Automazione I

13 Febbraio 2014

Esercizio 1

Si consideri un sistema di automazione industriale in cui, a livello di coordinamento, è necessario portare a termine i seguenti task periodici:

1. ogni 4 time units (t.u.) una lamiera viene portata in zona saldatura impiegando 1 t.u.;
2. ogni 16 t.u. le lamiere vengono saldate impiegando 3 t.u.;
3. ogni 8 t.u. le lamiere saldate vengono portate via dalla zona saldatura in 1 t.u.

A questi tre task se ne aggiunge un quarto di pulitura degli elettrodi di saldatura, operazione che deve intervenire con un intervallo minimo di 32 t.u. e durare al massimo 8 t.u. Per semplicità, si ipotizzi che i task periodici siano indipendenti l'uno dall'altro. Tutti i task fin qui descritti devono essere gestiti con una modalità di scheduling hard real time. Il sistema di automazione viene supervisionato da un tecnico il quale, attraverso un'interfaccia uomo-macchina costituita da un touch screen, può visualizzare una schermata di stato del sistema. In particolare, la prima occorrenza di tale task aperiodico è attivata all'istante $a_5(1) = 20$ t.u., ha un computation time pari a $C_5(1) = 6$ t.u. e una deadline assoluta pari a $D_5(1) = 32$ t.u. Questo task deve essere servito con una modalità di scheduling soft real time. Si chiede di rispondere ai seguenti punti:

- verificare se il problema di scheduling hard real time non sia inammissibile;
- nel caso in cui il sistema non sia inammissibile, verificare se sussista una condizione sufficiente per l'ammissibilità del problema con RMPO;
- mostrare il risultato dello scheduling usando RMPO come algoritmo di task scheduling hard real time e un servizio in background con scheduling di tipo FIFO per i task soft real time;
- determinare se il task soft real time venga eseguito in questo modo entro la deadline assoluta.

Esercizio 2

Rispondere a ciascuna domanda con un breve testo accompagnato da figure o formule:

1. I nodi di una rete sono connessi ad un bus di comunicazione il cui arbitraggio di accesso è gestito con il protocollo CSMA-CR (*Wired AND*, con valore logico 0 come bit dominante). Tre nodi vogliono trasmettere contemporaneamente sulla rete e i relativi messaggi hanno le seguenti codifiche di priorità:

$$\mathrm{Id}_1 = 010011001, \qquad \mathrm{Id}_2 = 010101001, \qquad \mathrm{Id}_3 = 010011011.$$

   Tracciare il diagramma temporale della contesa del bus indicando quale nodo avrà l'accesso per trasmettere.
2. Per un processo a tempo continuo contenente un ritardo finito, è stato rilevato sperimentalmente in uscita l'andamento di Fig. 1, in risposta a un comando a gradino unitario applicato all'istante $t = 1$ sec. Desiderando errore nullo a regime permanente e un buon transitorio, si forniscano i valori dei guadagni di un regolatore PI utilizzando il primo metodo di Ziegler-Nichols (indicare esplicitamente la forma della legge di controllo PI a cui si fa riferimento). In questo caso, sarebbe sufficiente operare con un semplice regolatore di tipo P?

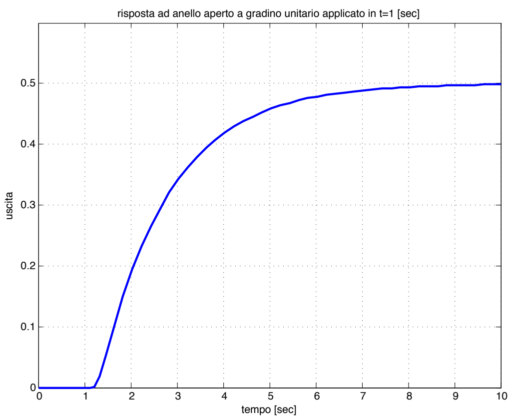

Figura 1: Risposta a gradino unitario di un processo con ritardo

Esercizio 3

In Figura 2 si riporta la rete di Petri relativa ad un'operazione di *pick-and-place* di singoli oggetti tra due stazioni di lavoro, eseguita in modo coordinato da due robot che si scambiano l'oggetto in un punto di incontro (*vedi Esercizio #3 nel compito scritto del 16 Gennaio 2014*).

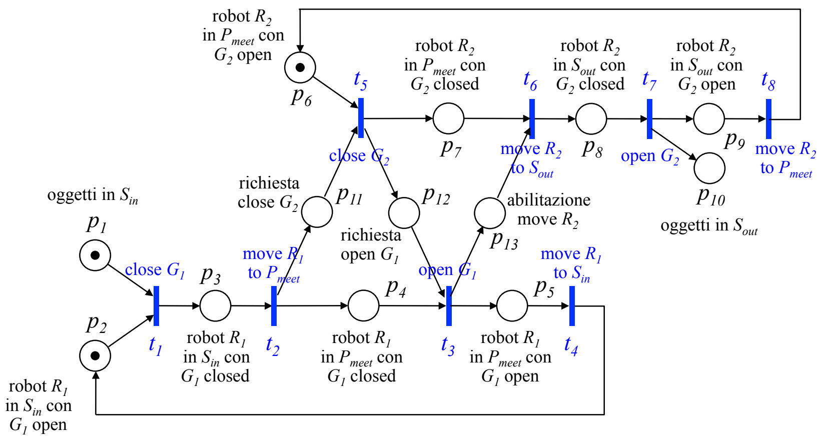

Figura 2: Rete di Petri relativa ad un'operazione di *pick-and-place* coordinata con due robot

- A partire dalla matrice di incidenza $\boldsymbol{C}$ della rete, si determini la struttura di tutti i $T-$invarianti e di tutti i $P-$invarianti (in entrambi i casi, aventi solo elementi interi). Si fornisca un'interpretazione dei $P-$invarianti trovati, in termini delle proprietà di conservatività (eventualmente di sottoparti della rete) e di limitatezza.
- Assumendo che siano sempre presenti oggetti nella stazione di ingresso e che non si voglia modellizzare il conteggio degli oggetti depositati nella stazione di uscita, è possibile eliminare i posti p_1 e p_{10} con i relativi archi (entrante in t_1 e uscente da t_7). Per la rete di Petri così ottenuta, come si modifica l'analisi delle proprietà del punto precedente? In particolare, verificare se la marcatura iniziale con un token presente solo nei posti p_2 e p_6 sia reversibile o meno, motivando la risposta.

[180 minuti; libri aperti]

Soluzioni

13 Febbraio 2014

Esercizio 1

La verifica di non inammissibilità si effettua calcolando il fattore di utilizzazione dei task periodici hard real time:

$$U = \frac{1}{4} + \frac{3}{16} + \frac{1}{8} + \frac{8}{32} = \frac{26}{32} = 0.8125 < 1.$$

Essendo verificata la condizione necessaria, controlliamo se sussista almeno una condizione sufficiente. Ad esempio, calcoliamo

$$U_{lsm} = n\left(2^{1/n} - 1\right) = 4\left(2^{1/4} - 1\right) \simeq 0.7568.$$

Dato che $U > U_{lsm}$, questa condizione sufficiente non è verificata. Si può però notare che i quattro task sono legati tra loro da relazioni armoniche e che questa costituisce un'altra condizione sufficiente per la risolubilità del problema posto. La soluzione dello scheduling RMPO con servizio in background FIFO è illustrata in Fig. 3. Da questa si evince che il task soft real time può essere portato a termine entro la deadline assoluta.

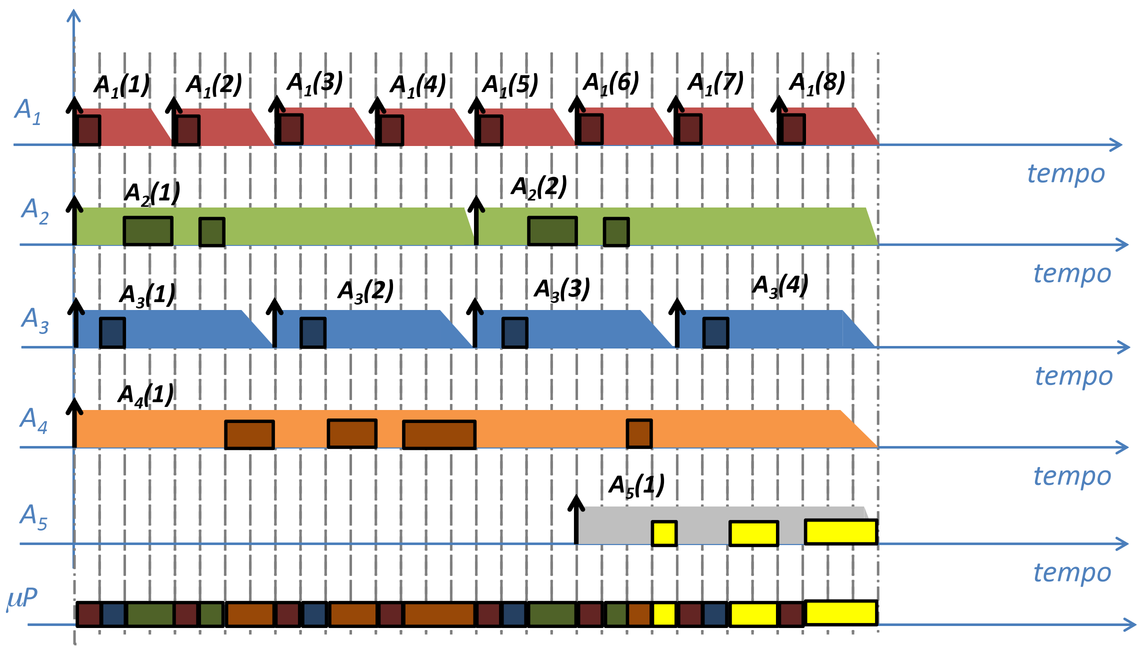

Figura 3: Scheduling RMPO con servizio in background FIFO

Esercizio 2

Domanda 1. L'accesso al bus di comunicazione verrà assegnato al nodo 1, come si evince dal diagramma temporale della contesa in Fig. 4.

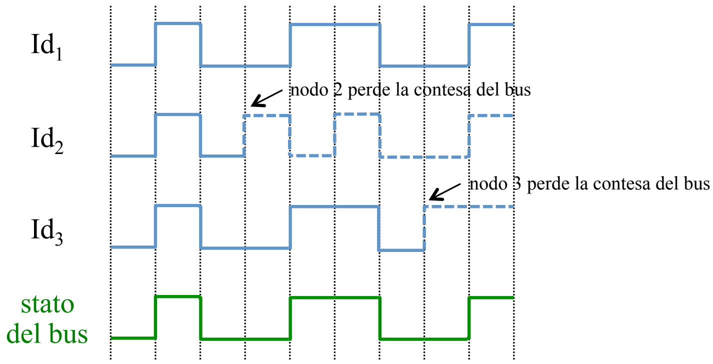

Figura 4: Diagramma temporale della contesa gestita con protocollo CSMA-CR (*Wired AND)*

Domanda 2. Sulla risposta a gradino del processo (vedi Fig. 5) si possono rilevare graficamente i seguenti valori, tracciando la tangente alla risposta nel punto di massima salita, che si trova vicino alla base della curva (in questa zona è presente anche un flesso, quasi impercettibile):

$\theta \simeq 0.2$ [s]   ritardo tra applicazione del gradino e intercetta della tangente con l'asse temporale;

$\tau \simeq 1.8$ [s]   tempo tra intercette della tangente con l'asse temporale e con il valore di regime;

$K = 0.5$ guadagno statico (a regime).

Il processo è quindi approssimato dalla funzione di trasferimento

$$P(s) = K\,\frac{e^{-\theta s}}{1+\tau s} = 0.5\,\frac{e^{-0.2\,s}}{1+1.8\,s}. \tag{1}$$

Per un regolatore PI della forma

$$u(t) = K_p\left(e(t) + \frac{1}{T_i}\int_0^t e(\sigma)d\sigma\right), \qquad e(t) = y_d(t) - y(t),$$

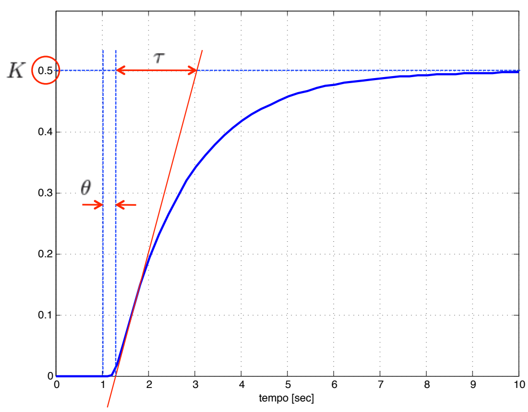

Figura 5: Rilevazione grafica dei parametri necessari per la sintonizzazione dei guadagni del regolatore secondo il metodo di Ziegler-Nichols

il primo metodo di sintonizzazione di Ziegler-Nichols prescrive dei valori[1]

$$K_p = \frac{0.9}{K}\frac{\tau}{\theta} = 16.2, \qquad T_i = 3.33\,\theta = 0.666, \qquad K_i = \frac{K_p}{T_i} = 24.3243, \tag{2}$$

dove K_i è l'effettivo guadagno del termine integrale. Poiché si vuole garantire con certezza errore nullo a regime nella risposta al gradino, il regolatore deve includere necessariamente un'azione integrale. Utilizzando i valori (2) nel regolatore PI, le prestazioni del sistema controllato risultano però ancora insoddisfacenti, in quanto la risposta è fortemente oscillatoria (il medodo di sintonizzazione si basa sul modello (1), ma evidentemente il processo originario non è ben approssimato da tale equazione). Non volendo introdurre un'azione derivativa, è prassi in tali casi dimezzare il guadagno proporzionale. La risposta del sistema ad anello chiuso e il relativo sforzo di controllo (uscita del regolatore) con i valori $K_p = 8.1$ e $K_i = K_p/T_i = 12.1622$ sono mostrati in Fig. 6.

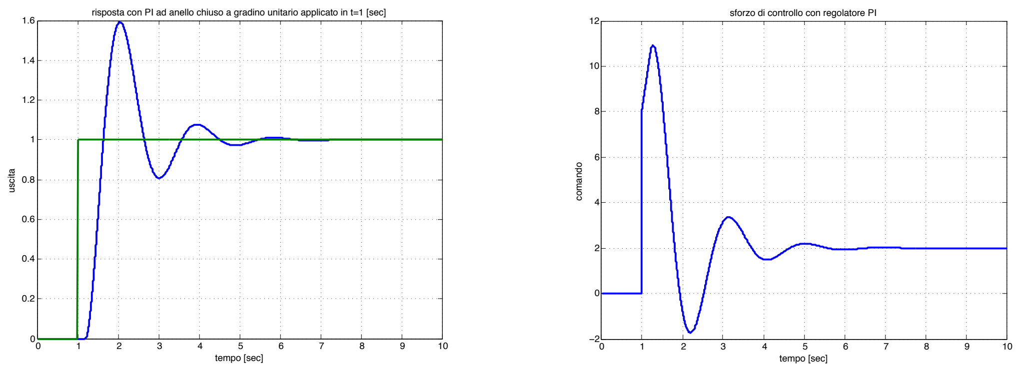

Figura 6: Risposta del sistema ad anello chiuso con il regolatore PI finale: uscita controllata (a sinistra) e sforzo di controllo (a destra)

Esercizio 3

La matrice di incidenza $\boldsymbol{C}$, di dimensioni (13×8), della rete di Petri in Fig. 2 è la seguente:

$$\boldsymbol{C} = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}. \tag{3}$$

[1] Non essendo specificato se il controllore sia implementato in modo analogico o digitale (con passo di campionamento T_c), si può porre $\theta' = \theta$ nella prima colonna della Tab. 4.2 nel libro di testo. Utilizzando un controllore PI digitale, si sarebbe posto invece $\theta' = \theta + (T_c/2)$, con T_c sufficientemente *piccolo* (al limite trascurabile, come nella soluzione presentata) in modo da poter catturare nei campioni dell'uscita l'effetto del ritardo θ.

E' semplice verificare (ad esempio con la funzione `rank` in Matlab) che la matrice $\boldsymbol{C}$ ha rango massimo (pari a 8, il numero delle sue colonne) e che quindi l'equazione che definisce i $T-$invarianti

$$\boldsymbol{C}\,\boldsymbol{\eta} = \boldsymbol{0}$$

non ha altra soluzione al di fuori di quella banale ($\boldsymbol{\eta} = \boldsymbol{0}$). Ne segue che tutte le marcature iniziali risultano essere *non reversibili*: in altri termini, non esistono sequenze di scatti (non vuote) che possano replicare almeno una marcatura iniziale.

Viceversa, a partire dall'equazione che definisce i $P-$invarianti

$$\boldsymbol{C}^T \boldsymbol{\gamma} = \boldsymbol{0}, \tag{4}$$

la dimensione dello spazio nullo della matrice $\boldsymbol{C}^T$ è pari a 5 (= # colonne di $\boldsymbol{C}^T$ − rango di $\boldsymbol{C}^T$). Occorre e basta quindi trovare cinque $P-$invarianti (con elementi nel dominio $\mathbb{Z}$ degli interi) che siano linearmente indipendenti per avere una base con la quale esprimere tutto l'insieme di interesse[2].

Dall'analisi dell'evoluzione della rete, è semplice notare che un primo $P-$invariante (in forma canonica) è dato da

$$\boldsymbol{\gamma}_0^T = \begin{pmatrix} 1 & 1 & 2 & 1 & 1 & 1 & 1 & 2 & 1 & 1 & 1 & 1 & 1 \end{pmatrix},$$

che verifica ovviamente la (4). Tale vettore è stato individuato raddoppiando i pesi di quei posti che si trovano da soli in uscita ad una transizione che preleva allo scatto due token in ingresso (ossia p_3 e p_8). Poiché il vettore $\boldsymbol{\gamma}_0$ è non negativo e il suo supporto è l'intero insieme dei posti $\mathcal{P}$, ne segue che la rete è *conservativa* e *limitata*.

Gli altri $P-$invarianti si determinano individuando sottoinsiemi di posti dove il numero totale di token si conserva e ai quali si può facilmente associare un significato fisico. Per il *ciclo operativo del robot* R_1 si ha

$$\boldsymbol{\gamma}_1^T = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

che rappresenta il fatto che il token-risorsa robot R_1 si propaga e si conserva sempre all'interno del ciclo di posti $\{p_2, p_3, p_4, p_5\}$ (in basso nella rete di Fig. 2). In modo analogo, per il *ciclo operativo del robot* R_2 si ha

$$\boldsymbol{\gamma}_2^T = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

con il token-risorsa robot R_2 che si propaga e si conserva nel ciclo di posti $\{p_6, p_7, p_8, p_9\}$. Un altro $P-$invariante è legato alla conservazione di un token nel ciclo di posti $\{p_2, p_3, p_{11}, p_{12}, p_5\}$

$$\boldsymbol{\gamma}_3^T = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix},$$

che può interpretarsi come relativo all'attività di *sincronizzazione del robot* R_1 (con il robot R_2). In modo speculare, la conservazione di un token nel ciclo di posti $\{p_6, p_{12}, p_{13}, p_8, p_9\}$

$$\boldsymbol{\gamma}_4^T = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$

[2] Ai fini dell'analisi che segue, l'istruzione `null` di Matlab, che fornisce una base per lo spazio nullo di una generica matrice, non è immediatamente utilizzabile. Nel caso presente, i cinque vettori forniti dalla `null(C')` (in notazione Matlab, `C'` = $\boldsymbol{C}^T$) presentano infatti componenti non intere e di segno misto (positivo o negativo), risultando di scarsa utilità (anche se, ovviamente, ciascuno dei vettori $\boldsymbol{\gamma}_i \in \mathbb{Z}$, $i = 0, 1, \ldots, 4$, è esprimibile come combinazione lineare, con coefficienti in $\mathbb{R}$, della base di vettori fornita dalla `null(C')`; i coefficienti di tali combinazioni lineari si trovano dalla `X=linsolve(null(C'),Gamma)`, con `Gamma` = $\boldsymbol{\Gamma}$). Questo tipo di situazione è piuttosto generale, in quanto le basi dei sottospazi calcolate da Matlab sono il risultato di una procedura numerica di ortogonalizzazione e normalizzazione. In altri termini, non viene sfruttato il fatto che la matrice di incidenza abbia sempre e solo elementi in $\mathbb{Z}$ (interi, con segno).

rappresenta l'attività di *sincronizzazione del robot* R_2 (con il robot R_1).

Si noti che i quattro $P-$invarianti $\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_4$ sono tutti non negativi, canonici e a supporto *minimo*. I cinque $P-$invarianti così trovati sono effettivamente tutti linearmente indipendenti. E' infatti facile verificare (ancora con la `rank` di Matlab) che

$$\boldsymbol{\Gamma} = \begin{pmatrix} \boldsymbol{\gamma}_0 & \boldsymbol{\gamma}_1 & \boldsymbol{\gamma}_2 & \boldsymbol{\gamma}_3 & \boldsymbol{\gamma}_4 \end{pmatrix} \qquad \Rightarrow \qquad \text{rango}\, \boldsymbol{\Gamma} = 5.$$

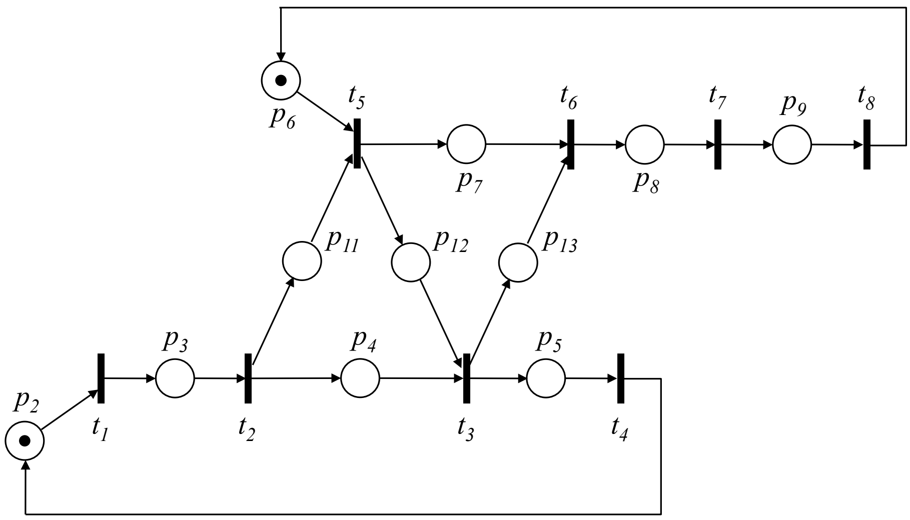

Figura 7: Rete di Petri modificata eliminando un posto in ingresso (p_1) e uno in uscita (p_{10})

Nella seconda parte dell'esercizio, si eliminano dalla rete di Petri originaria i posti p_1 (con il relativo token iniziale) e p_{10}. Per chiarezza la rete così modificata è mostrata in Fig. 7. La relativa matrice di incidenza $\boldsymbol{C}_m$, di dimensioni (11×8), si ottiene cancellando le due righe corrispondenti dalla matrice $\boldsymbol{C}$ in (3):

$$\boldsymbol{C}_m = \begin{pmatrix} -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}. \tag{5}$$

E' immediato verificare che la nuova matrice di incidenza $\boldsymbol{C}_m$ ha rango pari a 7, inferiore di un'unità rispetto al numero delle sue colonne. L'equazione che definisce i $T-$invarianti

$$\boldsymbol{C}_m\, \boldsymbol{\eta}_m = \boldsymbol{0}$$

ha pertanto un'infinità (semplice) di soluzioni nella forma $\boldsymbol{\eta}_m = \alpha\, \boldsymbol{\eta}_0$, dove

$$\boldsymbol{\eta}_0^T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{6}$$

e $\alpha \in \mathbb{R}$ (nell'analisi delle proprietà strutturali della rete interessano solo le soluzioni con $\alpha \in \mathbb{N}$). Il $T-$invariante canonico in (6) corrisponde ad uno e un solo scatto per ciascuna transizione. Tale *vettore delle occorrenze* è effettivamente associato ad una *sequenza di scatti* $\mathcal{S}$ ammissibile a partire dalla marcatura iniziale (mostrata in Fig. 7)

$$\begin{aligned} \boldsymbol{x}_0^T &= \begin{pmatrix} x(p_2) & x(p_3) & x(p_4) & x(p_5) & x(p_6) & x(p_7) & x(p_8) & x(p_9) & x(p_{11}) & x(p_{12}) & x(p_{13}) \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

Pertanto la marcatura iniziale $\boldsymbol{x}_0$ della rete di Petri modificata è *reversibile.*

Dall'equazione che definisce i $P-$invarianti

$$\boldsymbol{C}_m^T \boldsymbol{\gamma}_m = \boldsymbol{0}, \tag{7}$$

la dimensione dello spazio nullo della matrice $\boldsymbol{C}_m^T$ è ora pari a 4 (= # colonne di $\boldsymbol{C}_m^T-$rango di $\boldsymbol{C}_m^T$). E' facile riconoscere che quattro nuovi $P-$invarianti che siano linearmente indipendenti si ottengono direttamente da quattro dei precedenti cinque $P-$invarianti della rete completa (escludendo $\boldsymbol{\gamma}_0$), eliminando da essi gli elementi corrispondenti ai due posti rimossi (in posizione 1 e 10):

$$\begin{aligned} \boldsymbol{\gamma}_{m,1}^T &= \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\ \boldsymbol{\gamma}_{m,2}^T &= \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \\ \boldsymbol{\gamma}_{m,3}^T &= \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \\ \boldsymbol{\gamma}_{m,4}^T &= \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}. \end{aligned}$$

E' immediato verificare (ancora con la `rank` di Matlab) che

$$\boldsymbol{\Gamma}_m = \begin{pmatrix} \boldsymbol{\gamma}_{m,1} & \boldsymbol{\gamma}_{m,2} & \boldsymbol{\gamma}_{m,3} & \boldsymbol{\gamma}_{m,4} \end{pmatrix} \qquad \Rightarrow \qquad \text{rango } \boldsymbol{\Gamma}_m = 4.$$

Il significato fisico di tali $P-$invarianti è analogo al caso precedente, con sottoinsiemi ciclici di posti dove il numero totale di token si conserva. I quattro $P-$invarianti $\boldsymbol{\gamma}_{m,1}, \ldots, \boldsymbol{\gamma}_{m,4}$ sono tutti non negativi, canonici e a supporto *minimo.* Inoltre l'unione dei loro supporti ricopre l'intero insieme dei posti $\mathcal{P}_m$ della rete modificata. Quindi anche tale rete è *conservativa* e *limitata.*

* * * * *