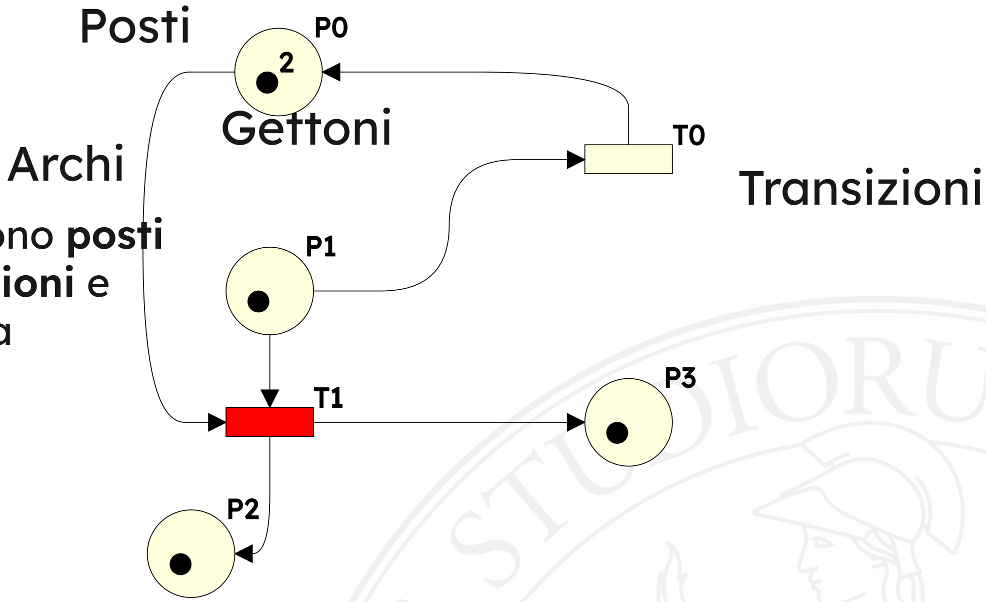
UNIVERSITÀ DEGLI STUDI
DI MILANO

Specifiche formali

Reti di Petri

- Sono in parte simili a macchine a stati finiti, ma nascono specificatamente per descrivere sistemi concorrenti
  - cambiano sia il concetto di stato che di transizione…
    - lo **stato** non è più visto a livello di sistema ma come composizione di tanti stati parziali
    - le **transizioni** (promosse a nodi invece di semplici archi) non operano più quindi su uno stato globale ma si limitano a variarne una parte
- Informazioni on line su Petri Nets: http://www.informatik.uni-hamburg.de/TGI/PetriNets/

Informalmente



Connettono **posti** a **transizioni** e viceversa

Definizione di Rete di Petri

Una Rete di Petri è una 5-upla $[P, T; F, W, M_0]$

P	l'insieme dei posti	
T	l'insieme delle transizioni	
F	relazione di flusso	$F \subseteq (P \times T) \cup (T \times P)$
W	la funzione peso	$W : F \to \mathbb{N}^+$
M_0	la funzione marcatura (iniziale)	$M_0 : P \to \mathbb{N}$

$pre(a) = \{d \in (P \cup T) \mid \langle d, a\rangle \in F\}$ *//preset*

$post(a) = \{d \in (P \cup T) \mid \langle a, d\rangle \in F\}$ *//postset*

Comportamento dinamico

$t \in T$ è abilitata in M se e solo se $\forall p \in pre(t) \quad M(p) \geq W(\langle p, t \rangle)$

`M[t>` t è abilitata in M

---

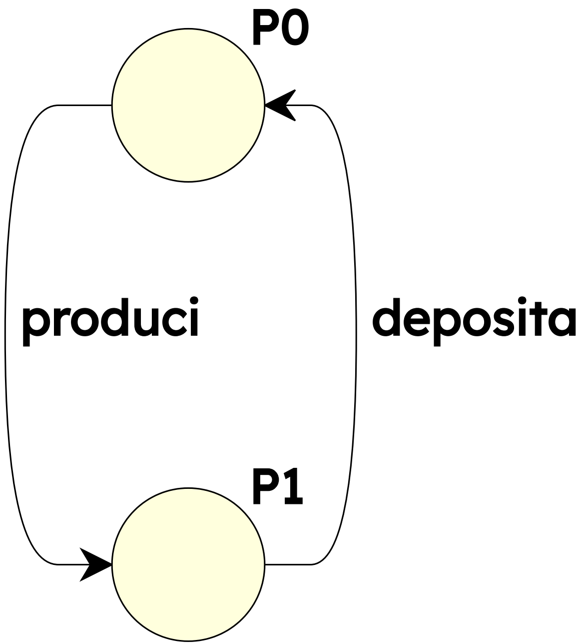
lo scatto di una transizione t in una marcatura M produce una nuova marcatura M'

`M[t>M'` lo scatto di t in M produce M'

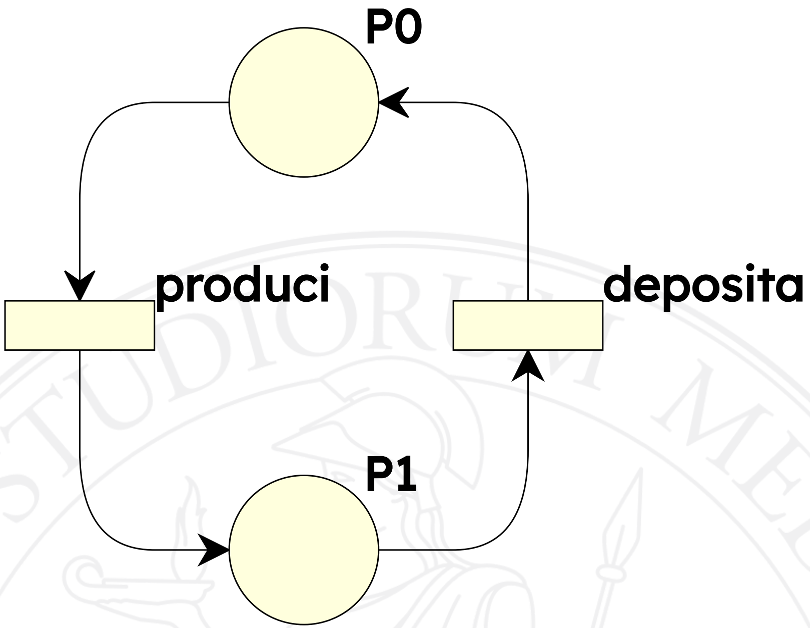
$\forall p \in pre(t) \setminus post(t)$	$M'(p) = M(p) - W(\langle p, t \rangle)$
$\forall p \in post(t) \setminus pre(t)$	$M'(p) = M(p) + W(\langle t, p \rangle)$
$\forall p \in pre(t) \cap post(t)$	$M'(p) = M(p) - W(\langle p, t \rangle) + W(\langle t, p \rangle)$
$\forall p \in P \setminus (pre(t) \cup post(t))$	$M'(p) = M(p)$

Esempio: il produttore

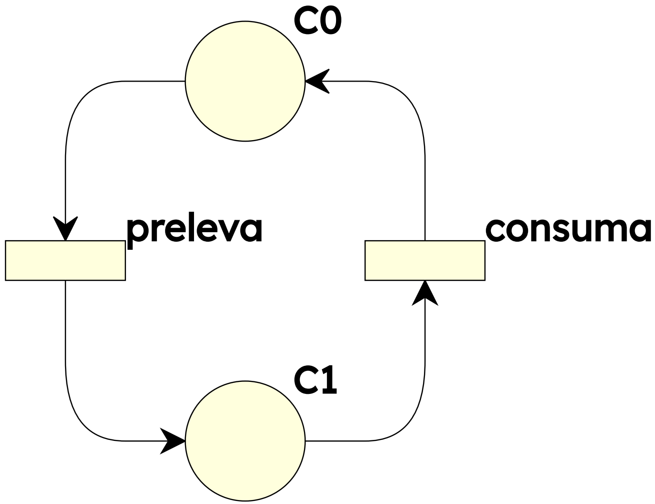
MSF

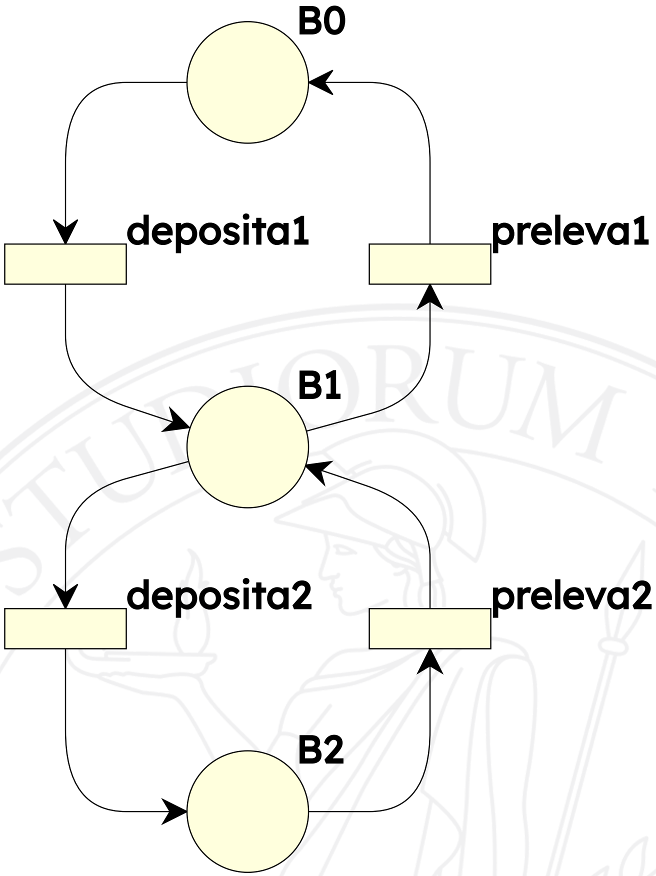


RdP

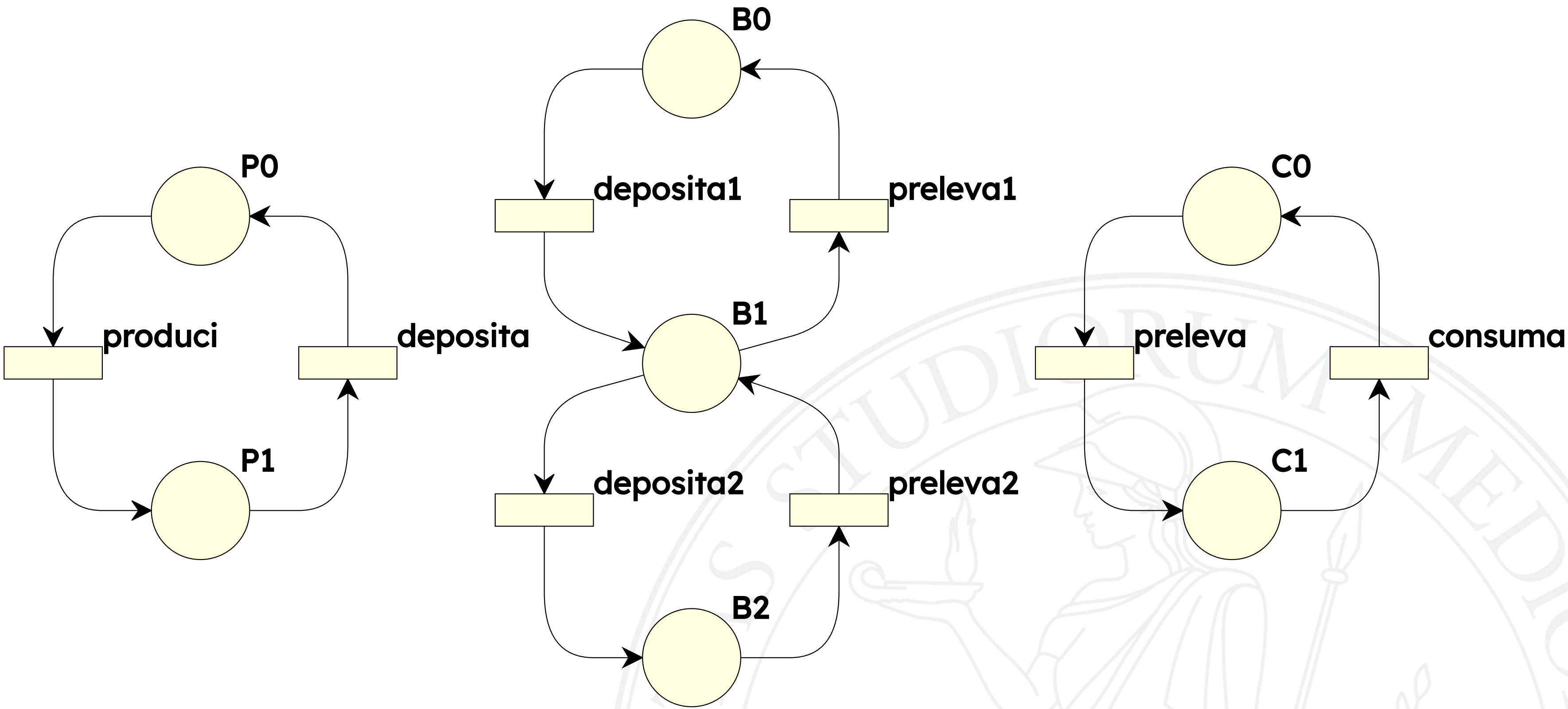


Esempio: consumatore e buffer

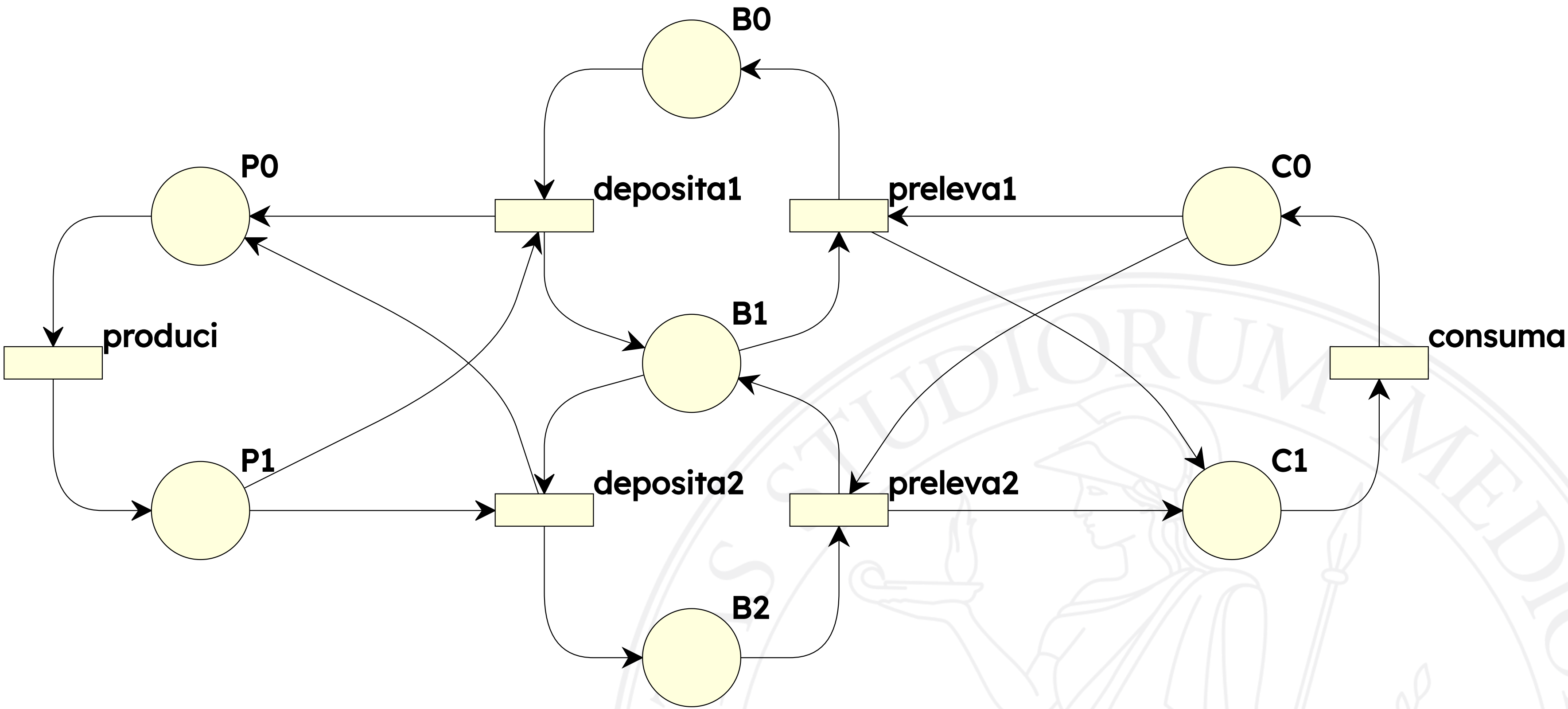




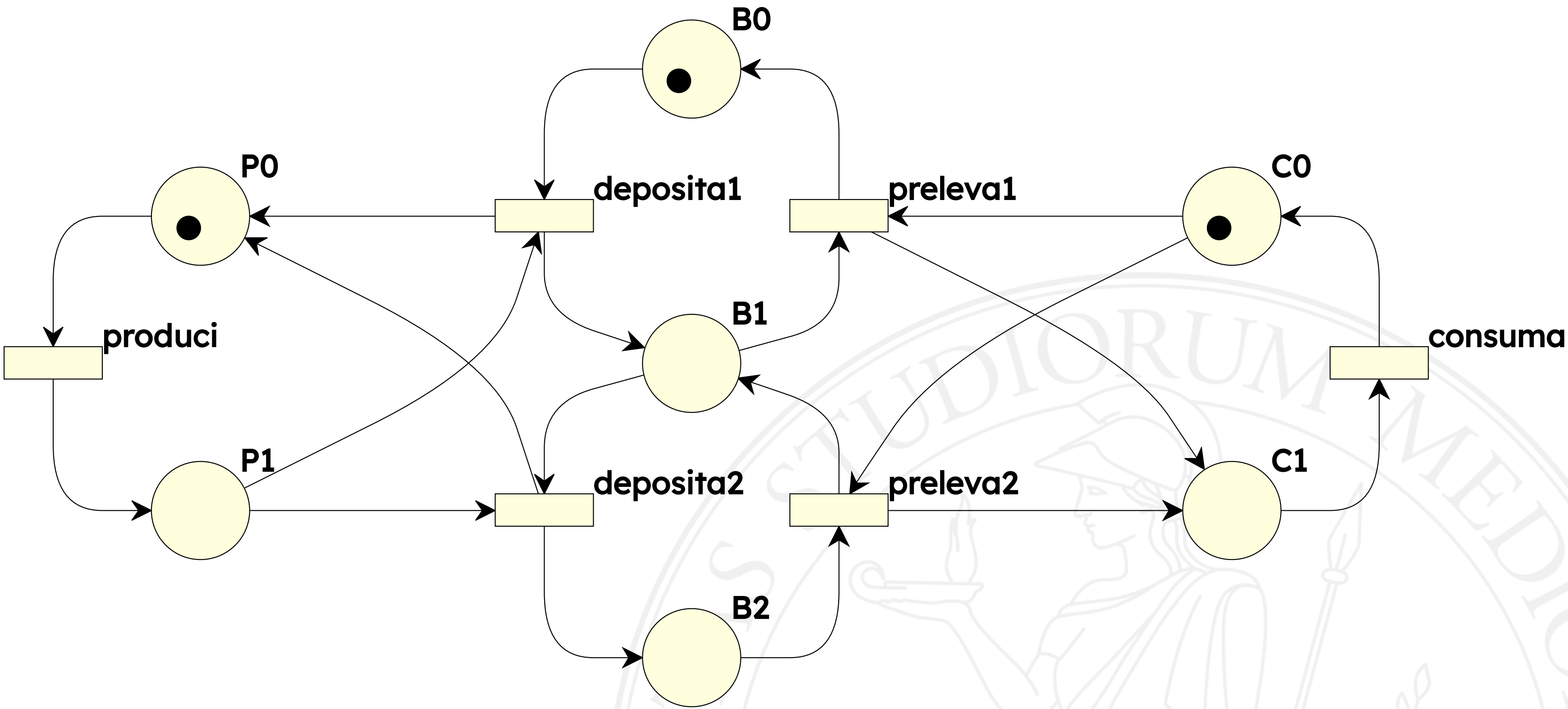
Componiamo


B0
P0
deposita1
preleva1
C0
produci
deposita
B1
preleva
consuma
P1
deposita2
preleva2
C1
B2

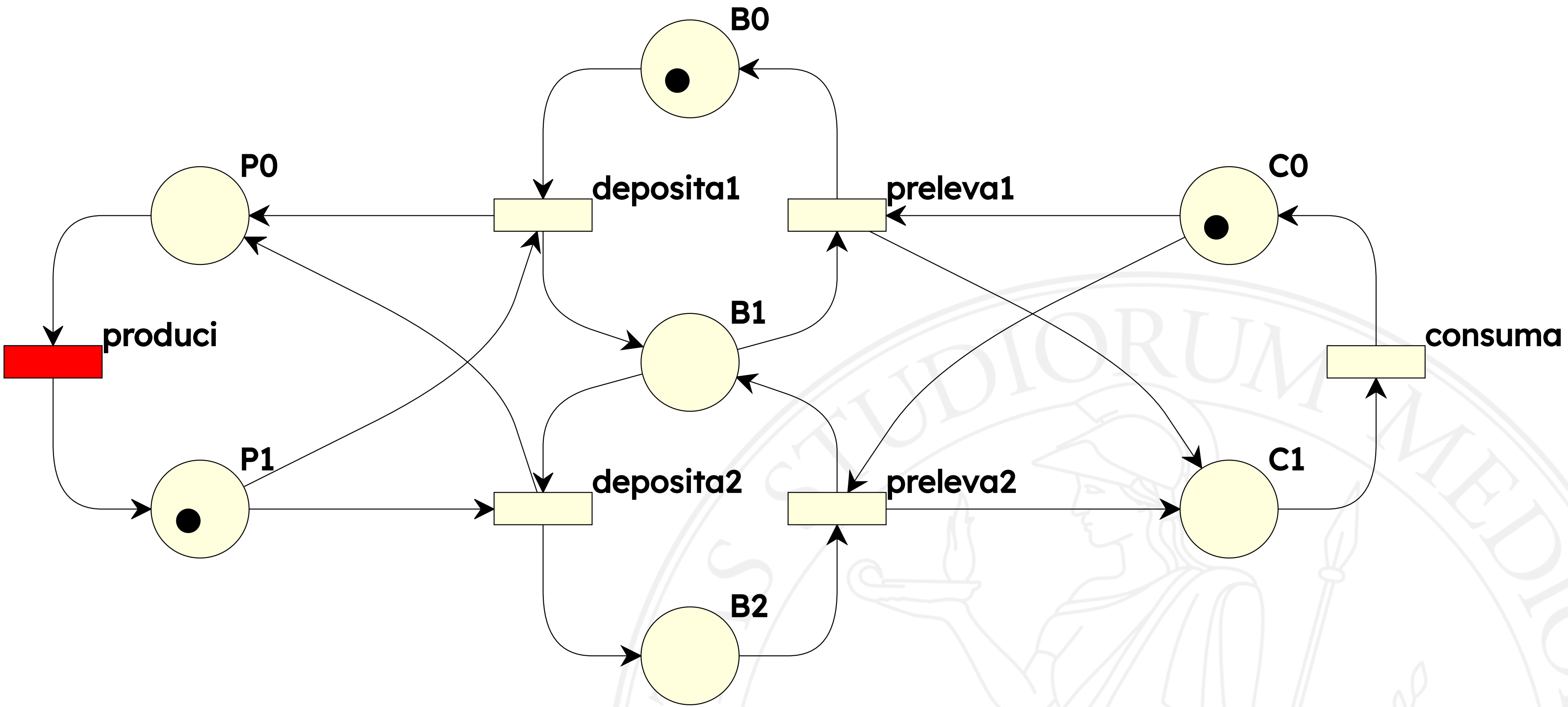
Componiamo


B0
P0
deposita1
preleva1
C0
B1
produci
consuma
P1
deposita2
preleva2
C1
B2

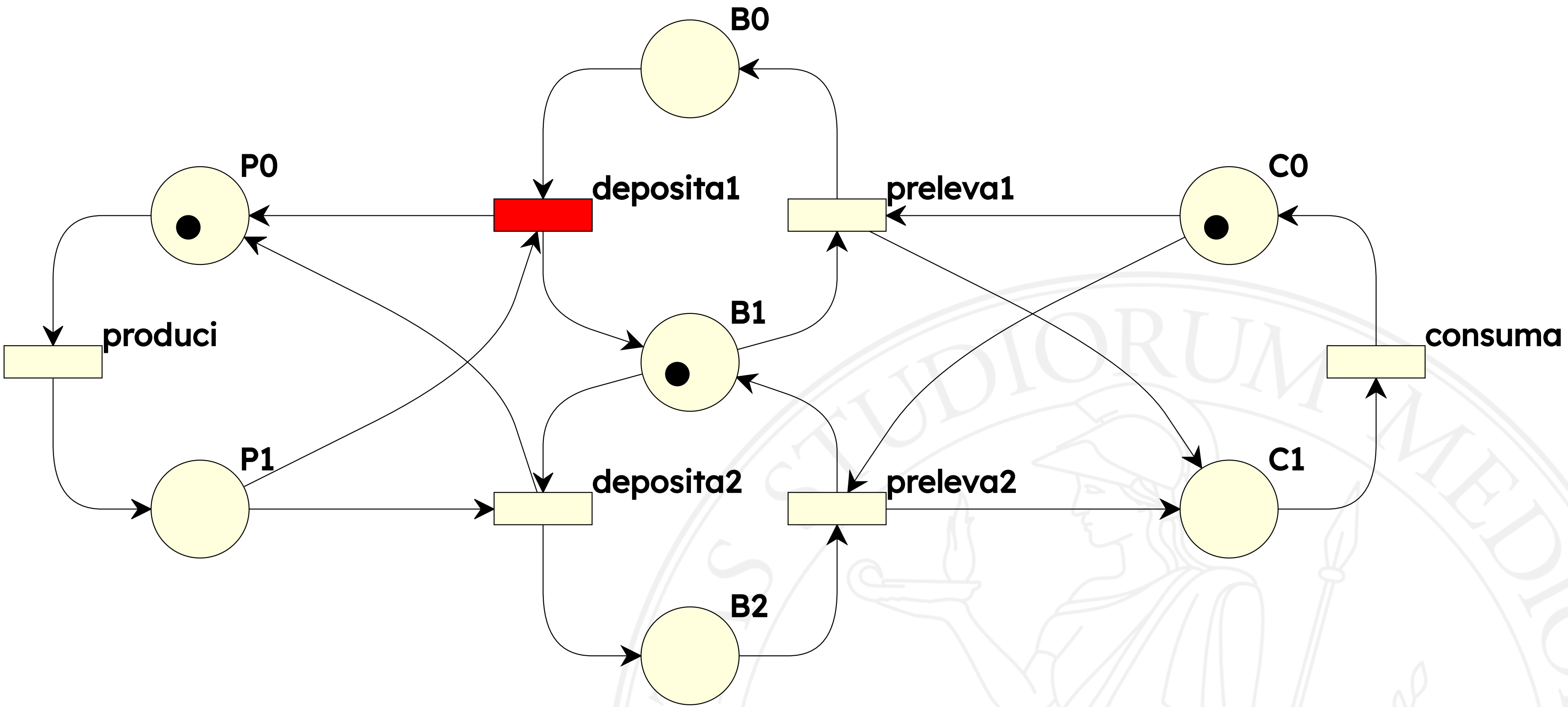
Come evolve? Abilitazioni... e scatti...



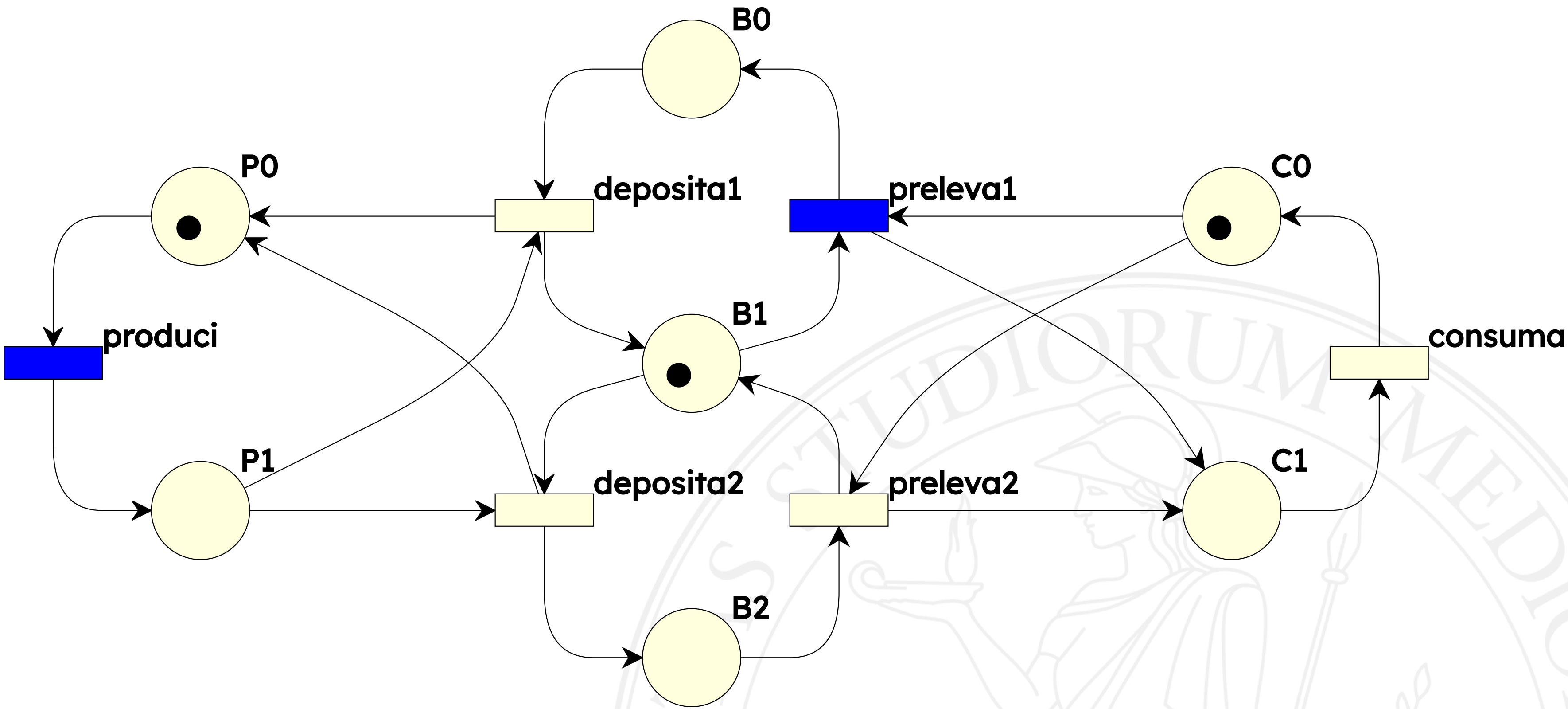
Come evolve? Abilitazioni… e scatti…



Come evolve? Abilitazioni… e scatti…



Come evolve? Abilitazioni… e scatti…



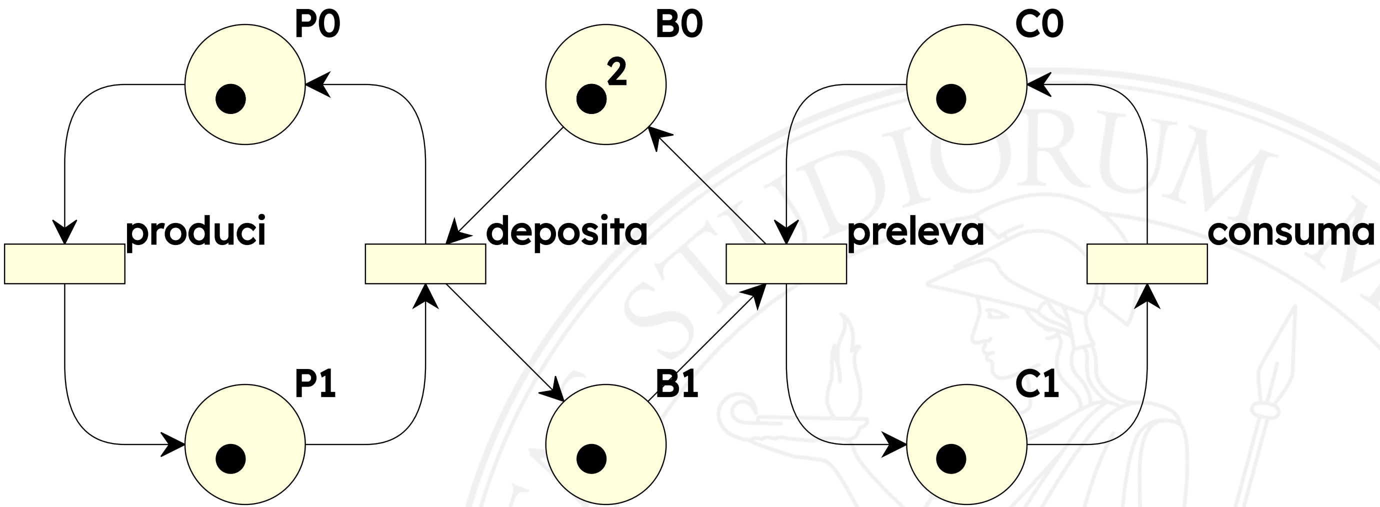
Quale scelgo?

- È un caso di non determinismo… non posso dire quale deve scattare
- tipico di sistemi concorrenti, dove non ho questo tipo di controllo globale…
- ATTENZIONE: se fosse stato necessario vuol dire semplicemente che la rete non era corretta,
  - cioè posso, con modifiche alla rete, forzare un determinato comportamento

Stiamo sfruttando redi di Petri?

- Non molto, abbiamo fatto una traduzione "automatica" dagli automi a stati finiti

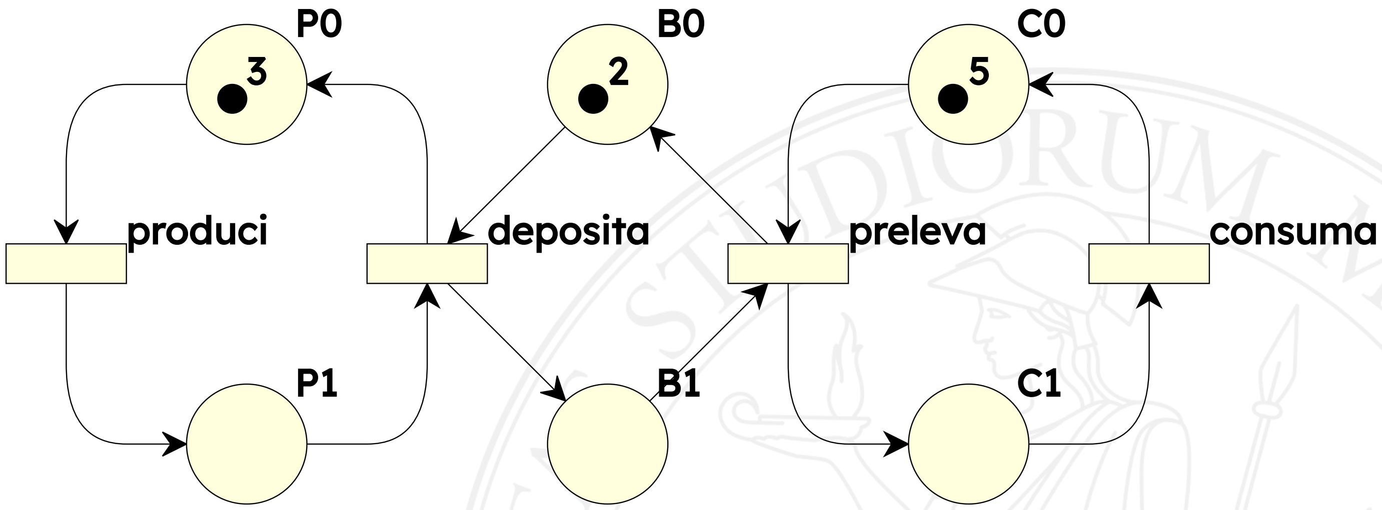
- proviamo qualche alternativa / modifica



Il numero di gettoni nel posto B0 (B1) indica quante celle vuote (piene) ci sono nel buffer

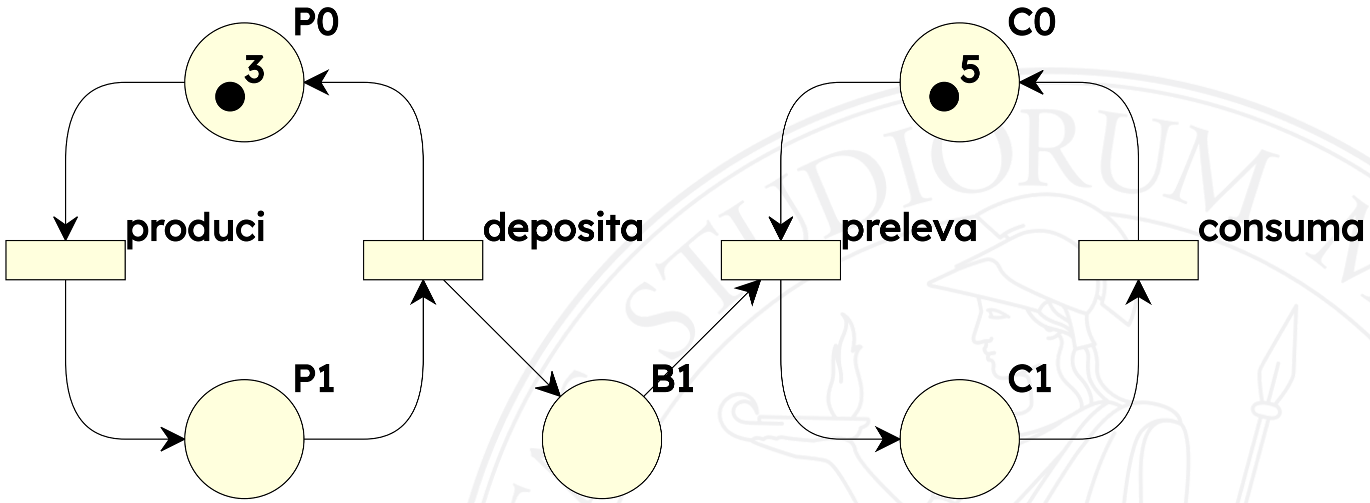
Altre modifiche

- Cosa succede se aumentiamo il numero di token in *P0* o in *C0*?



A volte "meno è di più"

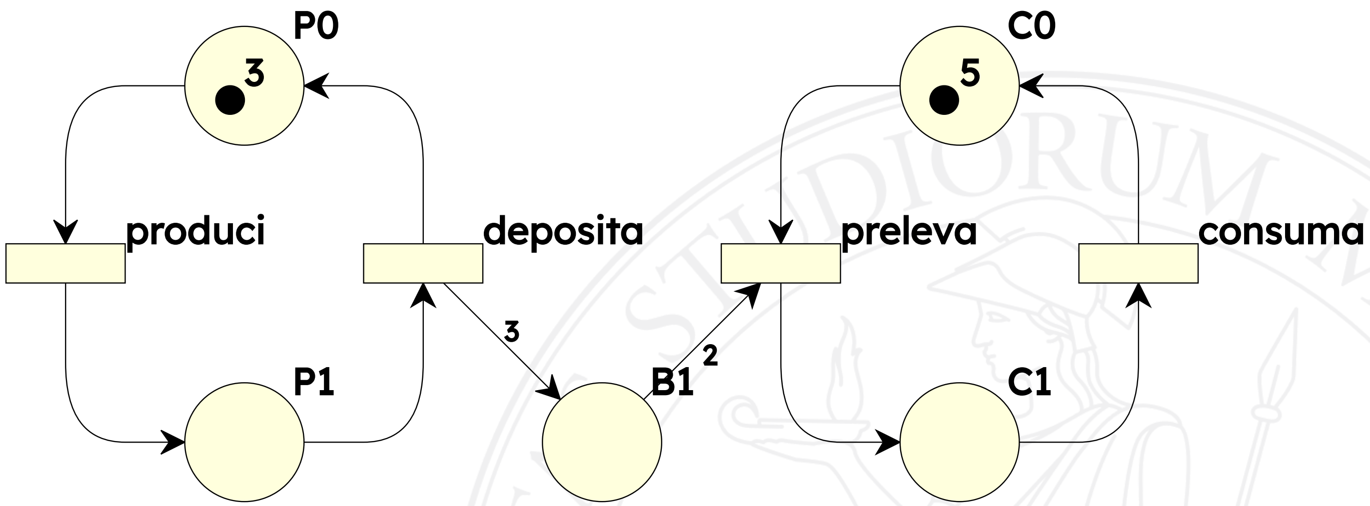
- Cosa modellava il posto *B0* ?
  - che vincoli abbiamo adesso sulla operazione di *deposita* ?



Possiamo quindi facilmente modellare un buffer infinito

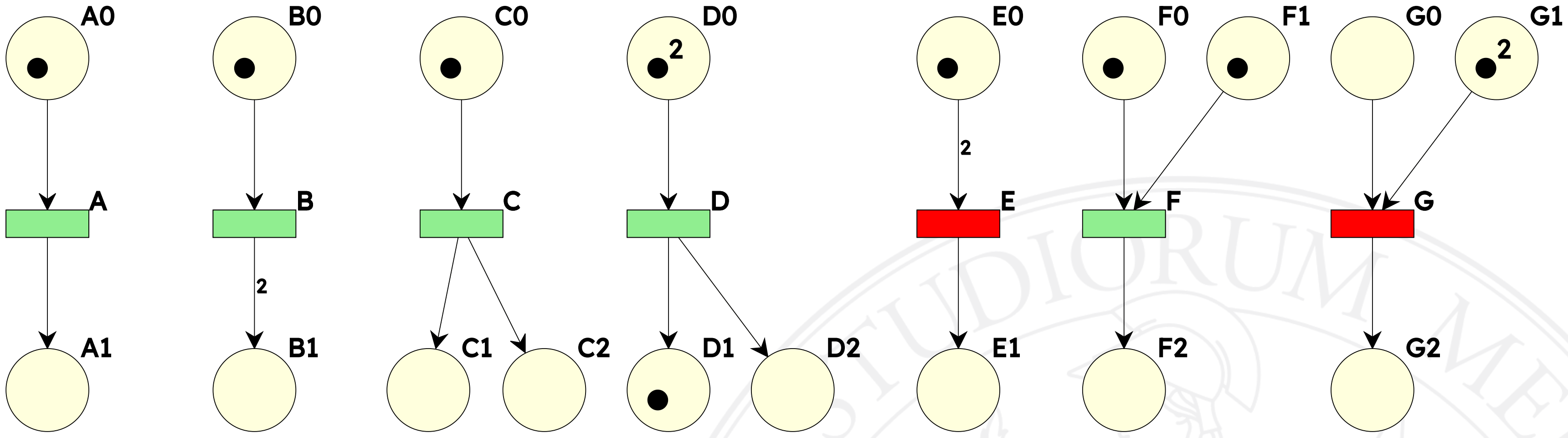
Ancora modifiche

- Cosa succede ad usare i pesi degli archi ?



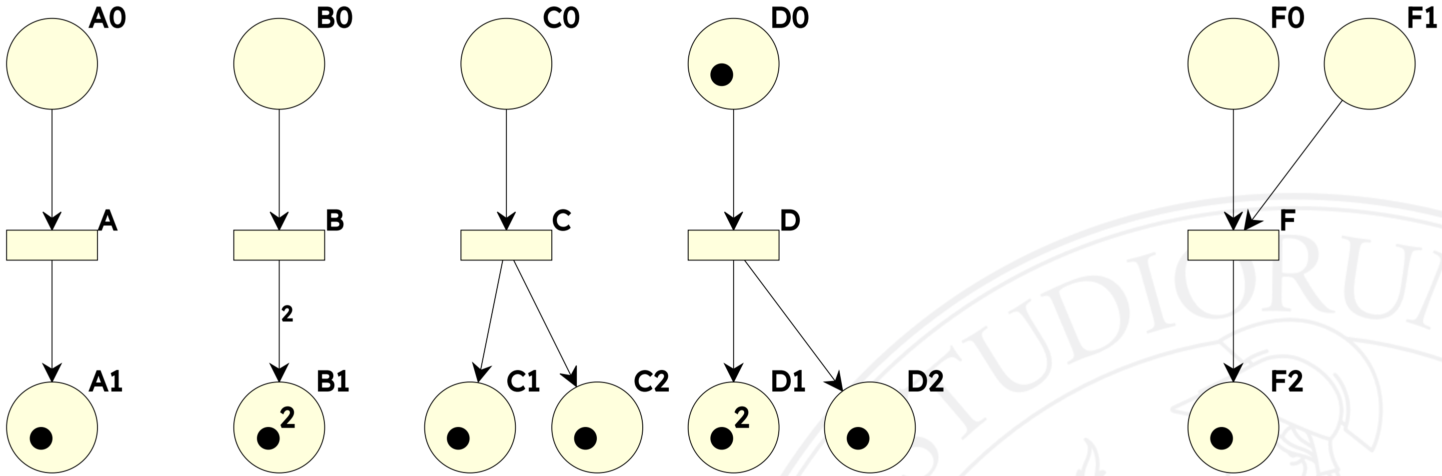
Possiamo quindi facilmente modellare un buffer infinito

Controlliamo abilitazioni



Basta ricordarsi la definizione: $\boxed{\forall p \in Pre(t) \quad M(p) \geq W(\langle p, t \rangle)}$

Controlliamo scatti



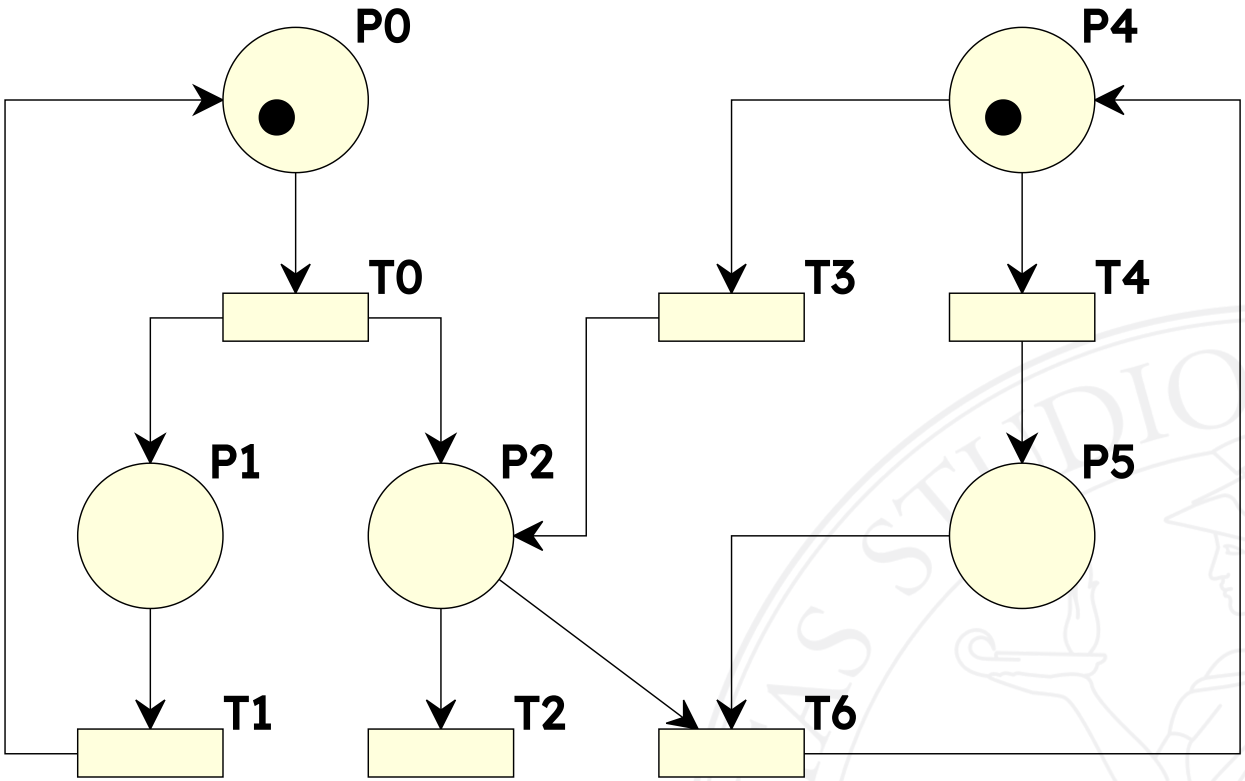
Relazioni: sequenza

- Una transizione t_1 è in sequenza con una transizione t_2 in una marcatura M se e solo se:

  $$M[t_1 > \wedge \neg M[t_2 \wedge M[t_1 t_2 >$$

  - t_1 è abilitata in M
  - t_2 non è abilitata in M
  - t_2 è abilitata nella marcatura M' prodotta dallo scatto $M[t_1 > M'$

Esempio



- T0 e T1
- T0 e T2
- T3 e T2

Relazioni: conflitto

- Due transizioni t_1 e t_2 sono in conflitto
  - strutturale se e solo se:

    $$pre(t_1) \cap pre(t_2) \neq \emptyset$$

  - effettivo in una marcatura M se e solo se:

    $$M[t_1 > \quad \wedge \quad M[t_2 > \quad \wedge$$

    $$\exists p \in pre(t_1) \cap pre(t_2) \quad M(p) < W(\langle p, t_1 \rangle) + W(\langle p, t_2 \rangle)$$

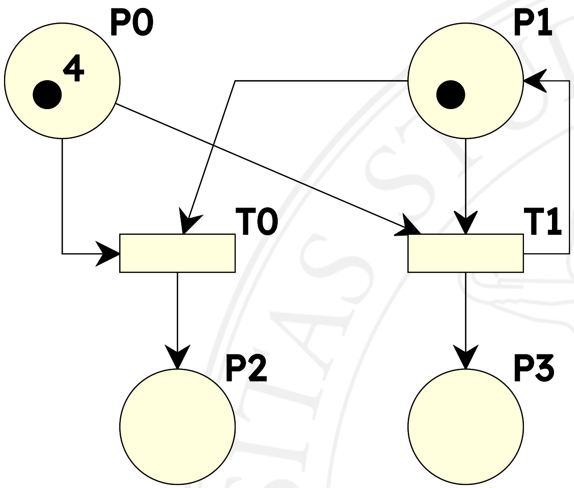
    - t_1 e t_2 sono abilitate in M
    - esiste un posto p in ingresso a entrambe le transizioni che non ha abbastanza token per farle scattare entrambe

Relazioni: conflitto (cont)

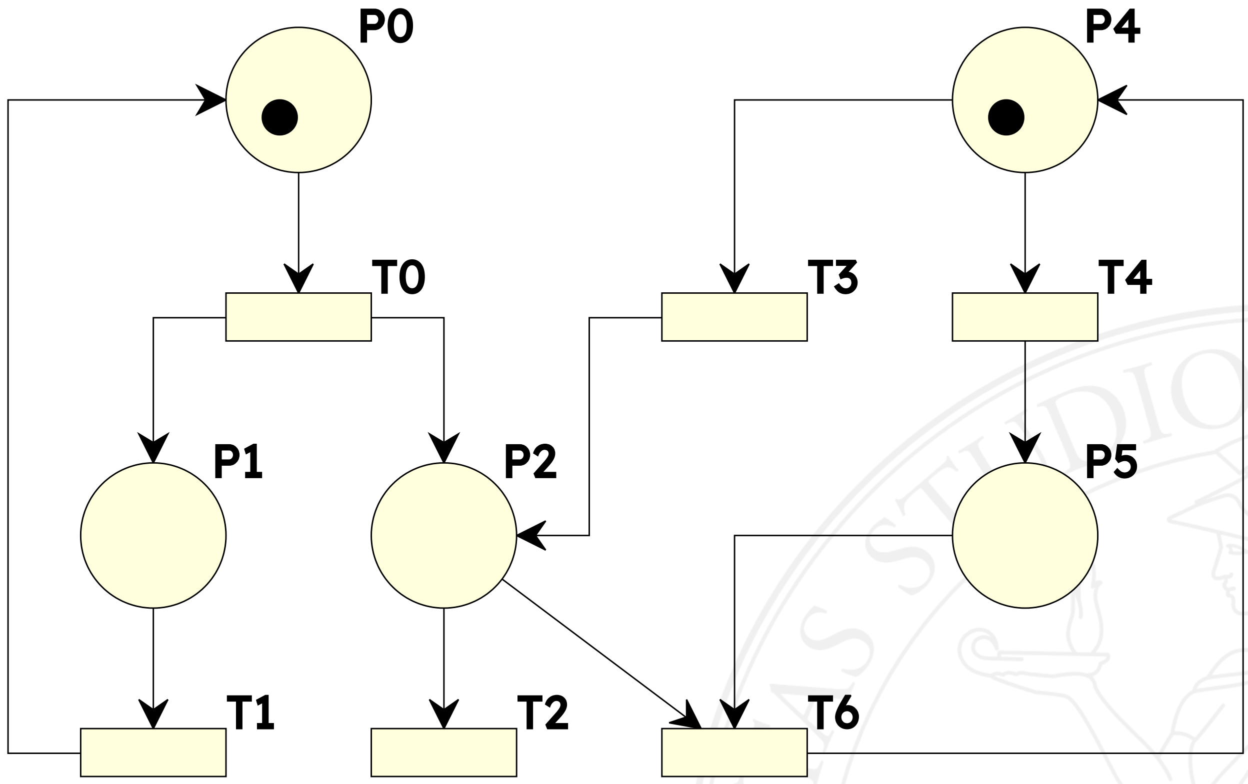
- È la stessa cosa di scrivere la seguente formula?

$$M[t_1 > \wedge M[t_2 > \wedge \neg M[t_1 t_2 >$$

  - t_1 e t_2 sono abilitate in M
  - ma $t_1 t_2$ non è sequenza ammissibile a partire da M



Esempio



- strutturale
  - T2 e T6
  - T3 e T4
- effettivo
  - T3 e T4

Relazione di concorrenza

- Due transizioni t_1 e t_2 sono in concorrenza:
  - strutturale se e solo se:

    $$pre(t_1) \cap pre(t_2) = \emptyset$$

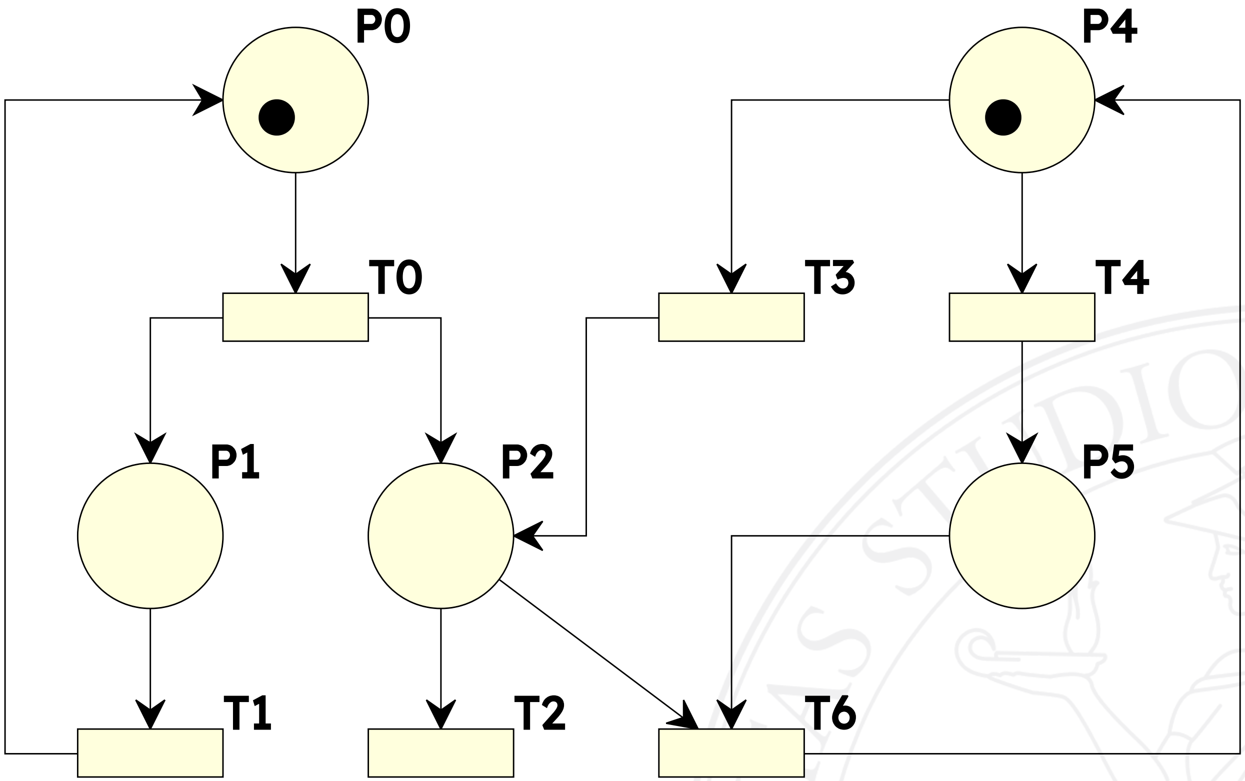
  - effettiva in una marcatura M se e solo se:

    $$M[t_1 > \quad \wedge \quad M[t_2 > \quad \wedge$$

    $$\forall p \in pre(t_1) \cap pre(t_2) \;\; M(p) \geq W(\langle p, t_1 \rangle) + W(\langle p, t_2 \rangle)$$

    - t_1 e t_2 sono abilitate in M
    - tutti i posti in ingresso a entrambe le transizoni hanno abbastanza token per farle scattare entrambe

Esempio



- strutturale:

  tutte tranne

  - T2 e T6
  - T3 e T4

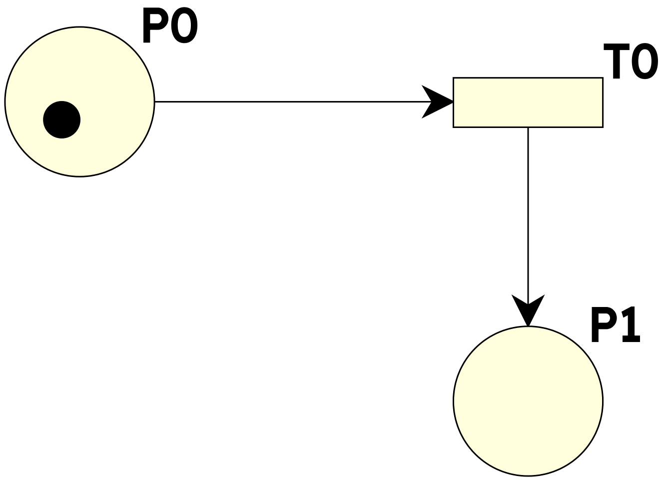
- effettivo
  - T0 e T3
  - T0 e T4

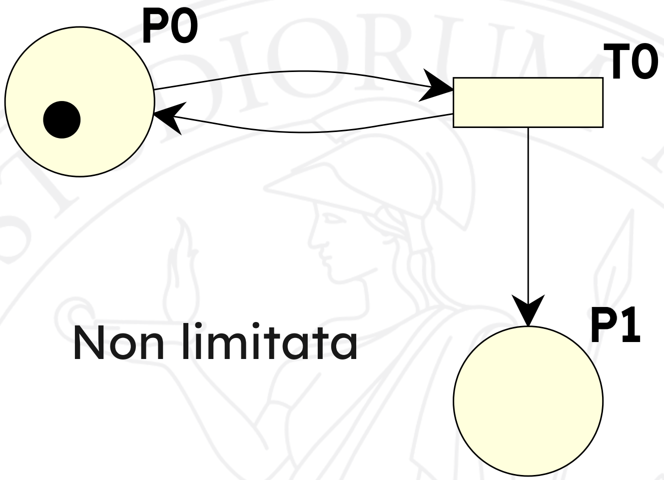
Insieme di raggiungibilità

- L'insieme di raggiungibilità di una rete a partire da una marcatura M è il più piccolo insieme di marcature tale che:
  - $M \in R(P/T, M)$
  - $(M' \in R(P/T, M)\ \wedge\ \exists t \in T\ M'[t > M'') \implies M'' \in R(P/T, M)$

Proprietà di limitatezza

- Una rete P/T con marcatura M si dice limitata se e solo se:
  - $\exists k \in \mathbb{N} \quad \forall M' \in R(P/T, M) \ \forall p \in P \quad \mathrm{M}'(p) \leq k$
- cioè se è possibile fissare un limite al numero di gettoni della rete





Non limitata

Legame tra reti di Petri e Automi a stati finiti

- Se la rete di Petri è *limitata*
- allora l'insieme di raggiungibilità è finito
- allora esiste un automa a stati finiti corrispondente che ne descrive il comportamento
  - gli stati sono le possibili marcature dell'insieme di raggiungibilità

Vitalità di una transizione

Una transizione t in una marcatura M si dice viva:

- di grado 0 (cioè **morta**):
  - non è abilitata in M e in nessuna delle marcature raggiungibili da M
    - $\forall M' \in R(P/T, M) \quad \neg M'[t>$
- di grado 1:
  - esiste almeno una marcatura raggiungibile da M in cui t è abilitata
    - $\exists M' \in R(P/T, M) \quad M'[t>$
- di grado 2:
  - per ogni numero k (quindi per k grande a piacere) esiste almeno una sequenza ammissibile da M in cui la transizione t scatta k volte
    - $\forall k \in \mathbb{N} \quad M[..\, t \,..\, t^1 .. \, t^{k-1} .. t^k >$

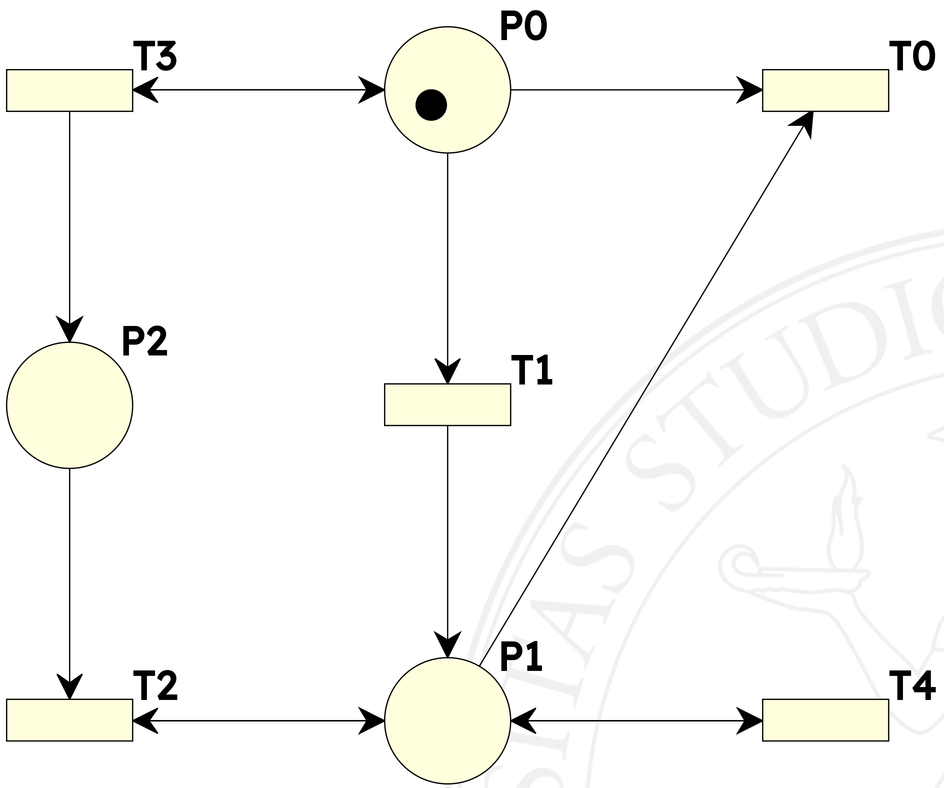
Vitalità di una transizione (cont.)

Una transizione t in una marcatura M si dice viva:

- di grado 3:
  - esiste almeno una sequenza ammissibile da M in cui la transizione t scatta infinite volte
- di grado 4 (cioè **viva**):
  - in qualunque marcatura raggiungibile da M, t non è morta
    - $\forall M' \in R(P/T, M) \quad \exists M'' \in R(P/T, M') \quad M''[t>$

Una **rete è viva** se tutte le sue transizioni sono vive

Esempio di rete con vitalità



Capacità dei posti

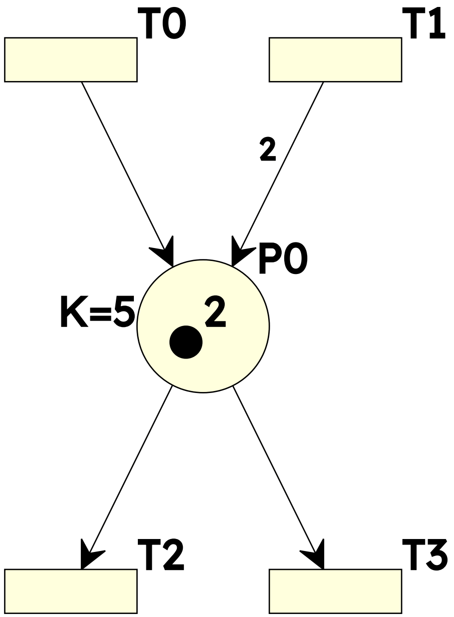
- Una possibile *estensione* delle reti di Petri consiste nel fissare un massimo numero di token ammissibili in un posto
  - si può forzare limitatezza
- Come cambia la regola di abilitazione?
- $t \in T$ è abilitata in M se e solo se:
  - $\forall p \in pre(t) \quad M(p) >= W(\langle p, t \rangle)$
  - $\forall p \in post(t) \quad M(p) + W(\langle t, p \rangle) <= K(p)$

È una estensione propria? Aumenta potenza?

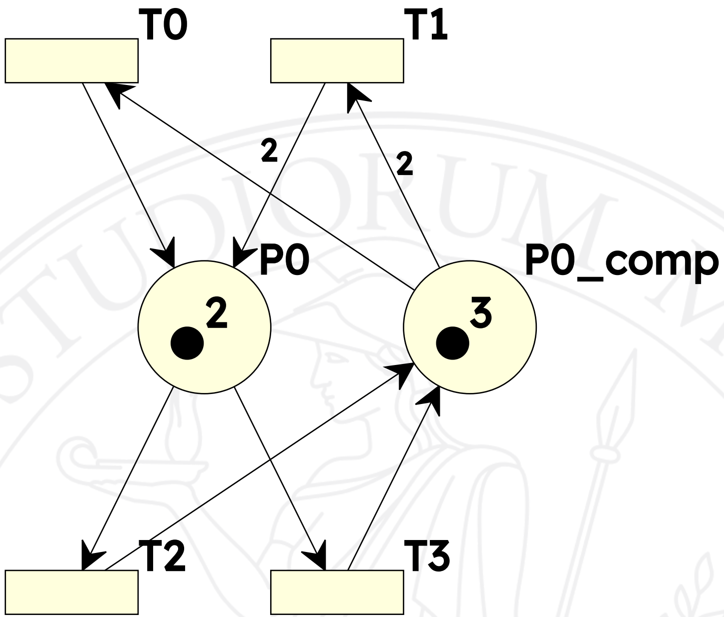
Posto complementare

- dato un posto p si dice che un posto p_c è il suo complementare se e solo se:
  - $\forall t \in post(p) \quad \exists \langle t, p_c \rangle \in F \quad W(\langle t, p_c \rangle) = W(\langle p, t \rangle)$
  - $\forall t \in pre(p) \quad \exists \langle p_c, t \rangle \in F \quad W(\langle p_c, t \rangle) = W(\langle t, p \rangle)$

Simulazione capacità dei posti



Creo un posto **complementare**

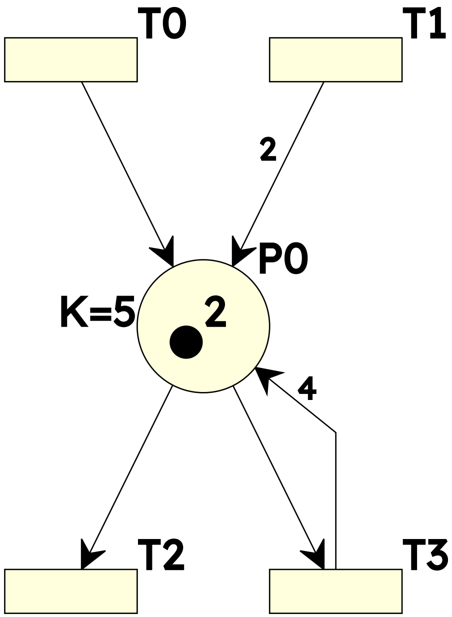


Problemi se reti non *pure*

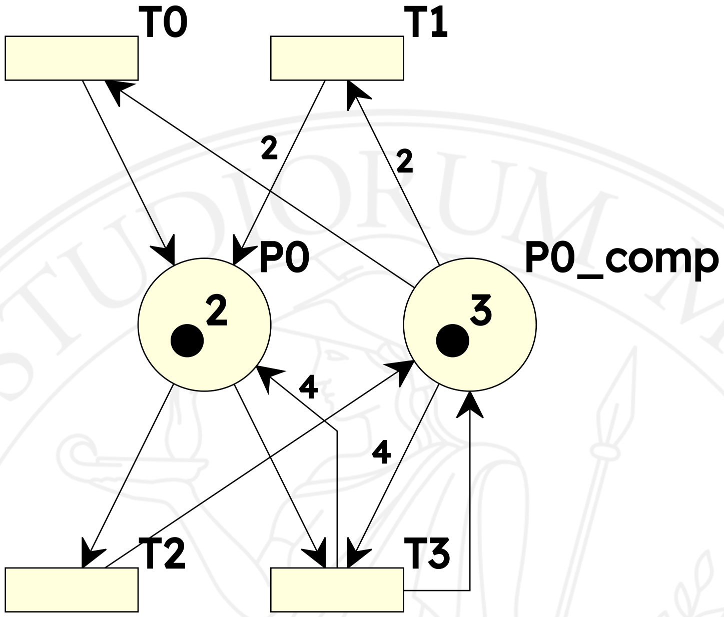
- Un rete è detta *pura* se: $\forall t \in T \quad pre(t) \cap post(t) = \emptyset$
- Cosa succede se non è vero?
- Come cambia la regola di abilitazione?
  - $\forall p \in pre(t) \quad M(p) >= W(\langle p, t \rangle)$
  - $\forall p \in post(t) \setminus pre(t) \quad M(p) + W(\langle t, p \rangle) <= K(p)$
  - $\forall p \in post(t) \cap pre(t) \quad M(p) + W(\langle t, p \rangle) - W(\langle p, t \rangle) <= K(p)$

UNIVERSITÀ DEGLI STUDI DI MILANO

In caso di rete non pura

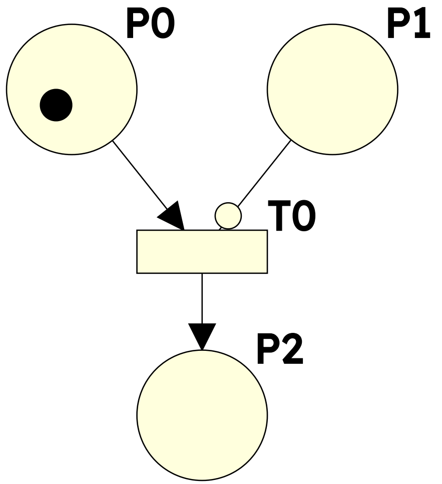


Creo il posto **complementare**

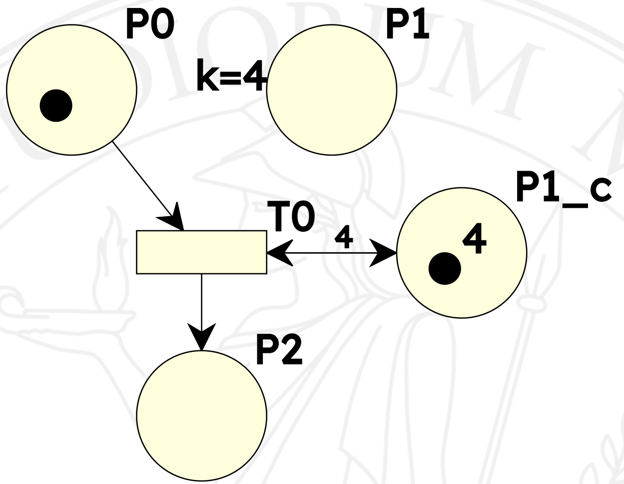


Archi inibitori

- Permettono di dire che non deve essere presente alcun token affinché la transizione sia abilitata

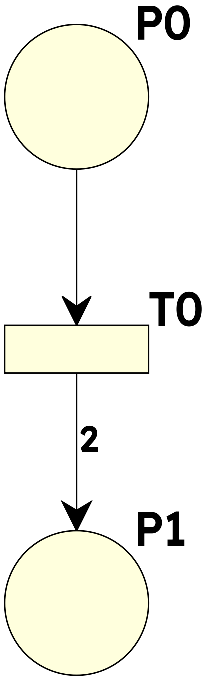


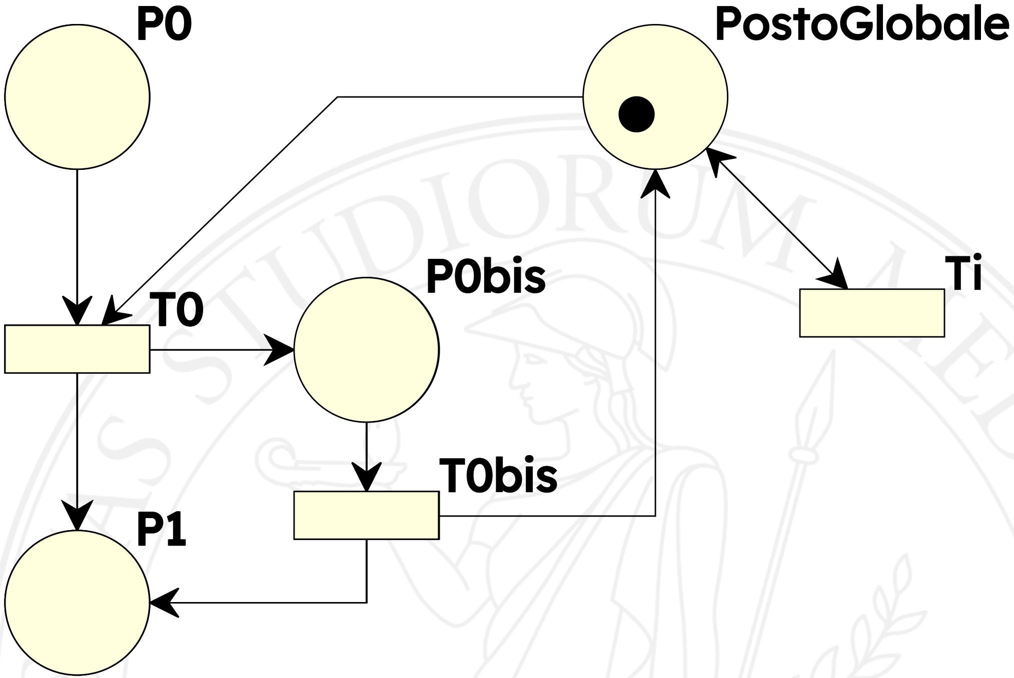
- in caso di rete limitata non cambia potenza. Perché?



Elminazione pesi archi ?

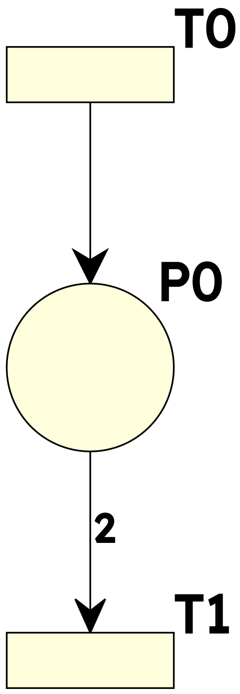
- bisogna generare 2 gettoni in maniera *atomica*

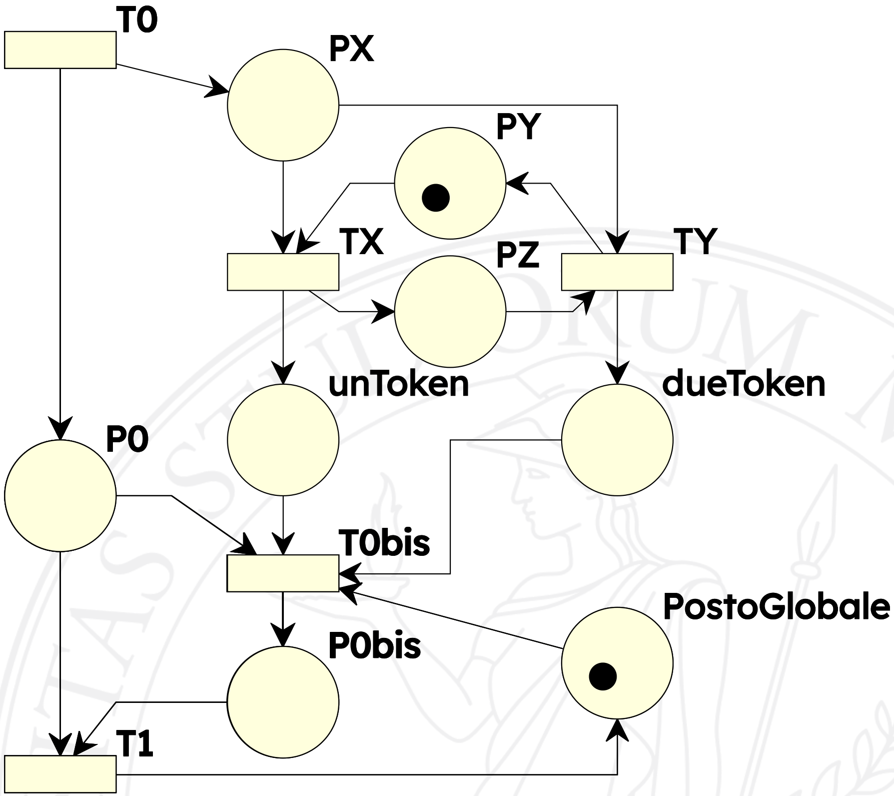




Eliminazione pesi archi (cont.)

- bisogna consumare 2 gettoni in maniera *atomica*





Reti Condizioni - Eventi

- All’altro estremo delle reti di Petri ci sono le **reti condizioni-eventi**:
  - tutti gli archi hanno peso 1
  - tutti i posti hanno capacità 1

I posti corrispondono alle condizioni (vere o false a seconda della presenza del gettone)

Le transizioni corrispondono agli eventi (che scattano quando un insieme di condizioni sono vere)

Una rete P/T limitata ha comunque una corrispondente nella classe C/E

Rete conservativa rispetto a una funzione H

- Una rete P/T con marcatura M_0 si dice conservativa rispetto a una funzione $H : P \to \mathbb{N}^+$ se e solo se:
  - $\forall M \in R(P/T, M_0) \quad \sum_{p\in P} H(p)M(p) = \sum_{p\in P} H(p)M_0(p)$
- Diremo che una rete P/T è conservativa in una marcatura M se esiste almeno una funzione H per cui è conservativa

Rete strettamente conservativa

- Una rete P/T con marcatura M conservativa rispetto a una funzione che assegna tutti pesi uguali (ad es. 1), si dice **strettamente conservativa**

- $\forall M \in R(P/T, M_0) \quad \sum_{p \in P} M(p) = \sum_{p \in P} M_0(p)$
  - il numero di gettoni è conservato, resta uguale in tutte le marcature

- $\forall t \in T \quad \sum_{p \in pre(t)} W(\langle p, t \rangle) = \sum_{p \in post(t)} W(\langle t, p \rangle)$
  non morta
  - il numero di gettoni distrutti dallo scatto di una transizione è uguale al numero di gettoni che crea

È una proprietà statica o dinamica?

Stato base e rete reversibile

- Una marcatura M' è detta **stato base** (*home state*) se
  - $\forall M \in R(P/T, M_0) \quad M' \in R(P/T, M)$
- Una rete di Petri è detta **reversibile** se la marcatura iniziale M_0 è uno stato base
- Altre possibili domande:
  - può essere raggiunta una determinata marcatura?
  - è possibile una certa sequenza di scatti?
  - esiste uno stato di deadlock?
  - la rete (o una certa transizione) è viva?

Tecniche di analisi

- Dinamiche
  - albero (grafo) delle marcature raggiungibili
  - albero (grafo) della copertura delle marcature raggiungibili
- Statiche (Strutturali)
  - identificazione P-invarianti
  - identificazione T-invarianti

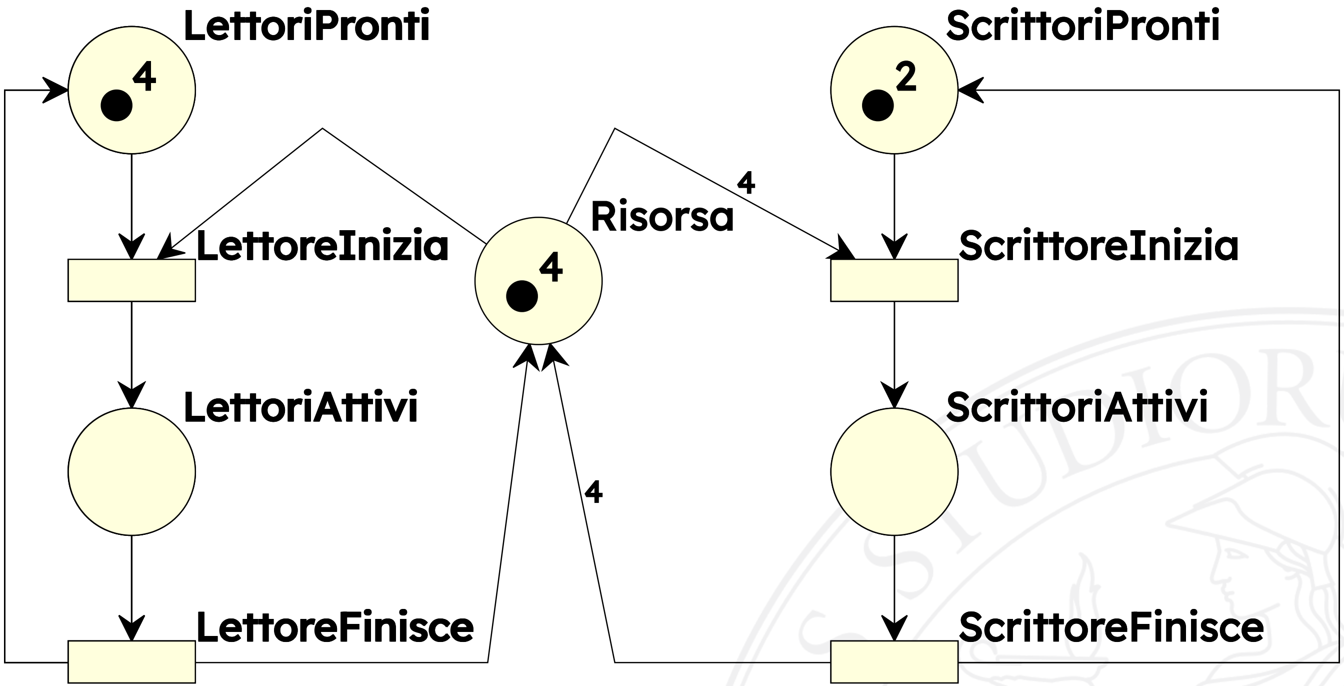
Albero delle marcature raggiungibili

1. Crea la radice corrispondente alla marcatura iniziale, etichetta il nodo come nuovo
2. finché esistono nodi etichettati nuovo esegui i seguenti passi:
   - seleziona una marcatura M con etichetta nuovo e togli etichetta
   - se M è identica ad una marcatura sul cammino dalla radice ad M, etichetta M come duplicata e passa ad un'altra marcatura
   - se nessuna transizione è abilitata in M, etichetta la marcatura come finale
   - finché esistono transizioni abilitate in M esegui i seguenti passi per ogni transizione t abilitata in M:
     - crea la marcatura M' prodotta dallo scatto di t
     - crea un nodo corrispondente a M', aggiungi un arco da M a M' ed etichetta M' come nuovo

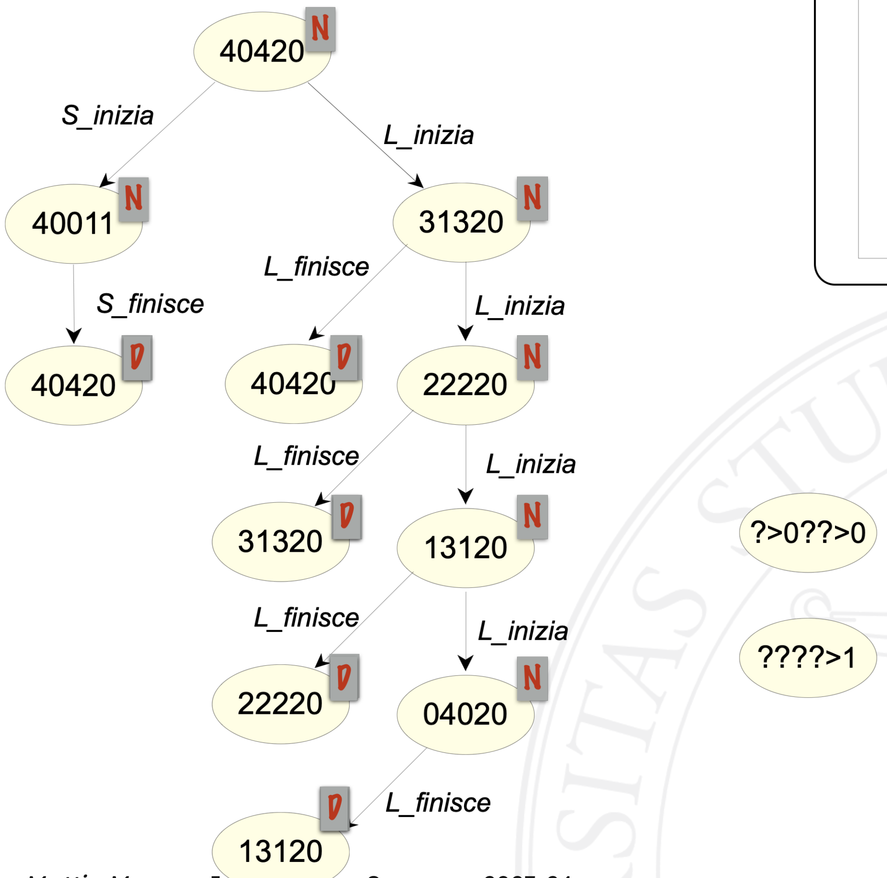
Esercizio

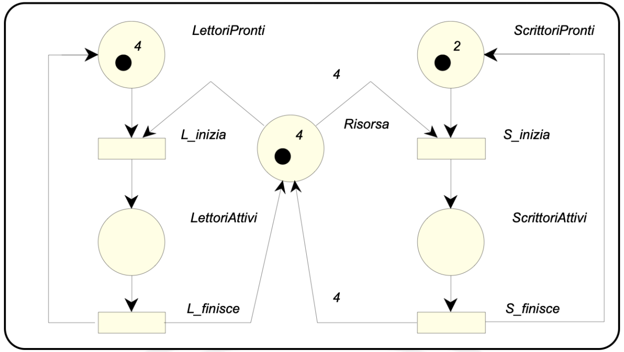
- Modellare con una rete di Petri l'accesso ad una risorsa condivisa da parte di 4 lettori e 2 scrittori
  - i lettori possono accedere simultaneamente alla risorsa
  - gli scrittori hanno bisogno di accesso esclusivo per potere modificare la risorsa

Possibile soluzione

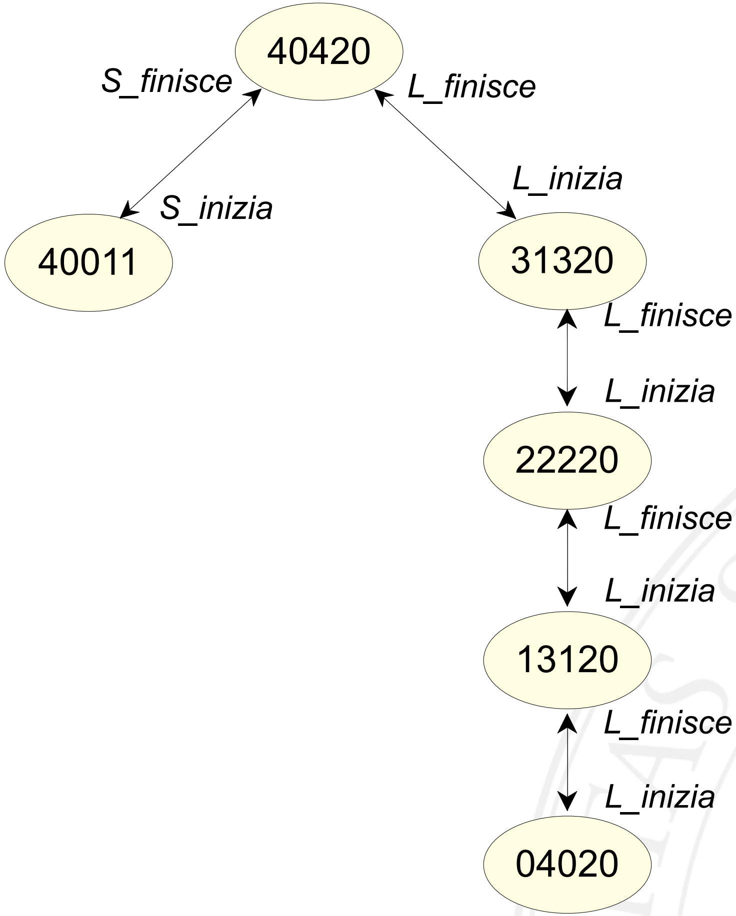


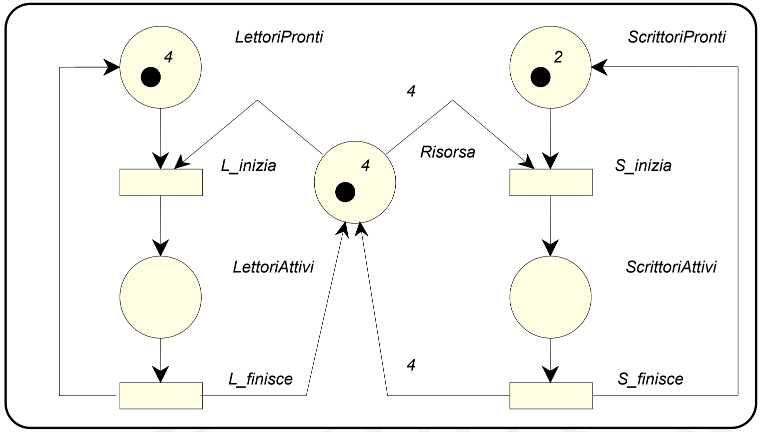
Albero delle marcature raggiungibili

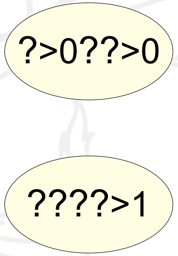




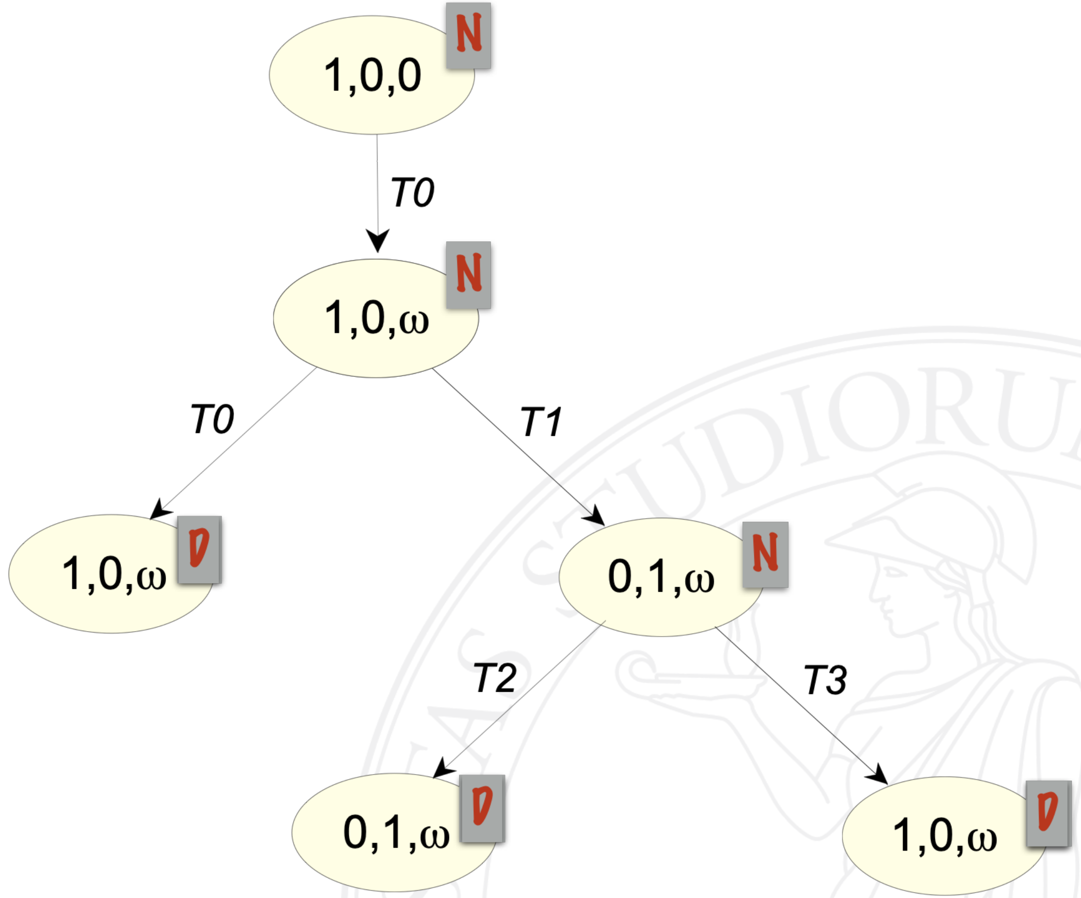
Grafo raggiungibilità



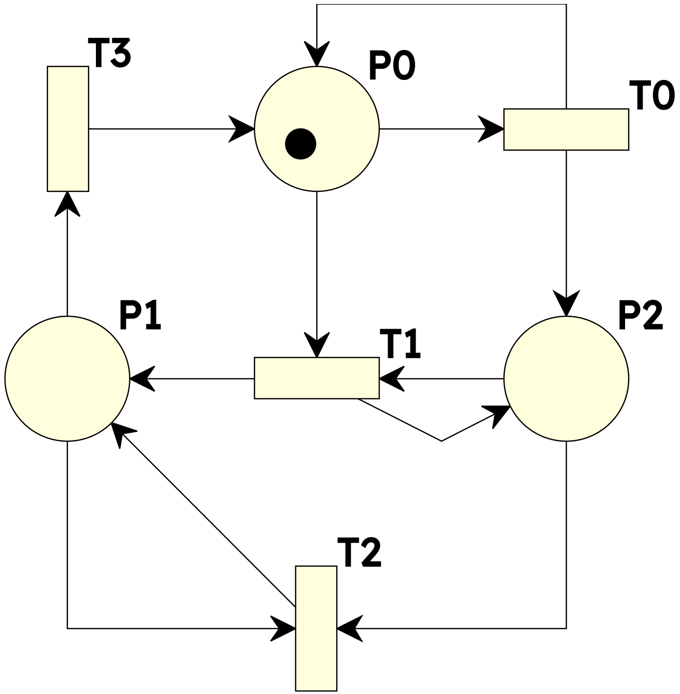


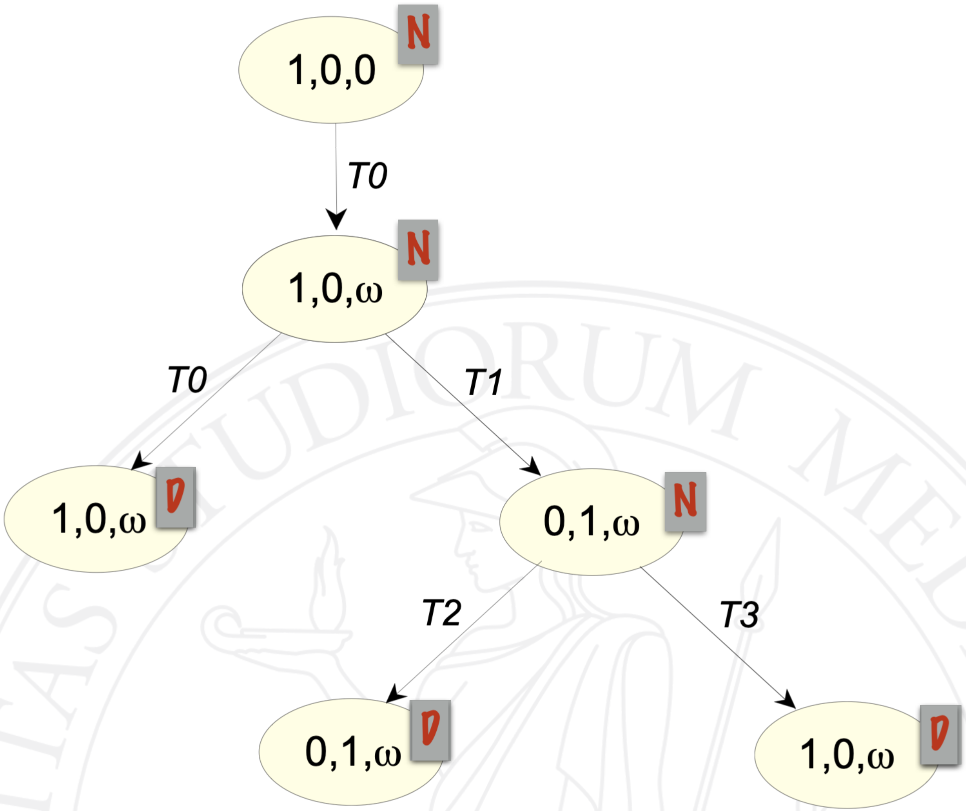


Grafo delle marcature raggiungibili



Esempio di albero di copertura





Analisi di raggiungibilità

- Problemi
  - deve enumerare tutte le possibili marcature raggiungibili
    - se la rete non è limitata sono infiniti e quindi non può essere completato
    - non ci sa dire se una rete è limitata
- però se la rete è limitata:
  - ci sa dire quasi tutto…
    - è la esplicitazione degli stati della rete (l'automa a stati finiti corrispondente)

Copribilità (*coverability*)

- Una marcatura M **copre** una marcatura M' (M' è coperta da M) se e solo se: $\forall p \in P \quad M(p) \geq M'(p)$
  - i posti per cui $M(p) > M'(p)$ di dice che son coperti in maniera *propria*
- Una marcatura M è detta **copribile** a partire da una marcatura M' se esiste una marcatura $M'' \in R(P/T, M')$ che copre M

Se M è la marcatura *minima* per abilitare t,

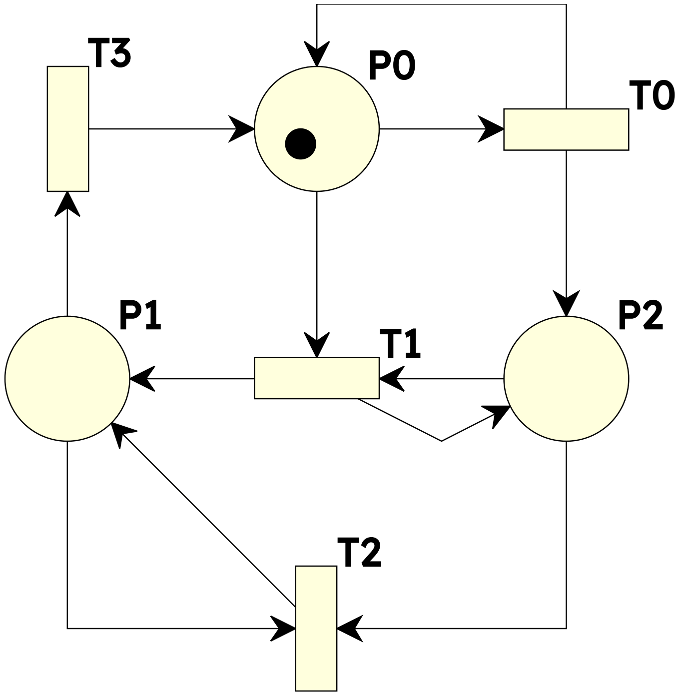
$$\forall p \in pre(t) \quad M(p) = W(\langle p, t \rangle) \quad \text{e} \quad \forall p \in P \setminus pre(t) \quad M(p) = 0$$

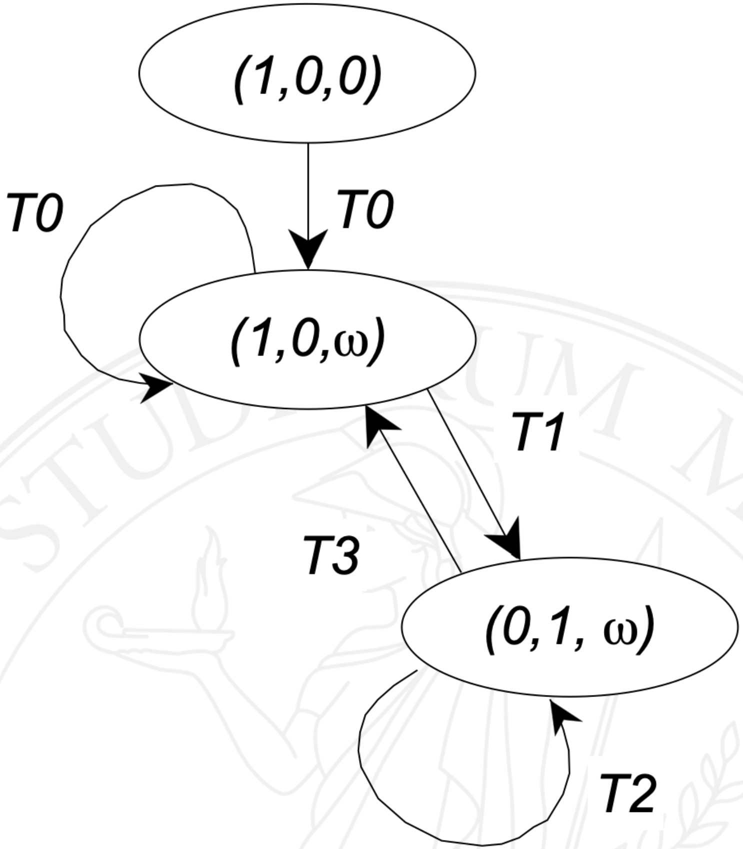
allora la transizione t è morta se e solo se M non è copribile a partire dalla marcatura corrente

Albero di copertura

1. Crea la radice corrispondente alla marcatura iniziale, etichetta il nodo come nuovo
2. finché esistono nodi etichettati nuovo esegui i seguenti passi:
   - seleziona una marcatura M con etichetta nuovo e togli etichetta
   - se M è identica ad una marcatura sul cammino dalla radice ad M, etichetta M come duplicata e passa ad un'altra marcatura
   - se nessuna transizione è abilitata in M, etichetta la marcatura come finale
   - finché esistono transizioni abilitate in M esegui i seguenti passi per ogni transizione t abilitata in M:
     - crea la marcatura M' prodotta dallo scatto di t
     - se sul cammino dalla radice a M esiste una marcatura M'' coperta da M', modifica M' scrivendo ω in tutte le posizioni corrispondenti a *coperture proprie*
     - crea un nodo corrispondente a M', aggiungi un arco da M a M' ed etichetta M' come nuovo

Esempio di grafo di copertura

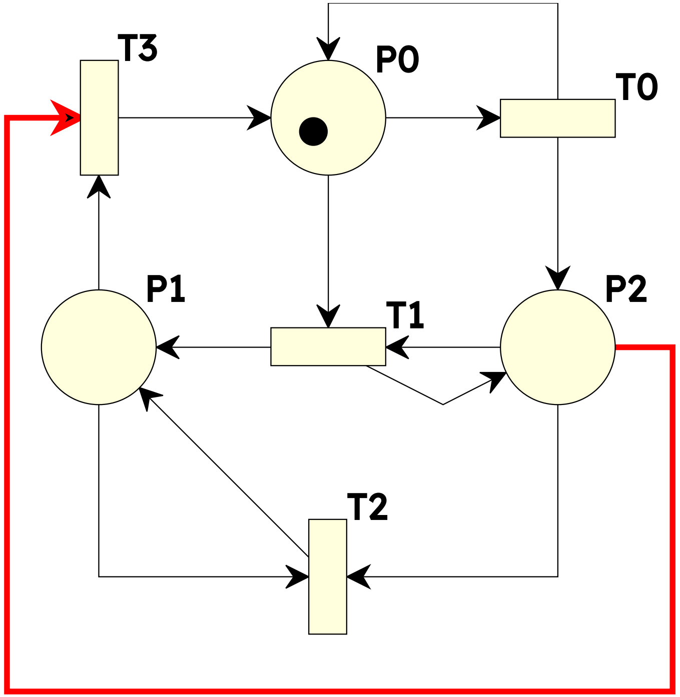




Analisi di copribilità

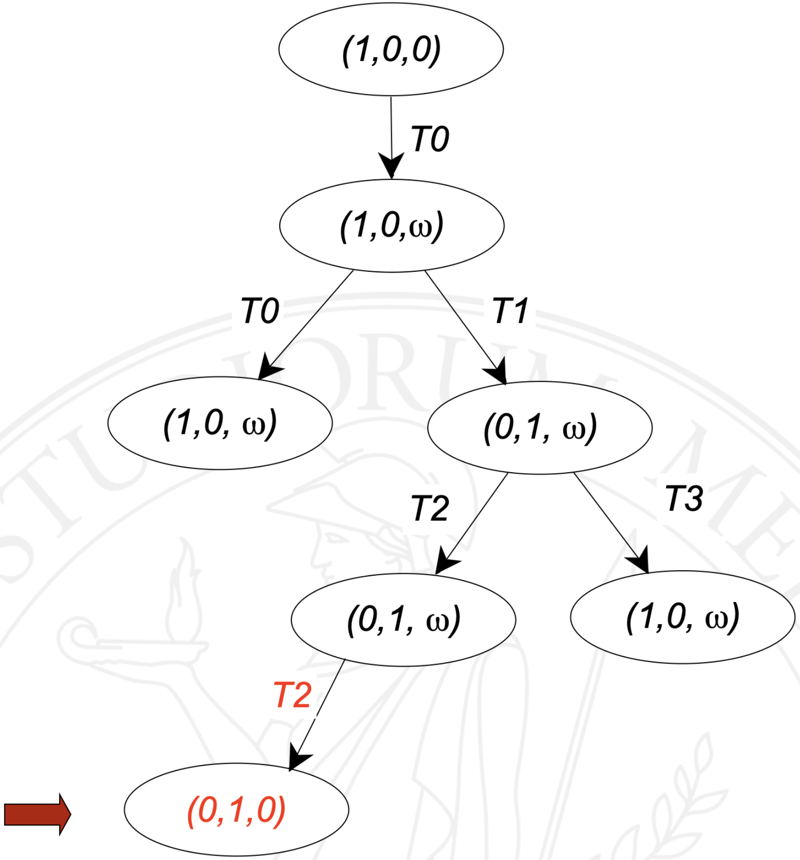
1. una rete di Petri è limitata se ω non compare in nessun nodo dell'albero di copertura
2. Una rete di Petri è binaria se nell'albero di copertura compaiono solo 0 e 1
3. una transizione è morta (0-live) se non appare come etichetta di un arco dell'albero di copertura
4. condizione necessaria affinché una marcatura M sia raggiungibile è l'esistenza di un nodo etichettato con una marcatura che copre M (non è sufficiente)
5. non è possibile decidere se una rete è viva

Albero di copertura limiti



Rete non viva

Non è possibile metterlo, e
non viene espresso dall'albero



Rappresentazione matriciale

- È possibile rappresentare (definire) una rete di Petri mediante delle matrici
  - ennesima vista …
  - trasformazione automatica…
  - facilmente trattabile matematicamente
- Uso diverse matrici:

I	archi in input alle transizioni
O	archi in output alle transizioni
M	marcatura dei posti

Matrice di Input e Output

- $p : 1..|P| \to P$ assegna un indice ad ogni posto
- $t : 1..|T| \to T$ assegna un indice ad ogni transizione

Le due matrici I e O sono $|P| \times |T|$

$$\forall \langle p(i), t(j) \rangle \in F \quad I[i][j] = W(\langle p(i), t(j) \rangle)$$

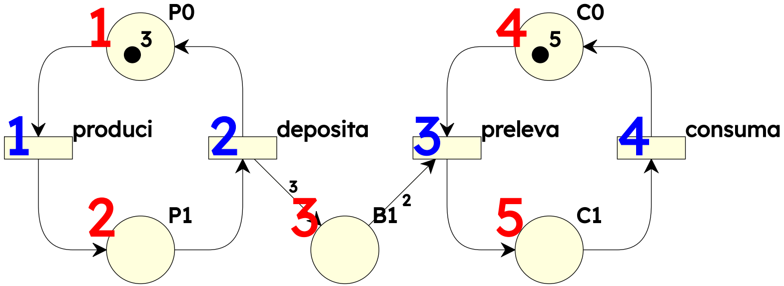
$$\forall \langle p(i), t(j) \rangle \notin F \quad I[i][j] = 0$$

$$\forall \langle t(j), p(i) \rangle \in F \quad O[i][j] = W(< t(j), p(i) >)$$

$$\forall \langle t(j), p(i) \rangle \notin F \quad O[i][j] = 0$$

Indicheremo il vettore colonna k di una matrice X con la notazione $X[.][k]$

Esempio matrici



Un vettore colonna m di dimensione $|P|$ rappresenta la marcatura M:

$$\forall i \in 1..|P| \quad m[i] = M(p(i))$$

I	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	0	0	2	0
4	0	0	1	0
5	0	0	0	1

O	1	2	3	4
1	0	1	0	0
2	1	0	0	0
3	0	3	0	0
4	0	0	0	1
5	0	0	1	0

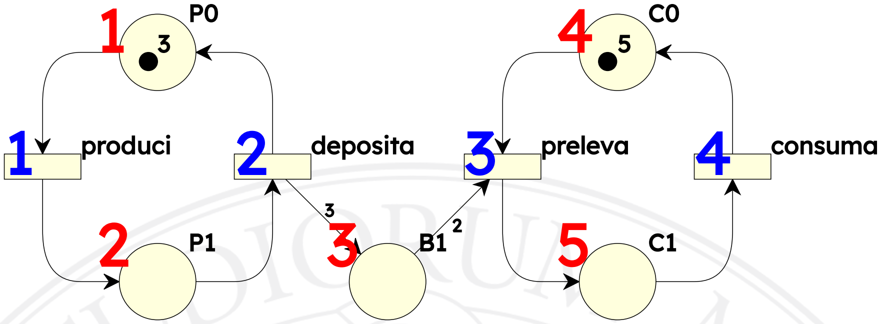
m	
1	3
2	0
3	0
4	5
5	0

Abilitazione di una transizione

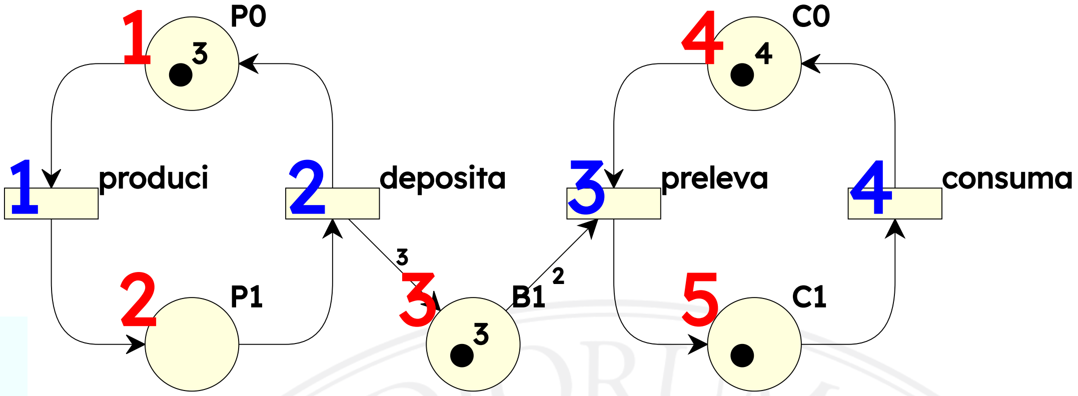
- La transizione t_j è abilitata in una marcatura espressa dal vettore m:
  - $m[t_j >$ se e solo se $\boxed{I[.][j] \leq m}$
    - (elemento per elemento)

$I[.][1]$		m
1	$\leq$	3
0	$\leq$	0
0	$\leq$	0
0	$\leq$	5
0	$\leq$	0

$I[.][2]$		m
0	$\leq$	3
1	$\leq$	0
0	$\leq$	0
0	$\leq$	5
0	$\leq$	0



Quali sono le transizioni abilitate?



I	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	0	0	2	0
4	0	0	1	0
5	0	0	0	1

m	
1	3
2	0
3	3
4	4
5	1

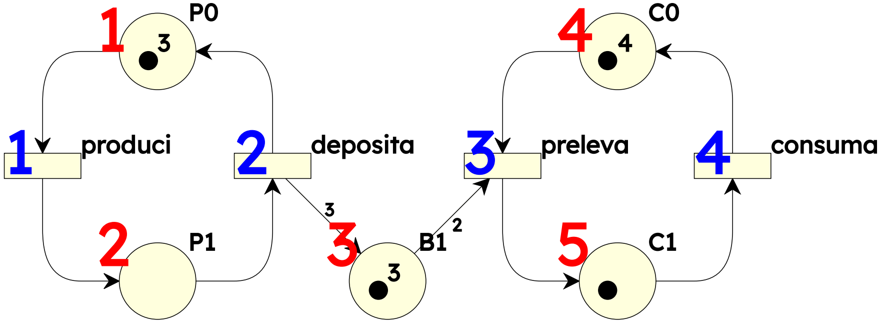
Tutte tranne t_2 (deposita)

Scatto di una transizione

- Lo scatto di una transizione t_j in una marcatura m produce la marcatura m':

$$m[t_j > m'$$

$$\boxed{m' = m - I[.][j] + O[.][j]}$$



m	
1	3
2	0
3	3
4	4
5	1

I	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	0	0	2	0
4	0	0	1	0
5	0	0	0	1

O	1	2	3	4
1	0	1	0	0
2	1	0	0	0
3	0	3	0	0
4	0	0	0	1
5	0	0	1	0

m'	
1	3
2	0
3	1
4	3
5	2

Matrice di incidenza C

$$\boxed{C = O - I}$$

- Risulta utile per ottimizzare il calcolo delo scatto di una transizione
  - ma è sufficiente per stabilirne la abilitazione **solo se** $pre(t) \cap post(t) = \emptyset$

Sequenza di scatti

$$M[t_1 > M'[t_2 > M''[t_3 > \ldots M_n$$
$$M[t_1 t_2 t_3 \ldots > M_n$$
$$M[S_n > M_n$$

$\boxed{m_n = m + Cs}$ s è il vettore delle *occorrenze* degli scatti delle transizioni nella sequenza.

ATTENZIONE: $m + Cs \geq 0$ è condizione necessaria ma non sufficiente affinché la sequenza sia **ammissibile**

- in tutte le marcature intermedie deve essere abilitata la transizione che deve scattare

Nuova tecnica di analisi

- Ricerca di **invarianti** all'interno della rete
  - P-invarianti
    - invarianti sui posti (si parla di caratteristiche invarianti sulle marcature)
  - T-invarianti
    - invarianti sulle transizioni (si parla di caratteristiche invarianti sulle sequenze di scatti)

P-invarianti

- Un vettore h di dimensione $|P|$ è un P-invariante se e solo se:

$$\boxed{\forall m' \in R(P/T, m) \quad hm = hm'}$$

Ricorda la funzione H della definizione di rete conservativa però i pesi non devono essere per forza maggiori di 0 (basta che non siano tutti uguali a 0)

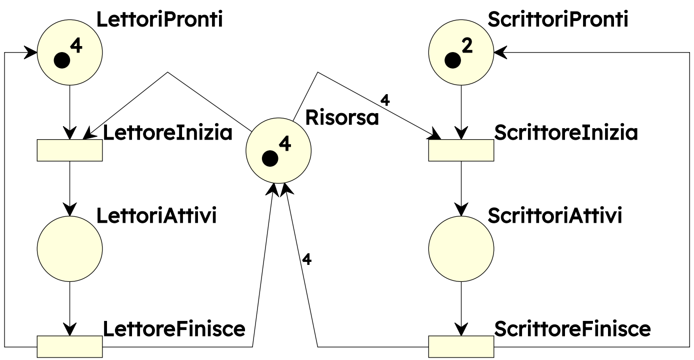
- $hm = hm'$ — ma se m' è raggiungibile da m allora esiste una sequenza s tale per cui $m + Cs = m'$ e sostituendo m'
- $hm = h(m + Cs)$
- $hm = hm + hCs$
- $hCs = 0$ — $\forall s$ che rappresenti una sequenza ammissibile (quindi non tutte ma quasi)
- $\boxed{hC = 0}$ — un unico sistema lineare con $|P|$ equazioni e $|T|$ incognite

Siamo passati da una proprietà dinamica a una tecnica di analisi statica che trova un sottoinsieme dei P-invarianti

Copertura di P-invarianti

- una combinazione lineare di P-invarianti è un P-invariante
  - h_1, h_2 P-invarianti $\quad \Rightarrow \quad \forall \alpha, \beta \in \mathbb{R} \quad \alpha h_1 + \beta h_2$ è un P-invariante
- un P-invariante che ha tutti pesi ≥ 0 è detto *P-invariante semi-positivo*
- Se un posto ha peso positivo in un P-Invarante semipositivo, allora quel posto è **limitato**
- Un rete P/T si dice copribile/ricoperta da P-Invarianti se per ogni posto esiste almeno un P-Invariante semipositivo che ha peso positivo in quel posto
  - esisterà una combinazione lineare di P-invarianti che copre ogni posto
  - quindi rete conservativa rispetto a questi pesi, quindi limitata

Esempio di P-invarianti



$$\begin{bmatrix} h_0 & h_1 & h_2 & h_3 & h_4 \end{bmatrix} \times \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ -1 & 1 & -4 & 4 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$$

$$hC = 0$$

$$\begin{cases} -h_0 + h_1 - h_2 = 0 \\ h_0 - h_1 + h_2 = 0 \\ -4h_2 - h_3 + h_4 = 0 \\ 4h_2 + h_3 - h_4 = 0 \end{cases}$$

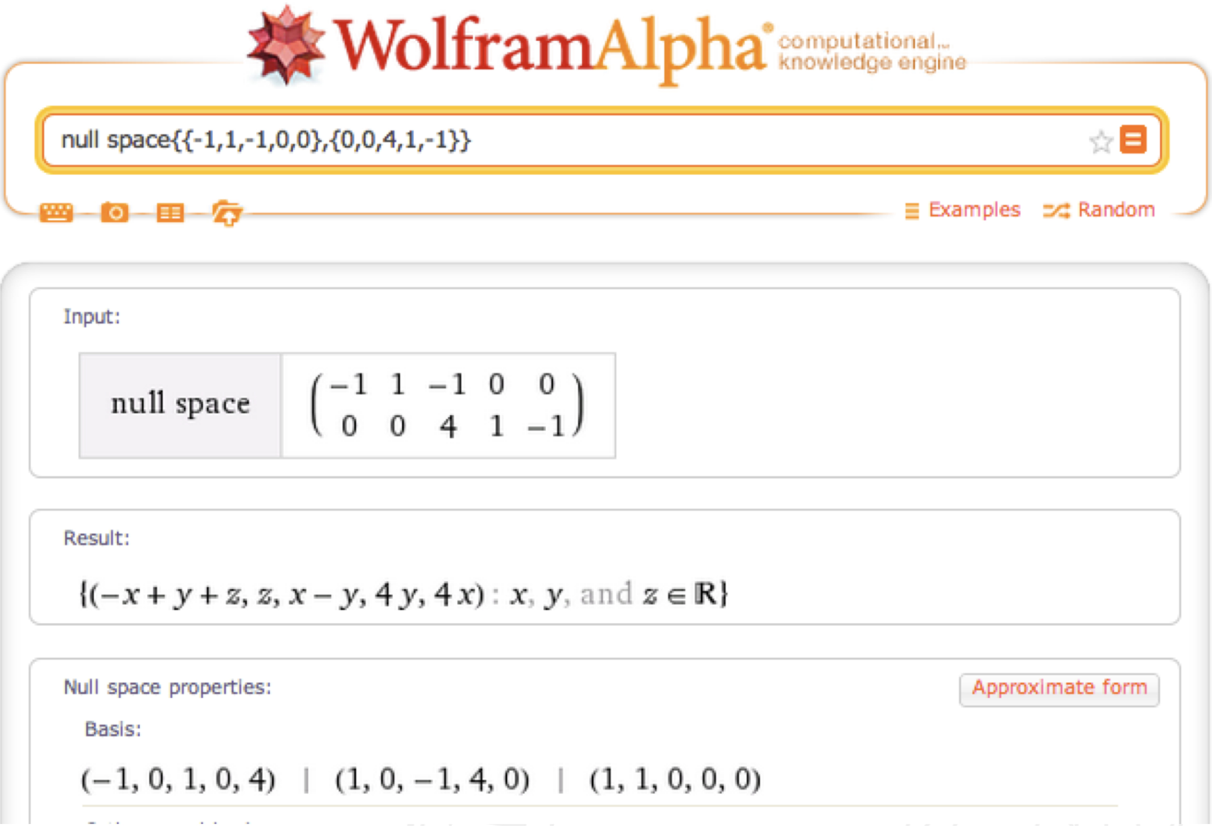
$$\begin{cases} h_0 - h_1 + h_2 = 0 \\ 4h_2 + h_3 - h_4 = 0 \end{cases}$$

I	Lettore Inizia	Lettore Finisce	Scrittore Inizia	Scrittore Finisce
Lettori Pronti	1	0	0	0
Lettori Attivi	0	1	0	0
Risorsa	1	0	4	0
Scrittori Pronti	0	0	1	0
Scrittori Attivi	0	0	0	1

O	Lettore Inizia	Lettore Finisce	Scrittore Inizia	Scrittore Finisce
Lettori Pronti	0	1	0	0
Lettori Attivi	1	0	0	0
Risorsa	0	1	0	4
Scrittori Pronti	0	0	0	1
Scrittori Attivi	0	0	1	0

C	Lettore Inizia	Lettore Finisce	Scrittore Inizia	Scrittore Finisce
Lettori Pronti	-1	1	0	0
Lettori Attivi	1	-1	0	0
Risorsa	-1	1	-4	4
Scrittori Pronti	0	0	-1	1
Scrittori Attivi	0	0	1	-1

Risolviamolo



UNIVERSITÀ DEGLI STUDI DI MILANO

Algoritmo di Farkas (1902)

- Trova basi minime semipositive

$D_0 := (C \mid E_n)$;
for $i := 1$ **to** m **do**
  for d_1, d_2 rows in D_{i-1} such that $d_1(i)$ and $d_2(i)$ have opposite signs **do**
    $d := |d_2(i)| \cdot d_1 + |d_1(i)| \cdot d_2$; (* $d(i) = 0$ *)
    $d' := d/\gcd(d(1), d(2), \ldots, d(m+n))$;
    augment D_{i-1} with d' as last row;
  endfor;
  delete all rows of the (augmented) matrix D_{i-1} whose i-th component is different from 0, the result is D_i;
endfor;
delete the first m columns of D_m

```
D = [C,eye(rows(C))]
for i = 1:columns(C)
    D1 = [];
    for j = 1:rows(D)
        for k = j+1:rows(D)
          if D(j,i)*D(k,i)<0
             d = abs(D(j,i))*D(k,:)+abs(D(k,i))*D(j,:);
             d = d/gcd(num2cell(A){:});
             A1 = [A1;d];
       endif
    endfor
    endfor
    for j = 1:rows(D)
      if (D(j,i) == 0)
        D1 = [D1;D(j,:)];
      endif
    endfor
    D = D1;
endfor
soluzioni = D(:,columns(C)+1:columns(D))
```

Simuliamolo

$$\begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ -1 & 1 & -4 & 4 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Concateno e poi per ogni coppia di righe tali che hanno nella prima colonna un numero diverso da 0 e di segno opposto, faccio la combinazione lineare (prima riga 1 e 2 poi riga 2 e 3)

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -4 & 4 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Guardo la terza colonna e identifico le coppie righe 2 e 4 e poi 3 e 4

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

"butto via" le prime 4 colonne che per costruzione saranno diventate tutti zeri e il rimanente sono le 3 basi dei miei invarianti

Interpretiamo i risultati

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 4 \end{bmatrix}$$

m	
Lettori Pronti	4
Lettori Attivi	0
Risorsa	4
Scrittori Pronti	2
Scrittori Attivi	0

- Si riparte da $\boxed{\forall m' \in R(P/T, m) \quad hm' = hm}$
- Ora però h non è più *ignoto*, e chiaramente conosciamo anche m (marcatura iniziale)

1. $LettoriPronti + LettoriAttivi = 4$
   - Il numero di lettori nel sistema è costante (e uguale a numero di gettoni inizialmente in LettoriPronti)
2. $ScrittoriPronti + ScrittoriAttivi = 2$
   - Il numero di scrittori nel sistema è costante (e uguale a numero di gettoni inizialmente in ScrittoriPronti)
3. $LettoriAttivi + Risorsa + 4ScrittoriAttivi = 4$
   - Questo 4 dipende invece dal numero di gettoni inizialmente nel posto Risorsa

- $LettoriAttivi > 0 \implies ScrittoriAttivi = 0$
- $ScrittoriAttivi > 0 \implies LettoriAttivi = 0$
- $ScrittoriAttivi \leq 1$
- $LettoriAttivi \leq 4$

T-invarianti

Fanno riferimento a sequenze di scatti cicliche (cioe` che possono essere ripetute) che riportano nella condizione iniziale

$$m' = m + Cs$$

$$m' = m$$

soluzioni del sistema: $\boxed{Cs = 0}$

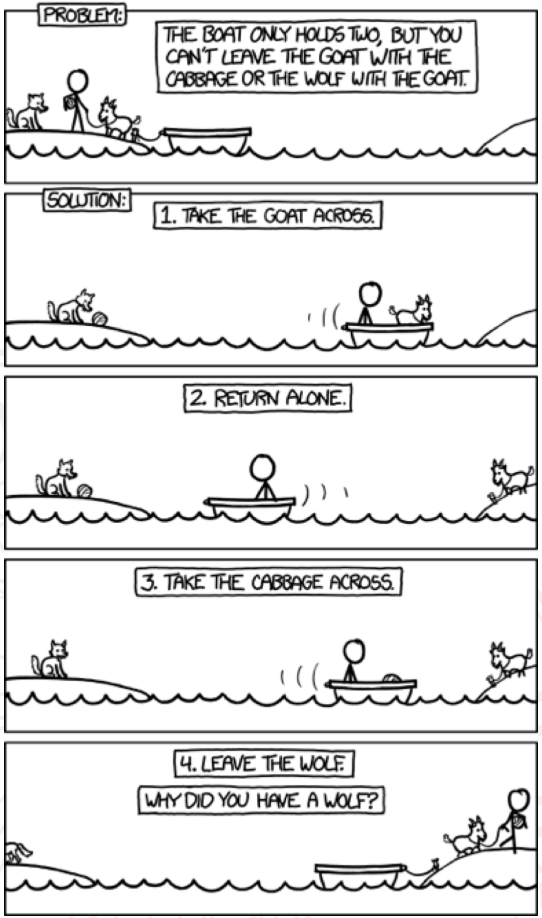
non è detto siano tutte valide!

Esercizio

Modellare il problema del barcaiolo che deve traghettare da una sponda all'altra di un torrente un lupo, una capra e un cavolo con una barca di capacità 1 (può trasportare solo un elemento alla volta oltre a sè stesso).

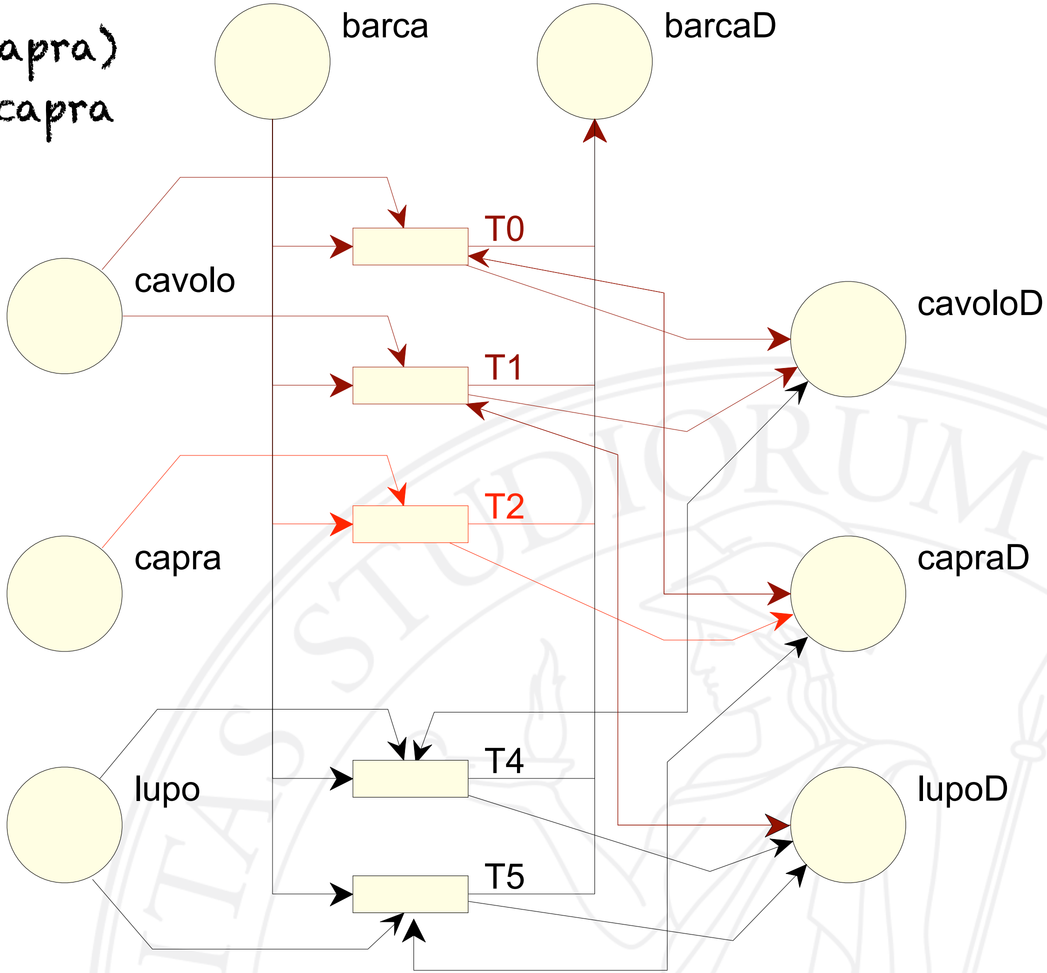
Il trasporto è vincolato dalla necessità di non lasciare soli:

- lupo e capra
- capra e cavolo



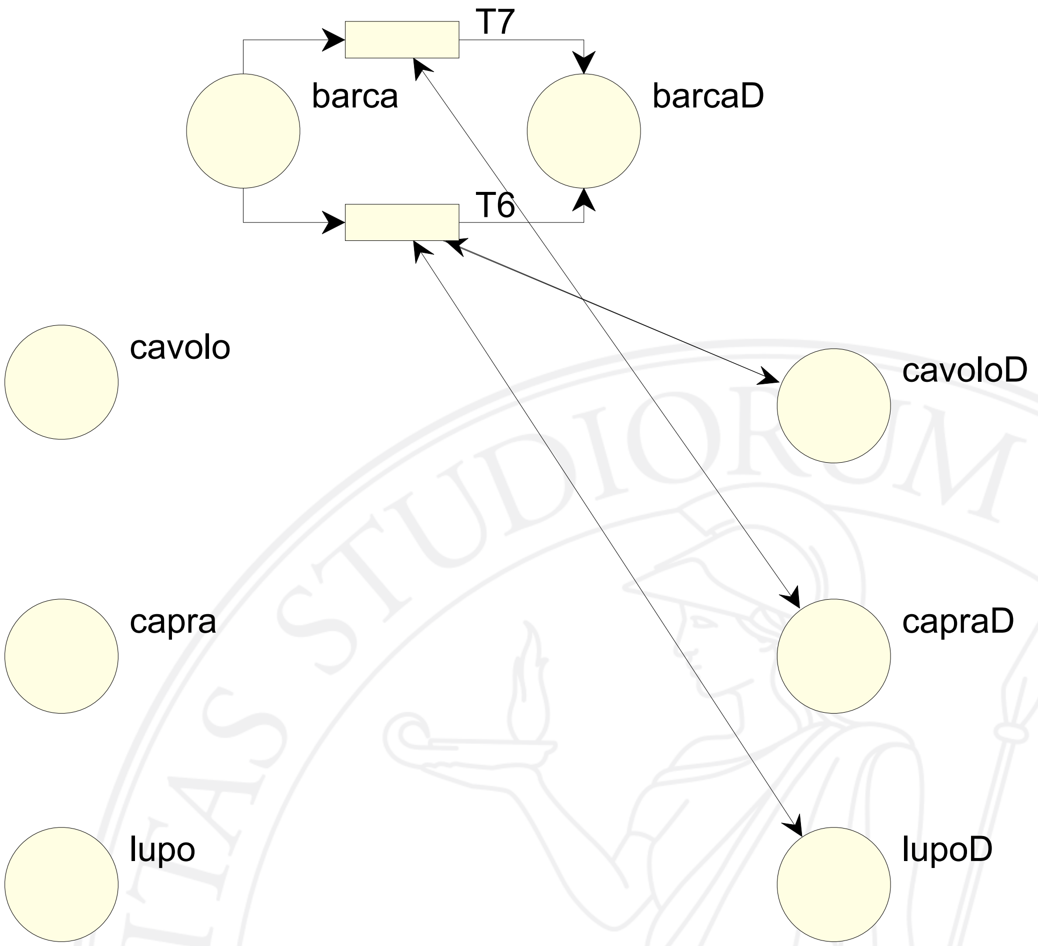
Soluzione

Lupo: NOT(cavolo AND capra)
== NOT cavolo OR NOT capra

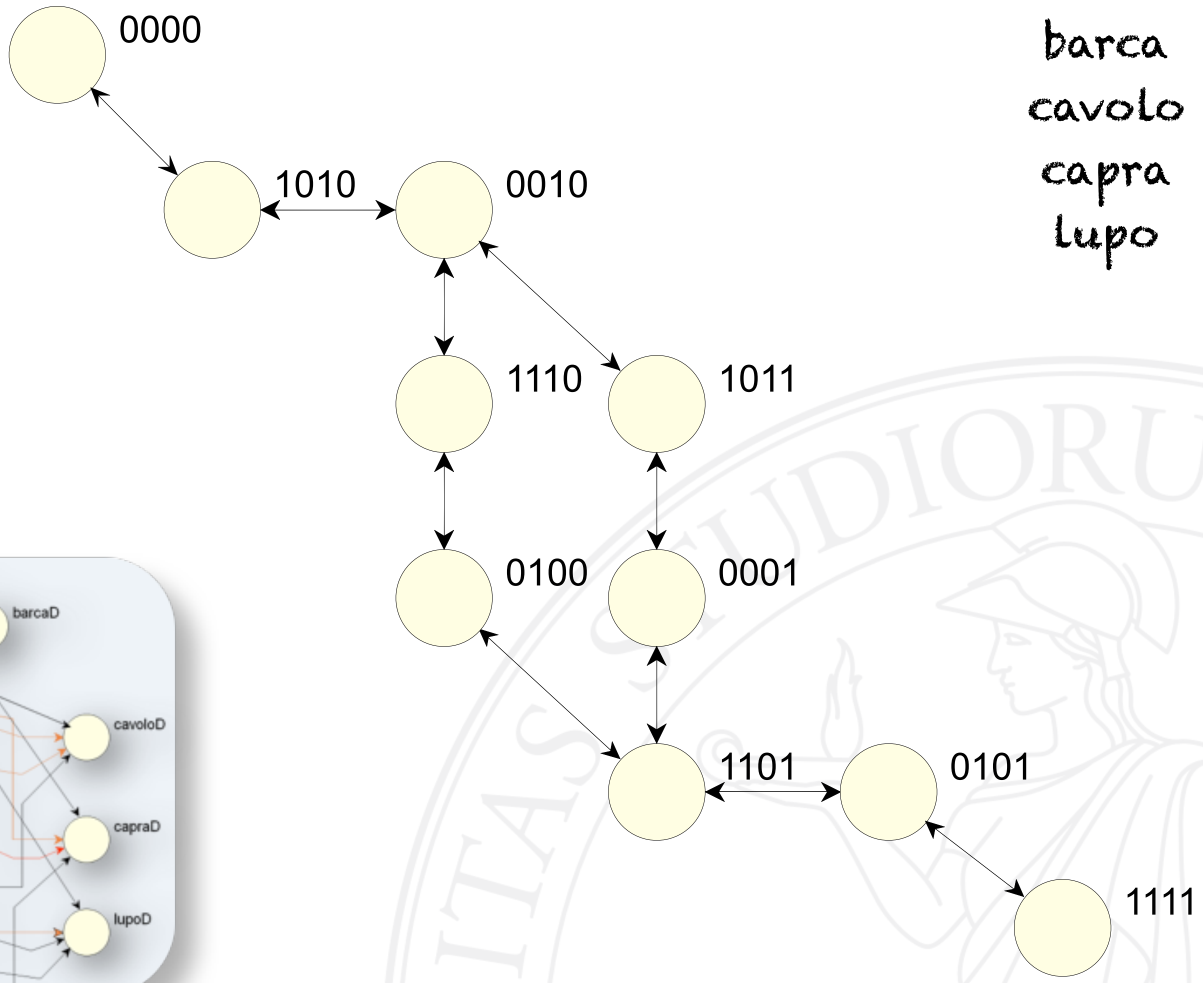


Soluzione (cont.)

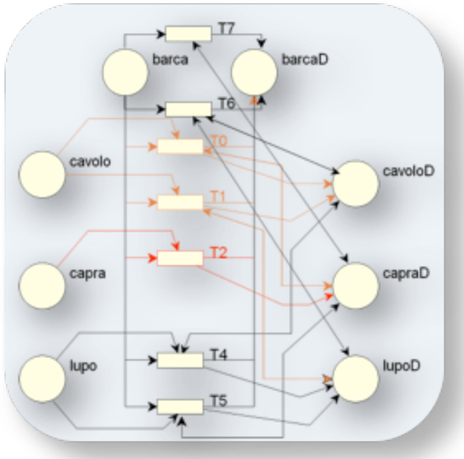
Barcaiolo da solo



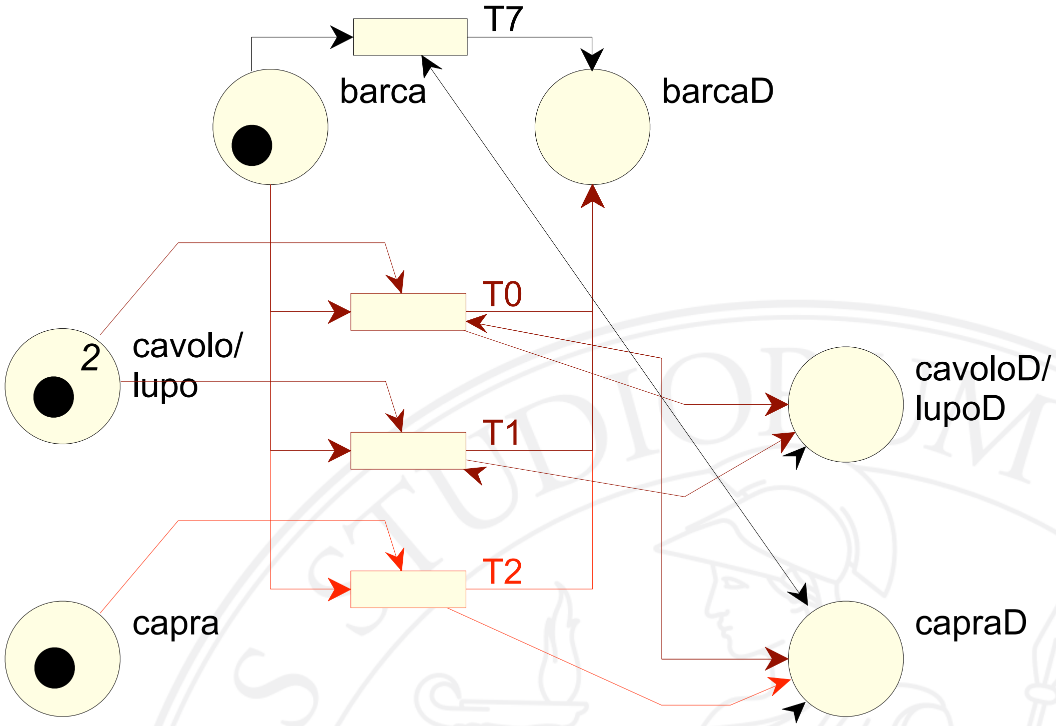
Grafo di raggiungibilità



barca
cavolo
capra
lupo



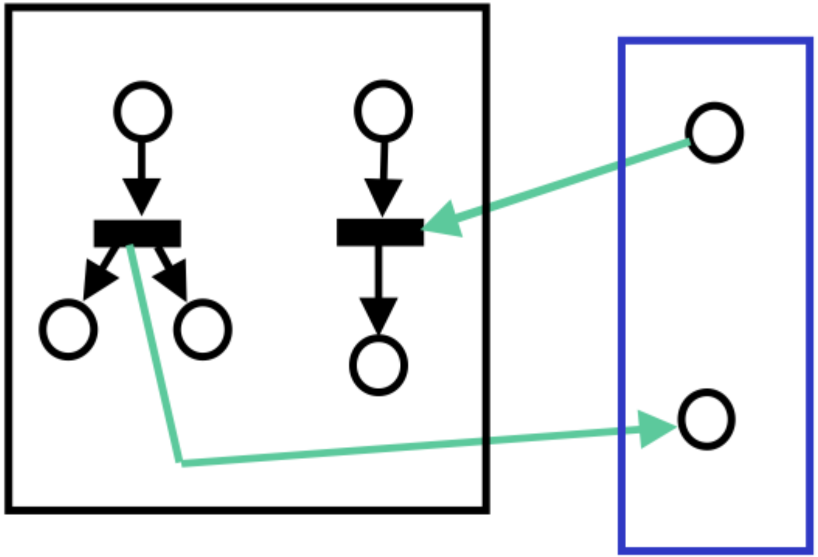
Soluzione alternativa



Il lupo... è un cavolo!

Controllori con specifica a stati proibiti

- Transizioni osservate
- Transizioni controllate



Attenzione a cosa si può controllare

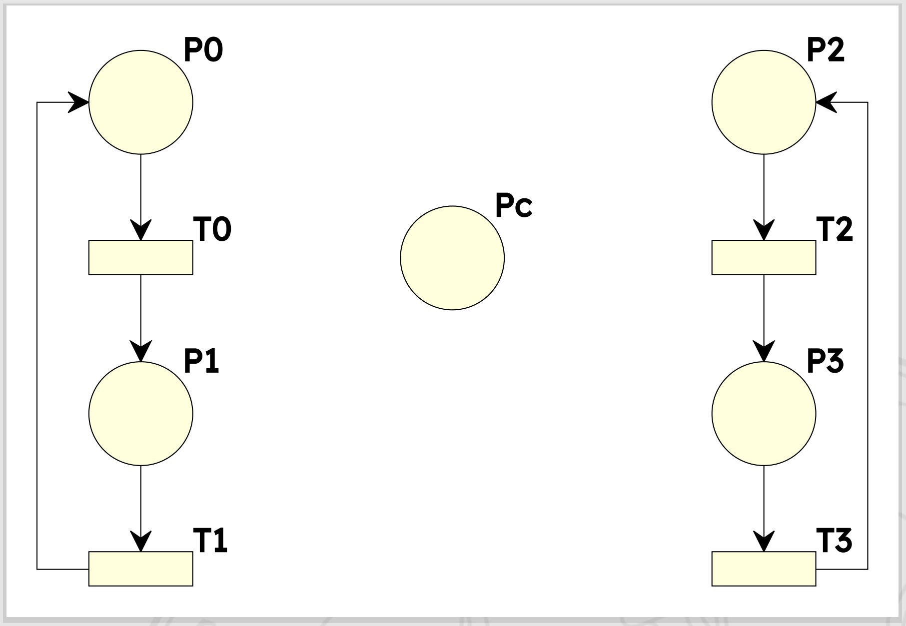
- Non tutte le transizioni sono osservabili
  - es. eventi che non sono rilevabili dal controllore, o troppo "costosi" da rilevare
- Non tutti gli eventi sono condizionabili
  - es. una transizione che modella un guasto (questo non può essere impedito dal controllore)

Quali vincoli esprimere?

- Esprimiamo il comportamento desiderato (le proprietà) del nostro sistema dicendo che una combinazione lineare delle marcature non deve superare un certo valore…
  - fissiamo perciò (quasi) dei P-invarianti "desiderati"

$$\boxed{L \cdot M \leq b}$$

Mutua esclusione



- $P1 + P3 \leq 1$
- aggiungiamo P_c
- $P1 + P3 + P_c = 1$

dobbiamo aggiungere riga opportuna a C

dobbiamo aggiungere riga opportuna a m

Sintesi del controllore

$$C = \begin{bmatrix} C_s \\ C_c \end{bmatrix} \qquad m = \begin{bmatrix} m_s \\ m_c \end{bmatrix}$$

$$Lm_s + m_c = b$$

$$\begin{bmatrix} L \; I \end{bmatrix} m = b$$

- Ma allora è quasi come dire che $\begin{bmatrix} L \; I \end{bmatrix}$ è un P-invariante
  - quindi deve valere:

$$\begin{bmatrix} L \; I \end{bmatrix} C = 0$$

$$LC_s + IC_c = 0$$

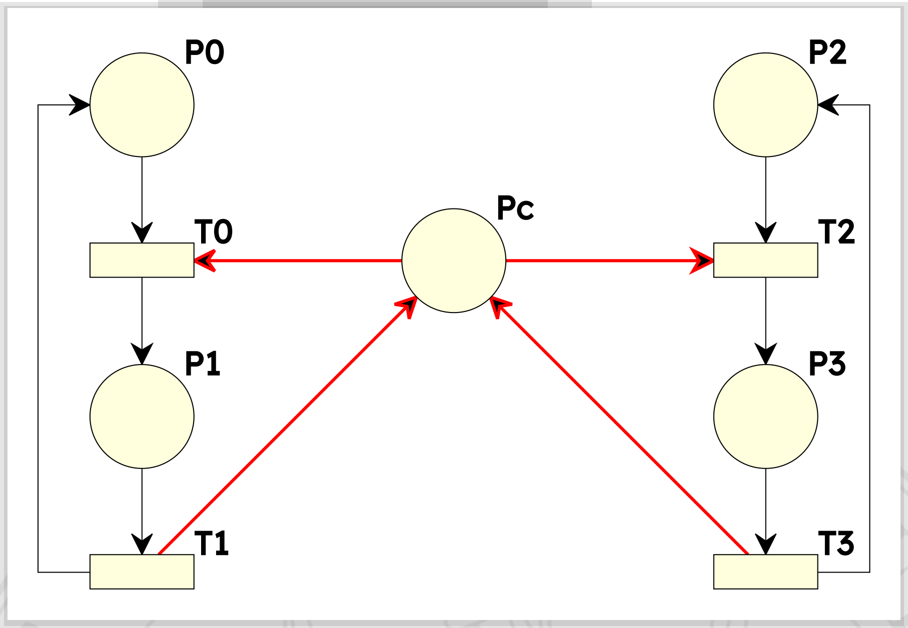
$$\boxed{C_c = -LC_s}$$

Mutua esclusione (sintesi)

corretto!

$$C_s = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

$$L = \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix}$$



$$\boxed{C_c = -L \cdot C_s = \begin{bmatrix} -1 & 1 & -1 & 1 \end{bmatrix}}$$

Sintesi della marcatura del controllore

$$C = \begin{bmatrix} C_s \\ C_c \end{bmatrix} \qquad m_0 = \begin{bmatrix} m_{0s} \\ m_{0c} \end{bmatrix}$$

$$Lm_{0s} + m_{0c} = b$$

$$\boxed{m_{0c} = b - Lm_{0s}}$$

$$m_{0c} = 1 - \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \end{bmatrix} = 1$$