

Automazione

12 Febbraio 2016

Esercizio 1

Si consideri un sistema di controllo che implementi l'algoritmo RMPO in uno scheduler avente la struttura mostrata in Fig. 1.

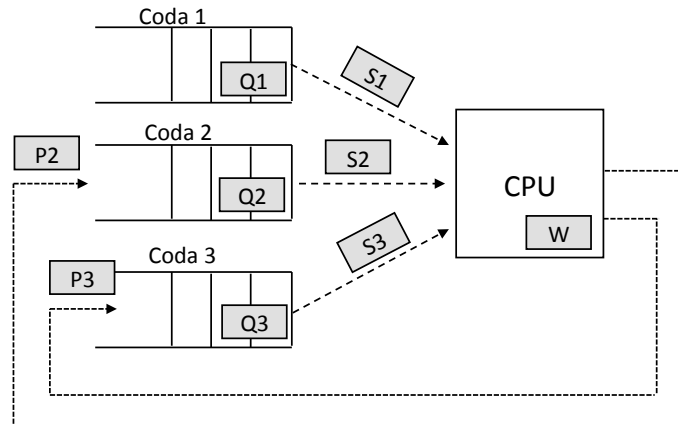


Figura 1: Struttura dello scheduler

Lo scheduler è composto da 3 code. La coda 1 ospita i task che hanno alta priorità, la coda 2 ospita i task che hanno priorità media e la coda 3 ospita i task che hanno bassa priorità. I sensori Q1, Q2 e Q3 valgono 1 quando è presente almeno un task nella coda, valgono 0 altrimenti.

Lo scheduler può mandare in esecuzione un task presente in una delle tre code attivando, in maniera esclusiva, uno dei relativi percorsi di instradamento verso la CPU. A regolare l'attivazione o disattivazione di tali percorsi di instradamento pensano gli attuatori S1, S2 e S3: se il valore è 1, un task viene spostato dalla relativa coda nella CPU per essere eseguito, se il valore è 0 i task rimangono in attesa nella coda.

Quando la CPU ospita un task, il sensore W ne rileva la presenza e assume valore 1. Quando la CPU termina l'esecuzione del task, lo scarta automaticamente e il sensore W assume il valore 0.

Infine lo scheduler può attuare pre-emption di task a media e bassa priorità attivando i percorsi di instradamento dalla CPU verso le relative code. In particolare gli attuatori P2 e P3 attivano i percorsi di pre-emption verso le code 2 e 3, rispettivamente. Quando un task in esecuzione subisce pre-emption e viene estratto dalla CPU, il sensore W ritorna al valore 0.

Si richiede di progettare un diagramma SFC di funzionamento del sistema di controllo che implementi l'algoritmo RMPO tenendo conto che:

- inizialmente tutte le code e la CPU sono vuote;
- non è necessario tenere conto della percentuale di esecuzione di un task che ha subito pre-emption o del tempo di esecuzione necessario a portare a termine un task;
- non è necessario tenere conto delle deadline dei task (questo controllo esula dall'obiettivo dell'esercizio).

[continua]

Esercizio 2

Si consideri la rete di Petri mostrata in Fig. 2, con la sua marcatura iniziale $\mathbf{x}_0 = (1 \ 1 \ 0 \ 0 \ 0)^T$.

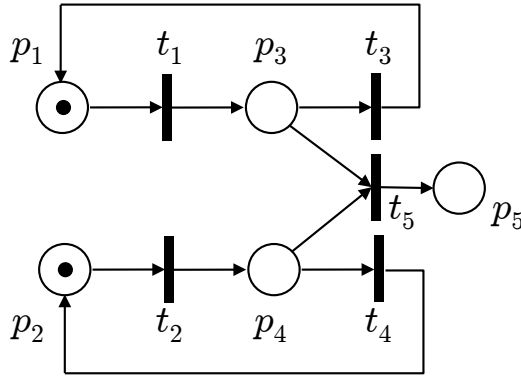


Figura 2: Una rete di Petri con la marcatura iniziale \mathbf{x}_0

- Utilizzando l'albero di raggiungibilità, mostrare che l'evoluzione della rete può terminare in un deadlock. Descrivere in modo compatto l'insieme di tutte le marcature raggiungibili.
- L'aggiunta di un posto di controllo con un arco in ingresso alla transizione t_5 e senza alcun token all'avvio è sufficiente a bloccare per sempre la possibilità di scatto di questa transizione. In tal modo, anche la situazione di deadlock viene eliminata. Si mostri che questa semplice struttura di supervisione può essere ricavata mediante la tecnica dei P -invarianti, imponendo un'opportuna disequazione lineare sulle marcature raggiungibili.
- Con la stessa tecnica di progetto, ricavare (almeno) due altri supervisor che garantiscano l'eliminazione di situazioni di deadlock e che utilizzino rispettivamente un solo posto e due posti monitor. Discutere i risultati ottenuti e ricavare per uno dei casi trovati il nuovo albero di raggiungibilità.

[continua]

Esercizio 3

Parte A. In una macchina punzonatrice automatica, come quella mostrata in Fig. 3(a-b), due servomotori elettrici M_1 e M_2 devono azionare in modo coordinato il moto oscillatorio periodico della tavola di supporto lungo un asse orizzontale e quello oscillatorio periodico della testa porta-attrezzo lungo l'asse verticale. Si consideri la schematizzazione di Fig. 3(c), dove è indicato il sistema di riferimento assi (x, y) rispetto al quale sono espresse tutte le grandezze geometriche. Per il motore M_1 (master) che aziona la tavola, si progetti una legge di moto $y = y(t)$ di periodo T , che parta da $y = 0$ in $t = 0$ e abbia ampiezza picco-picco L . Per asservire il motore M_2 (slave) a M_1 , si realizzi una cosiddetta *camma elettronica*. La posizione x dell'asse verticale deve essere espressa come funzione $x = x(y)$ della sola posizione y , con le seguenti condizioni al contorno: per $y = 0$ e $y = L$, sia $x(0) = x(L) = 0$ (posizione di attesa); per $y = L/2$, ovvero al centro corsa dell'asse orizzontale, sia $x(L/2) = \Delta$ (posizione di punzonatura). Il profilo di velocità dei due motori deve essere una funzione continua del tempo e partire dal valore nullo al tempo iniziale $t = 0$. Si traccino i grafici delle funzioni $y(t)$ e $x(y)$ scelte. Fornire inoltre l'espressione delle velocità $\dot{y}(t)$ e $\dot{x}(t)$ e, se possibile, anche i grafici dei loro andamenti temporali (almeno qualitativi).

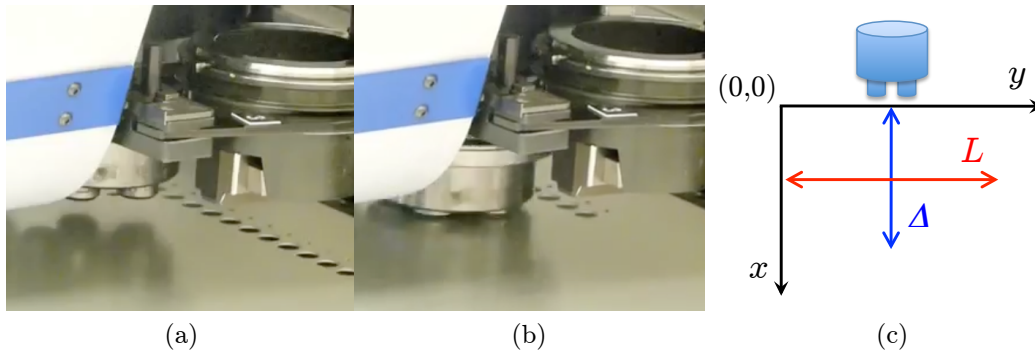


Figura 3: Una macchina punzonatrice automatica, in fase di riposo (a) e di punzonatura (b), con uno schema di riferimento per il movimento coordinato lungo i due assi (c)

Parte B. Siano dati i valori numerici degli spostamenti massimi $L = 10$ [cm] e $\Delta = 2$ [cm] della macchina punzonatrice. I due servomotori montano, ciascuno sul proprio asse di rotazione, lo stesso modello di encoder assoluto e sono dotati di riduttori, che convertono anche il moto da angolare a lineare, con rapporti di trasformazione rispettivamente pari a $N_1 = 50$ giri/cm e $N_2 = 20$ giri/cm. Quanti bit deve avere il modello utilizzato di encoder assoluto affinché l'incertezza nel posizionamento lungo gli assi cartesiani non sia superiore a 0.01 mm nel caso peggiore? Quali altre considerazioni sono importanti nella scelta del livello di risoluzione di un encoder assoluto?

[180 minuti; libri aperti]

Soluzioni

12 Febbraio 2016

Esercizio 1

La soluzione è mostrata in Fig. 4.

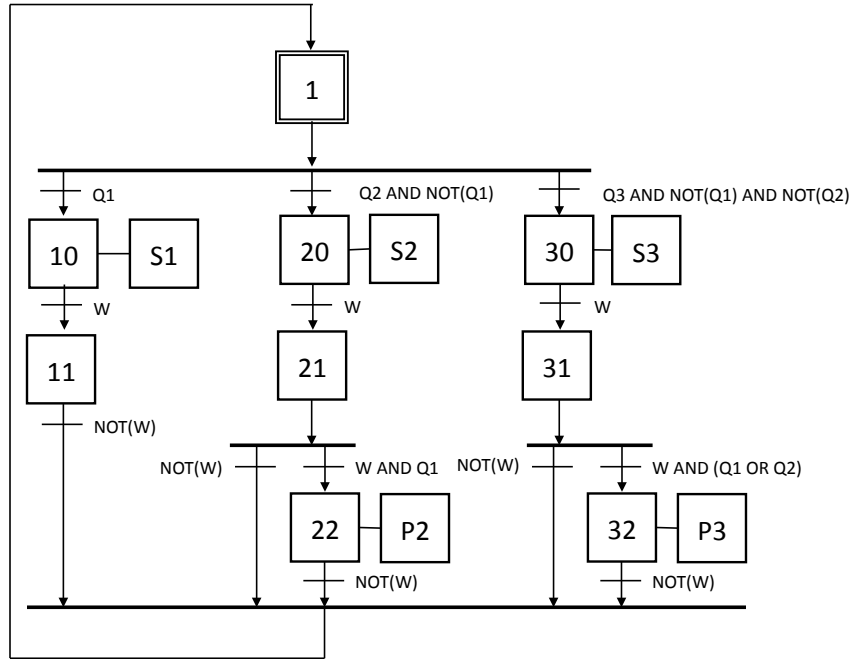


Figura 4: Diagramma SFC dell'algoritmo RMPO

Il diagramma SFC può essere interpretato come segue.

- Lo stato 1 è lo stato iniziale in cui la CPU non sta eseguendo alcun task.
- La divergenza è attivata dalla presenza di un task in una delle tre code. La priorità delle code 1, 2 e 3 è stabilita dalle condizioni associate alla divergenza.
- Quando la scelta è effettuata, si attiva lo stato corrispondente (10, 20 o 30) e l'azione associata è quella di attivare il percorso di instradamento dalla coda di attesa verso la CPU, così che il task venga mandato in esecuzione (si aspetta che $W = 1$).
- Se il task è a massima priorità, viene eseguito senza interruzione (stato 11) fino a che il processore non si libera dello stesso: condizione $\text{NOT}(W)$.
- Se il task in esecuzione è di priorità media (stato 21), verrà eseguito o fino alla sua terminazione o fino all'arrivo in coda 1 di un task ad alta priorità. In quest'ultimo caso (stato 22) si attiva il percorso di pre-emption ($P2 = 1$) verso la coda 2 fino a quando il task non esce dalla CPU, $\text{NOT}(W)$, per ritornare nello stato iniziale (stato 1) e lasciare spazio all'esecuzione del task ad alta priorità.

- Se il task in esecuzione è di priorità bassa (stato 31), verrà eseguito o fino alla sua terminazione o fino all'arrivo in coda 1 o in coda 2, di un task a priorità alta o media. In quest'ultimo caso (stato 32) si attiva il percorso di pre-emption ($P3 = 1$) verso la coda 3 fino a quando il task non esce dalla CPU, NOT(W), per ritornare nello stato iniziale (stato 1) e lasciare spazio all'esecuzione del task a priorità alta o media.
- In ogni caso, quando un task termina o subisce pre-emption, la convergenza riporta il diagramma nello stato iniziale (stato 1).

Esercizio 2

La matrice di incidenza \mathbf{C} della rete di Petri in Fig. 2 ha dimensioni (5×5) :

$$\mathbf{C} = \begin{pmatrix} -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 & -1 \\ 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (1)$$

Sebbene non esplicitamente richiesto, possiamo analizzare le caratteristiche dei P -invarianti della rete. E' possibile verificare (con la funzione `rank` in Matlab, ma non solo!) che la matrice \mathbf{C} in (1) ha rango pari a 3. La dimensione dello spazio nullo di \mathbf{C}^T è quindi pari a $5 - 3 = 2$. L'equazione matriciale che definisce i P -invarianti,

$$\boldsymbol{\gamma}^T \mathbf{C} = \mathbf{0}^T \quad \Leftrightarrow \quad \mathbf{C}^T \boldsymbol{\gamma} = \mathbf{0},$$

si traduce nelle tre equazioni scalari indipendenti

$$\begin{aligned} \gamma_1 - \gamma_3 &= 0 \\ \gamma_2 - \gamma_4 &= 0 \\ \gamma_5 - \gamma_3 - \gamma_4 &= 0 \end{aligned}$$

che ammettono dunque ∞^2 soluzioni linearmente indipendenti. E' semplice determinare la struttura di una base per tutti i P -invarianti, parametrizzata dagli scalari $\alpha \in \mathbb{Z}^+$ e $\beta \in \mathbb{Z}^+$:

$$\boldsymbol{\gamma}_1^T = (\alpha \ 0 \ \alpha \ 0 \ \alpha), \quad \boldsymbol{\gamma}_2^T = (0 \ \beta \ 0 \ \beta \ \beta).$$

Il fatto che queste due basi per i P -invarianti (per $\alpha = \beta = 1$, $\boldsymbol{\gamma}_1$ e $\boldsymbol{\gamma}_2$ sono due vettori canonici di supporto minimo) ricoprono tutto l'insieme dei posti della rete indica che la rete è limitata nella sua evoluzione.

La rete può bloccarsi in una situazione di deadlock, come è possibile vedere dalla presenza di foglie nell'albero di raggiungibilità di Fig. 5 nelle quali nessuna transizione è abilitata. L'insieme delle marcature raggiungibili a partire da \mathbf{x}_0 è dato da:

$$\mathcal{R}(PN) = \left\{ \mathbf{x}_0, \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C, \mathbf{x}_D \right\} = \left\{ \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right\}.$$

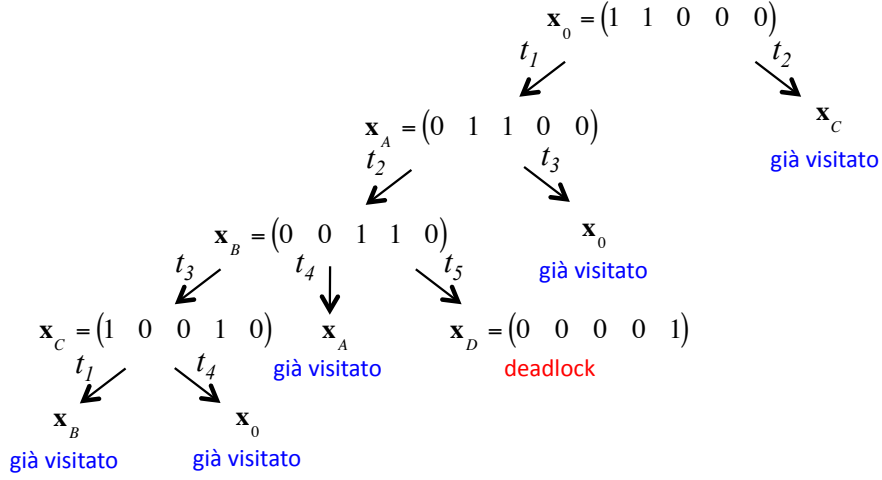


Figura 5: Albero di raggiungibilità della rete di Petri in Fig. 2, esplorato in profondità (*depth-first*) e con ordine crescente di numerazione delle transizioni abilitate in una data marcatura

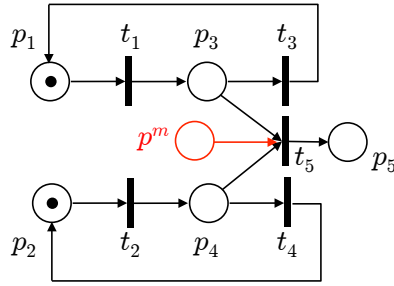


Figura 6: La rete di Fig. 2 con un posto monitor p^m vuoto che disabilita la transizione t_5

L'unica marcatura di deadlock è la \mathbf{x}_D e si raggiunge se e solo se scatta la transizione t_5 . La semplice idea suggerita nel testo è quella di bloccare definitivamente l'abilitazione allo scatto della transizione t_5 , aggiungendo un posto *vuoto* di controllo p^m in ingresso come mostrato in Fig. 6.

E' facile verificare che tale soluzione si può ottenere con il metodo di progetto del supervisore basato sui P -invarianti. In questo metodo si lavora sulla rete di Petri originaria, aggiungendo un posto monitor per ogni disequazione imposta sulle marcature raggiungibili e, in corrispondenza, una riga alla matrice di incidenza (1). Basta allora imporre la disequazione

$$x(p_5) \leq 0 \quad \Rightarrow \quad \mathbf{h}^T \mathbf{x} = (0 \ 0 \ 0 \ 0 \ 1) \mathbf{x} \leq 0 = k. \quad (2)$$

L'interpretazione della condizione (2) è altrettanto semplice: non si vogliono avere marche nel posto p_5 e quindi non si farà scattare l'unica transizione che ce li possa mettere, che è anche quella il cui scatto porta al deadlock. In base alla (2), il metodo prevede l'aggiunta di una riga alla matrice di incidenza e l'inizializzazione del posto monitor p^m come segue:

$$\begin{aligned} \mathbf{C}^m &= -\mathbf{h}^T \mathbf{C} = (0 \ 0 \ 0 \ 0 \ -1), \\ x_0(p^m) &= k - \mathbf{h}^T \mathbf{x}_0 = 0. \end{aligned} \quad (3)$$

Il risultato è proprio il posto monitor (senza token) aggiunto in Fig. 6. Si noti che alla stessa conclusione si arriva imponendo

$$\sum_{i=1}^4 x(p_i) + 3x(p_5) \leq 2 \Rightarrow \mathbf{h}^T \mathbf{x} = (1 \ 1 \ 1 \ 1 \ 3) \mathbf{x} \leq 2 = k, \quad (4)$$

da cui è semplice verificare che si ritrovano le stesse \mathbf{C}^m e $x(p^m)$ determinate in (3). L'interpretazione è meno immediata: la somma (pesata su $x(p_5)$) di tutti i token nella rete non deve eccedere quella iniziale. Il peso 3 sul posto p_5 impone di fatto che nessun token possa andare a finire nel posto p_5 , pena la violazione della (4) (essendo tutte le marcature sempre non negative).

Un progetto alternativo evita la marcatura simultanea dei posti p_3 e p_4 che portano all'abilitazione allo scatto della transizione t_5 . Si pone quindi

$$x(p_3) + x(p_4) \leq 1 \Rightarrow \mathbf{h}^T \mathbf{x} = (0 \ 0 \ 1 \ 1 \ 0) \mathbf{x} \leq 1 = k. \quad (5)$$

Da questa segue

$$\begin{aligned} \mathbf{C}^m &= -\mathbf{h}^T \mathbf{C} = (-1 \ -1 \ 1 \ 1 \ -2), \\ x_0(p^m) &= k - \mathbf{h}^T \mathbf{x}_0 = 1 - 0 = 1. \end{aligned}$$

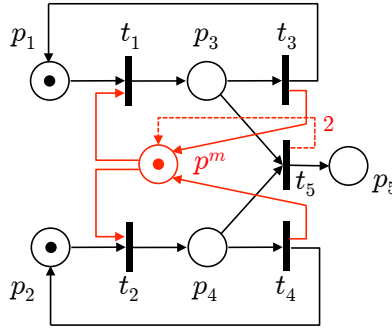


Figura 7: Supervisore con un posto monitor p^m che impone la disequazione (5)

Il supervisore ottenuto in base alla (5) è evidenziato nella Fig. 7. L'arco tratteggiato ha peso 2. Poiché la marcatura iniziale \mathbf{x}_0 soddisfa già la disequazione (5), questa sarà soddisfatta anche da tutte le marcature raggiungibili. Nell'evoluzione della rete c'è ancora conflitto tra gli scatti delle transizioni t_1 e t_2 (risolto in modo casuale), ma non più tra la transizione t_3 e la t_5 né tra la t_4 e la t_5 . Non potendo mai scattare t_5 , l'arco tratteggiato tra questa transizione e il posto monitor p_m può essere eliminato senza alcuna conseguenza. Di fatto, questo risultato si ottiene con la scelta

$$x(p_3) + x(p_4) + 2x(p_5) \leq 1 \Rightarrow \mathbf{h}^T \mathbf{x} = (0 \ 0 \ 1 \ 1 \ 2) \mathbf{x} \leq 1 = k, \quad (6)$$

da cui segue (in rosso il termine che risulta modificato)

$$\begin{aligned} \mathbf{C}^m &= -\mathbf{h}^T \mathbf{C} = (-1 \ -1 \ 1 \ 1 \ 0), \\ x_0(p^m) &= k - \mathbf{h}^T \mathbf{x}_0 = 1 - 0 = 1 \end{aligned}$$

come mostrato in Fig. 8. L'albero di raggiungibilità della rete così supervisionata è riportato di Fig. 9. La rete supervisionata è ora reversibile (quindi con assenza di deadlock) e con la transizione t_5 mai attiva (non viva).

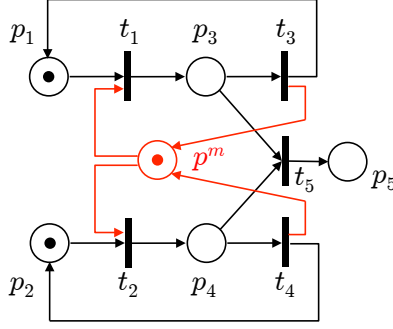


Figura 8: Supervisore con il posto monitor p^m modificato per imporre la disequazione (6)

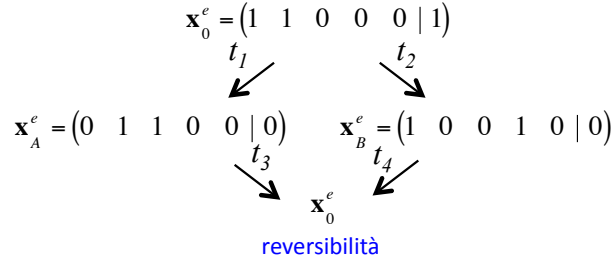


Figura 9: Albero di raggiungibilità della rete di Petri supervisionata di Fig. 8

Passiamo ora ad analizzare il progetto di un supervisore che eviti deadlock e abbia due posti monitor. Una soluzione banale al problema consiste nell'imporre simultaneamente le due disequazioni (2) e (6). Il supervisore sarà ovviamente costituito dalla composizione dei due posti monitor e dai relativi archi in rosso nelle Figg. 6 e 8.

Una soluzione più interessante è quella che impone un limite inferiore e uno superiore alla somma delle marche nei posti p_2 e p_3 :

$$1 \leq x(p_2) + x(p_3) \leq 2 \quad \Rightarrow \quad \mathbf{H}^T \mathbf{x} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 \end{pmatrix} \mathbf{x} \leq \begin{pmatrix} 2 \\ -1 \end{pmatrix} = \mathbf{k}. \quad (7)$$

In tal modo, si 'scardina' la marcatura che abiliterebbe t_5 sbilanciando la rete in modo che ci siano successioni di scatti che si alternano tra la sequenza $\{t_3, t_1\}$ (anello superiore) e la sequenza $\{t_2, t_4\}$ (anello inferiore). Infatti, dalle disequazioni (7) segue

$$\begin{aligned}
 \mathbf{C}^m &= -\mathbf{H}^T \mathbf{C} = \begin{pmatrix} -1 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 \end{pmatrix}, \\
 \mathbf{x}_0(p^m) &= \begin{pmatrix} x_0(p_1^m) \\ x_0(p_2^m) \end{pmatrix} = \mathbf{k} - \mathbf{H}^T \mathbf{x}_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.
 \end{aligned}$$

La rete di Petri supervisionata così risultante è mostrata in Fig. 10. L'albero di raggiungibilità è riportato in Fig. 11. Le sequenze di scatto ammissibili sono, come previsto, del tipo $\{t_1, (t_2, t_4)^h, (t_3, t_1)^k, \dots\}$, con arbitrari $h, k \geq 0$ che si possono ripetere all'infinito. Si noti che un comportamento del tutto analogo può ottenersi, specularmente, sostituendo alle (7) le disequazioni

$$1 \leq x(p_1) + x(p_4) \leq 2,$$

con sequenze di scatto del tipo $\{t_2, (t_1, t_3)^h, (t_4, t_2)^k, \dots\}$.

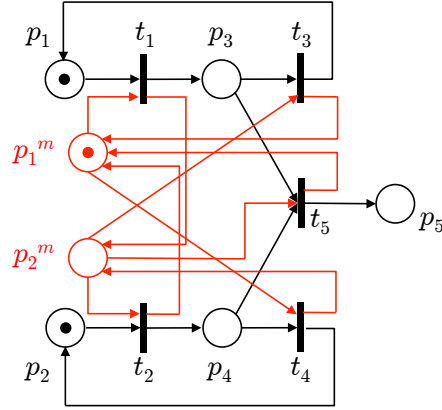


Figura 10: Supervisore con due posti monitor p_1^m e p_2^m per imporre le disequazioni (7)

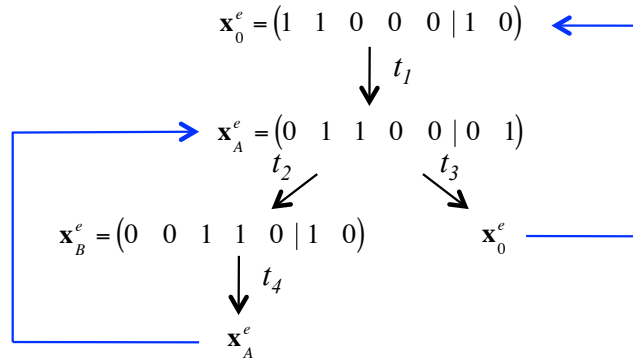


Figura 11: Albero di raggiungibilità della rete di Petri supervisionata di Fig. 10

Un ulteriore supervisore che evita deadlock e utilizza due posti di controllo è mostrato in Fig. 12. In questo caso il flusso dei token avviene in maniera strettamente alternata tra i due cicli $\{t_1, t_3\}$ e $\{t_2, t_4\}$, senza mai alcun conflitto nella sequenza di scatti. Si può però mostrare che tale struttura non è ottenibile tramite il progetto di due disequazioni sulle marcature dei posti dei tipi visti in precedenza. Si prenda ad esempio la riga aggiuntiva C_1^m della matrice di incidenza estesa relativa al posto p_1^m ,

$$C_1^m = (1 \ 0 \ 0 \ -1 \ 0).$$

Per essere generabile da una disequazione del tipo $\mathbf{h}_1^T \mathbf{x} \leq k_1$, dovrebbe essere risolubile rispetto ad \mathbf{h}_1 il sistema di equazioni lineari

$$-\mathbf{h}_1^T \mathbf{C} = C_1^m \quad \Rightarrow \quad \mathbf{C}^T \mathbf{h}_1 = (-1 \ 0 \ 0 \ 1 \ 0)^T.$$

E' facile verificare che non esiste nessuna soluzione \mathbf{h}_1 per questo sistema: il vettore dei termini noti non appartiene infatti all'immagine $\mathcal{R}\{\mathbf{C}^T\}$ della matrice \mathbf{C}^T (che, come visto, ha rango pari a 3). Analoghe considerazioni valgono per il posto di controllo p_2^m .

Al termine di questa lunga carrellata di possibilità, vale la pena di citare una soluzione di progetto che fa parte di una classe di supervisori più ampia di quella finora considerata, ossia contenente sia

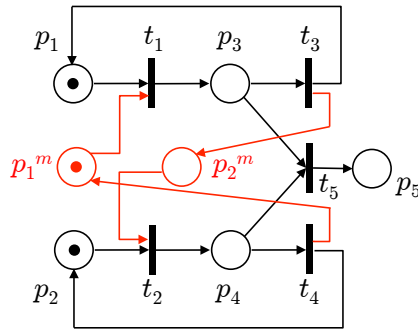


Figura 12: La rete di Petri di Fig. 2 supervisionata con due posti di controllo p_1^m e p_2^m

posti di controllo che transizioni di controllo ausiliarie. Questo tipo di supervisore, con dinamica di evoluzione propria (legata alle eventuali transizioni aggiunte interne al supervisore), permette di aggiungere sia archi tra posti del supervisore e transizioni della rete (come fatto finora) sia anche archi tra posti della rete e transizioni del supervisore. La rete supervisionata con un posto e due transizioni di controllo della Fig. 13 risolve elegantemente il problema di impedire deadlock nella rete, mantenendo stavolta tutte le transizioni vive. Ovviamente si può fare anche a meno del posto di controllo p^m e delle seconda transizione t_2^m , collegando due archi in uscita dalla transizione di controllo t_1^m ai posti originali p_1 e p_2 .

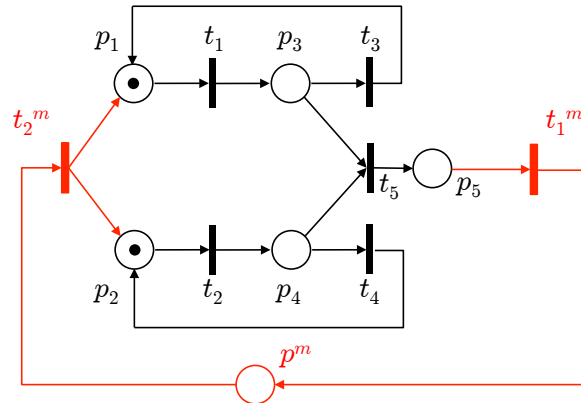


Figura 13: La rete di Petri di Fig. 2 con un supervisore dinamico avente un posto p^m e due transizioni t_1^m e t_2^m di controllo

Esercizio 3

Un profilo temporale di posizione dell'asse movimentato dal motore M_1 (master) che soddisfa i requisiti richiesti può essere

$$y(t) = \frac{L}{2} \left(1 - \cos \frac{2\pi t}{T} \right), \quad \forall t \geq 0.$$

Si noti che $y \in [0, L]$ (sempre non negativo) e che in particolare $y(0) = y(T) = 0$, $y(T/2) = L$ e $y(T/4) = y(3T/4) = L/2$. La legge parametrica $x = x(y)$ per la posizione di riferimento dell'asse

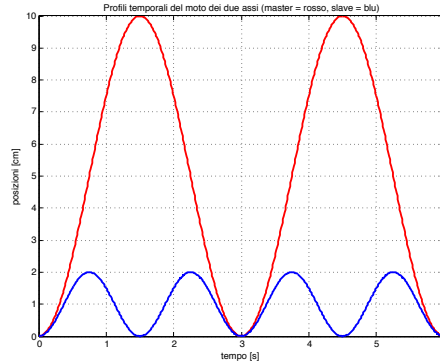


Figura 14: I profili di posizione $y(t)$ dell'asse master (in rosso) e $x(t)$ dell'asse slave (in blu)

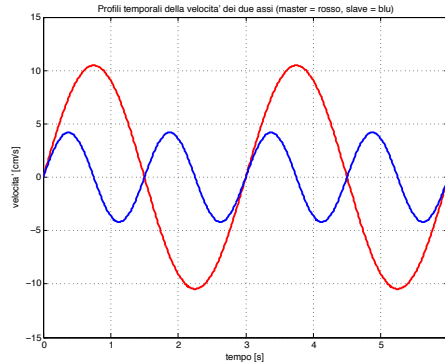


Figura 15: I profili delle velocità $\dot{y}(t)$ e $\dot{x}(t)$ dei due assi

guidato dal motore M_2 (slave) deve soddisfare alle seguenti tre condizioni al contorno:

$$x(0) = 0, \quad x(L) = 0, \quad x\left(\frac{L}{2}\right) = \Delta. \quad (8)$$

Si può allora scegliere ad esempio una generica funzione quadratica (che ha tre parametri)¹:

$$x(y) = a_0 + a_1y + a_2y^2. \quad (9)$$

Imponendo le condizioni (8) alla (9) si ottiene

$$a_0 = 0, \quad a_1 = \frac{4\Delta}{L}, \quad a_2 = -\frac{4\Delta}{L^2},$$

e la (9) può risciversi come

$$x(y) = \Delta \left(1 - \left(\frac{2}{L}\right)^2 \left(y - \frac{L}{2}\right)^2 \right), \quad \text{con } x \in [0, \Delta]. \quad (10)$$

¹Aumentando il grado del polinomio $x(y)$ si possono imporre ulteriori condizioni al contorno. Ad esempio, volendo imporre una derivata spaziale $x' = \partial x / \partial y$ nulla per $y = 0$ e $y = L$, si può utilizzare un polinomio di grado 4 (con cinque parametri).

La (10) soddisfa dunque tutti i requisiti. Le velocità lungo i due assi sono

$$\dot{y}(t) = \frac{L\pi}{T} \sin \frac{2\pi t}{T},$$

$$\dot{x}(t) = \frac{\partial x(y)}{\partial y} \dot{y} = -\frac{8\Delta}{L^2} \left(y(t) - \frac{L}{2} \right) \dot{y}(t) = \frac{4\pi\Delta}{T} \sin \frac{2\pi t}{T} \cos \frac{2\pi t}{T} = \frac{2\pi\Delta}{T} \sin \frac{4\pi t}{T}.$$

Nelle Figg. 14 e 15 sono riportati rispettivamente i profili di posizione e di velocità degli assi master e slave ottenuti per i valori $L = 10$ [cm], $\Delta = 2$ [cm] e $T = 3$ [s]. Sono mostrati due cicli di moto dell'asse master. Le velocità verificano la continuità richiesta. Il grafico della funzione geometrica $x = x(y)$ scelta in (10) è mostrato in Fig. 16, mentre quello della sua tangente spaziale $x'(y) = \partial x(y)/\partial y$ è in Fig. 17.

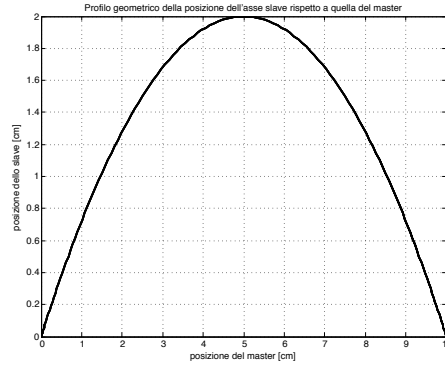


Figura 16: La funzione geometrica $x(y)$ fornisce la posizione desiderata dello slave in funzione della posizione del master

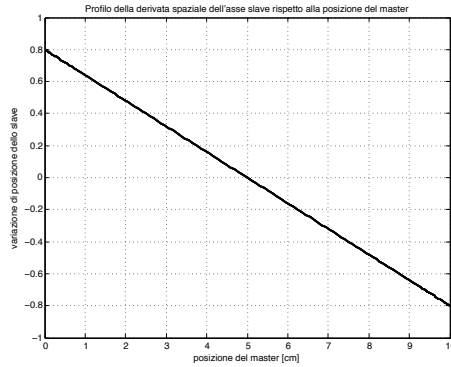


Figura 17: Il grafico della tangente $x'(y)$ al profilo spaziale dello slave in funzione della posizione del master

La risoluzione angolare di un encoder assoluto con N_t tracce = bit è pari a $360^\circ/2^{N_t}$ sull'asse di rotazione del motore dove è montato il dispositivo di misura. Questo valore, quando riportato a valle del riduttore, si riduce per un fattore pari a $n_i = 1/N_i < 1$ per il motore M_i ($i = 1, 2$). Per

soddisfare la specifica sull'incertezza massima nel posizionamento lungo i due assi di moto, si ha

$$\frac{360^\circ}{2^{N_t}} \cdot \max \left\{ \frac{1/50}{360^\circ}, \frac{1/20}{360^\circ} \right\} = \frac{1}{2^{N_t} \cdot 20} \leq 0.001 \text{ [cm]} \Rightarrow 2^{N_t} \geq 50 \Rightarrow N_t = \lceil \log_2 50 \rceil = 6.$$

Si noti che tale numero $N_t = 6$ di bit non dipende dalle escursioni massime L e Δ .

* * * * *