f = 2$ è dimostrata dal seguente diagramma che mostra i blocchi di schedulazione tra gli istanti 24 e 60:



Si noti che l'istante 24 corrisponde al rilascio del primo job di T_3 : nei blocchi di schedulazione dei frame precedenti a $t = 24$ sono assenti i job di T_3 . Analogamente, nei blocchi di schedulazione dei frame precedenti a $t = 6$ sono assenti anche i job di T_1 .

I blocchi di schedulazione dei 15 frame tra $t = 30$ e $t = 60$ possono ripetersi indefinitivamente a partire da $t = 60$. Infatti si consideri che:

- Agli istanti $t = 30$ e $t = 60$ è rilasciato un job di T_1 , e tutti i job precedentemente rilasciati sono stati già completati.
- Agli istanti $t = 30$ e $t = 60$ sono rilasciati i due frammenti di T_2 , e tutti i frammenti precedentemente rilasciati sono stati già completati.
- Agli istanti $t = 30$ e $t = 60$ risulta rilasciato e non eseguito un job di T_3 con scadenza pari a $t + 3$ (quindi, da eseguire immediatamente).

Pertanto, dato che le condizioni a $t = 60$ sono identiche a quelle di $t = 30$, la schedulazione tra 30 e 60 può ripetersi all'infinito.

Esercizio 2. Si consideri un sistema di task periodici indipendenti, sempre interrompibili e che non si auto-sospendono: $T_1 = (3, 1)$, $T_2 = (4, 1)$, $T_3 = (6, 1)$, $T_4 = (12, 1)$.

(a) Supponendo che venga utilizzato uno scheduler EDF e che l'overhead dello scheduler e del cambio di contesto siano trascurabili, verificare analiticamente che il sistema è schedulabile su un singolo processore.

Osserviamo preliminarmente che le scadenze relative dei task coincidono con i rispettivi periodi. Possiamo dunque cercare di applicare una condizione di schedulabilità basata sul fattore di utilizzazione del sistema di task.

Il fattore di utilizzazione totale del sistema è pari a

$$\frac{1}{3} + \frac{1}{4} + \frac{1}{6} + \frac{1}{12} = \frac{5}{6}.$$

Poiché $5/6 < 1 = U_{\text{EDF}}$, possiamo concludere immediatamente che il sistema di task è schedulabile.

(b) Supponendo che lo scheduler EDF sia periodico con periodo $p_0 = 1/5$, che esso controlli la coda dei job pendenti in tempo $e_0 = 1/200$, e che il costo dell'attivazione di un job sia $CS_0 = 1/10$, verificare analiticamente che il sistema non è più schedulabile su un singolo processore.

Poiché i job non si auto-sospendono, è possibile modellare il costo di attivazione aumentando i tempi di esecuzione dei job per un tempo pari a CS_0 . Pertanto, il sistema di task equivalente (considerando anche lo scheduler T_0) è

$$T_0 = (1/5, 1/200), \quad T'_1 = (3, 11/10), \quad T'_2 = (4, 11/10), \quad T'_3 = (6, 11/10), \quad T'_4 = (12, 11/10).$$

Il fattore di utilizzazione del sistema di task equivalente è:

$$U_{T'} = \frac{5}{200} + \frac{11}{30} + \frac{11}{40} + \frac{11}{60} + \frac{11}{120} = \frac{113}{120}.$$

La periodicità dello scheduler può rallentare l'esecuzione dei job. Infatti un job appena rilasciato non può entrare in esecuzione immediatamente, in quanto è lo scheduler che controlla la coda dei job pendenti e forza il cambio di contesto. Questo rallentamento può essere modellato considerandolo come un fattore aggiuntivo p_0 al tempo di blocco per non interrompibilità. Poiché i job per ipotesi sono sempre interrompibili si ha:

$$b_i(\text{np}) = p_0 = 1/5, \quad \text{per } 1 \leq i \leq 4.$$

Poiché i job possono bloccare è necessario controllare la schedulabilità di ciascun job separatamente:

- Task T'_1 : $U_{T'} + b_1/p_1 = 113/120 + 1/15 = 121/120 > 1 \quad \Rightarrow$ non schedulabile!
- Task T'_2 : $U_{T'} + b_2/p_2 = 113/120 + 1/20 = 119/120 < 1 \quad \Rightarrow$ schedulabile
- Task T'_3 : $U_{T'} + b_3/p_3 = 113/120 + 1/30 = 39/40 < 1 \quad \Rightarrow$ schedulabile
- Task T'_4 : $U_{T'} + b_4/p_4 = 113/120 + 1/60 = 23/24 < 1 \quad \Rightarrow$ schedulabile

Poiché T'_1 può mancare la propria scadenza, il sistema di task non è schedulabile.

(c) Con riferimento al punto (b): esiste una frequenza p_0 per lo scheduler che consente di garantire la schedulabilità del sistema? Giustificare la risposta.

Dall'analisi svolta al punto (b) risulta evidente che è sufficiente garantire la schedulabilità del task T'_1 . Infatti, se T'_1 fosse schedulabile, allora lo sarebbero certamente anche tutti gli altri task, in quanto hanno lo stesso identico tempo di blocco di T'_1 e periodi più grandi.

La condizione di schedulabilità per T'_1 può essere riformulata come:

$$\frac{e'_1}{p_1} + \frac{e'_2}{p_2} + \frac{e'_3}{p_3} + \frac{e'_4}{p_4} + \frac{e_0}{p_0} + \frac{p_0}{p_1} \leq 1.$$

In questa disuguaglianza tutti i termini sono noti tranne p_0 , che possiamo considerare come l'incognita x . Si ottiene pertanto la disuguaglianza

$$\frac{11}{12} + \frac{1}{200 \cdot x} + \frac{x}{3} \leq 1,$$

ossia

$$\frac{200 \cdot x^2 - 50 \cdot x + 3}{600 \cdot x} \leq 0.$$

Poiché possiamo assumere che $x > 0$, le soluzioni della disequazione sono costituite da tutti i valori entro l'intervallo determinato dalle eventuali soluzioni reali dell'equazione di secondo grado:

$$x_{1,2} = \frac{50 \pm \sqrt{50^2 - 4 \cdot 200 \cdot 3}}{2 \cdot 200} = \frac{5 \pm 1}{40} \quad \Rightarrow \quad x_1 = \frac{1}{10}, \quad x_2 = \frac{3}{20}$$

Possiamo dunque concludere che qualunque valore per il periodo p_0 dello scheduler compreso tra 0,1 e 0,15 garantisce la schedulabilità del sistema.

Esercizio 3. *Un sistema multiprocessore è dotato di $m = 16$ processori identici. Su di esso deve essere eseguito in ogni unità di tempo un lavoro totale stimato in 13 unità di tempo. Il carico è rappresentato da un certo numero di task periodici di frequenza minima pari a 1 schedulati per mezzo dell'algoritmo EDF-FF.*

È possibile garantire la schedulabilità del sistema suddividendo in modo appropriato il carico tra 50 task periodici (non necessariamente identici)? Giustificare la risposta.

Per avere la garanzia della schedulabilità del sistema si deve imporre:

$$U_T \leq U_{\text{EDF-FF}}.$$

Il fattore di utilizzazione dell'algoritmo EDF-FF è

$$U_{\text{EDF-FF}} = \frac{\beta \cdot m + 1}{\beta + 1} \quad \text{ove} \quad \beta = \left\lfloor \frac{1}{\max\{u_k\}} \right\rfloor.$$

D'altra parte, il carico di lavoro che il sistema dovrà sostenere è pari a $U_T = 13$. Quindi deve valere la disuguaglianza

$$13 \leq \frac{16\beta + 1}{\beta + 1}, \quad \text{ossia} \quad \beta \geq 4.$$

Consideriamo ora che il carico di lavoro entro un intervallo di una unità di tempo è pari a 13, e deve essere eseguito da 50 task periodici con periodo massimo pari a 1 (quindi

tutti rilasciati ed eseguiti entro l'intervallo di tempo). Ciò significa che almeno uno dei task periodici deve avere un fattore di utilizzazione non minore di $13/50$. Infatti in caso contrario si avrebbe:

$$\sum_{k=1}^{50} u_k < 50 \cdot 13/50 = 13, \quad \text{cioè} \quad U_T < 13.$$

Di conseguenza si ha che:

$$\max \{u_k\} \geq 13/50 \quad \Rightarrow \quad 1/\max \{u_k\} \leq 50/13 < 3,9$$

e dunque

$$\beta = \left\lfloor \frac{1}{\max \{u_k\}} \right\rfloor \leq \frac{1}{\max \{u_k\}} < 3,9 \quad \Rightarrow \quad \beta \leq 3.$$

La conclusione è che non è possibile garantire la schedulabilità del sistema in quanto la condizione $\beta \geq 4$ non è soddisfatta.