



## **Automazione industriale dispense del corso**

### **6. Reti di Petri: definizioni, evoluzione e proprietà**

Luigi Piroddi  
piroddi@elet.polimi.it

## Reti di Petri

Un formalismo molto potente e compatto per rappresentare i sistemi a eventi discreti è costituito dalle reti di Petri.

Esse sono un'estensione significativa degli automi a stati finiti:

- ▶ possono rappresentare sistemi ad infiniti stati con un numero finito di nodi di un grafo
- ▶ in generale, sono strutture estremamente più compatte degli automi

Si prestano in modo particolare per rappresentare comportamenti complessi come:

- ▶ la concorrenza e la sincronizzazione tra operazioni
- ▶ il succedersi asincrono di eventi
- ▶ conflitti
- ▶ condivisione di risorse

## Caratteristiche principali:

- ▶ rappresentazione grafica semplice e di agevole interpretazione
- ▶ rappresentazione matematica (matriciale) facile
- ▶ disponibilità di diverse tecniche di analisi formale con le quali è possibile verificare le proprietà fondamentali di una rete e analizzare il comportamento del sistema modellizzato
- ▶ compattezza e modularità (non c'è esplosione dimensionale)
- ▶ *modello locale e distribuito*
  - ▼ lo stato di un modello e la transizione da uno stato all'altro sono concetti distribuiti
  - ▼ ogni porzione di rete modella una specifica operazione o un dispositivo

Sia gli automi che le reti di Petri rappresentano il meccanismo di evoluzione dello stato di un sistema ad eventi discreti:

- ▶ negli automi esso è rappresentato *esplicitamente*, enumerando tutti i possibili stati e tutte le transizioni possibili tra coppie di stati
- ▶ nelle reti di Petri, invece, esso è *implicito* nella topologia e marcatura della rete e la struttura del modello è più ricca di informazione:
  - ▼ l'informazione sullo stato del sistema è *distribuita* su parti della rete che descrivono le condizioni fondamentali che governano il funzionamento del sistema
  - ▼ lo stato complessivo della rete è interpretabile come composto da più stati parziali ed indipendenti relativi a sotto-reti
  - ▼ analogamente, una transizione influenza solo una parte dello stato complessivo

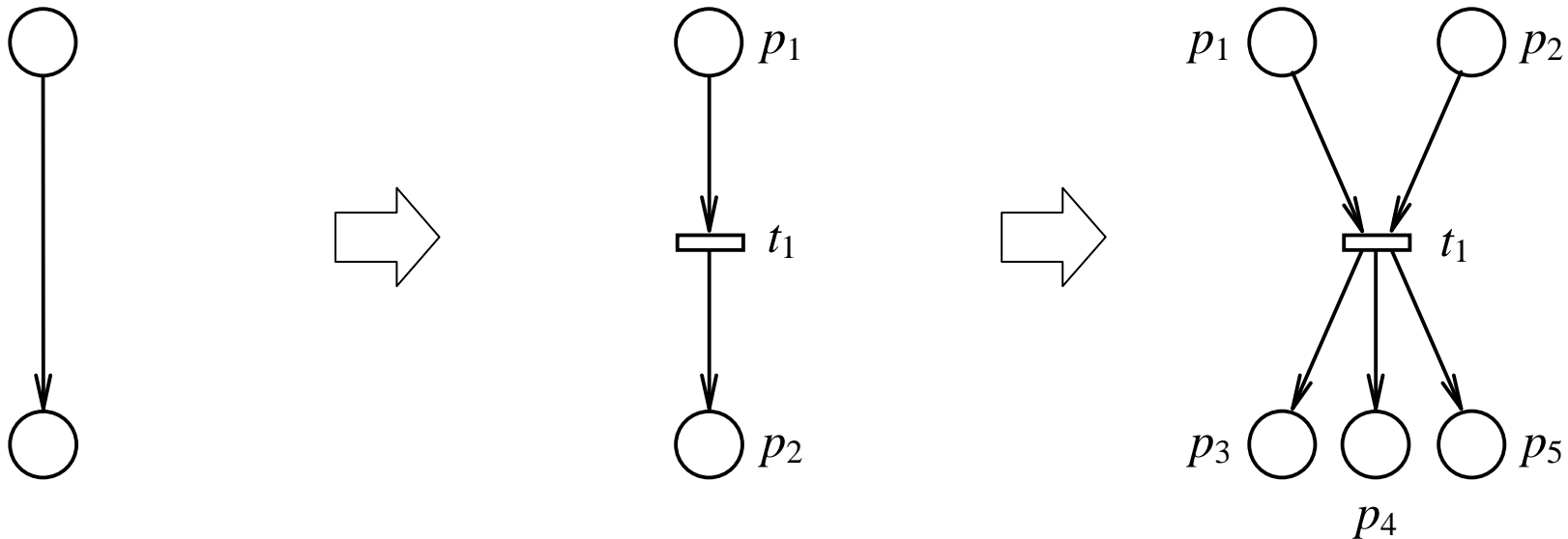
Le reti di Petri possono essere impiegate in due modi:

- ▶ *strumento ausiliario di analisi*: si modella il sistema con una rete di Petri e si analizza quest'ultima per verificare la correttezza del progetto del sistema
- ▶ *strumento diretto di progetto*: l'intero processo di specifica e progetto viene svolto con reti di Petri; occorrono tecniche di traduzione delle reti in sistemi adatti per un'implementazione diretta

## Estensione del concetto di transizione

Negli automi una transizione descrive l'evoluzione da uno stato (globale) ad un altro stato (globale).

Si può estendere il concetto di transizione in modo che possa dipendere da più stati (locali) e influenzare più stati (locali).



In questo modo si giunge naturalmente alla definizione di rete di Petri.

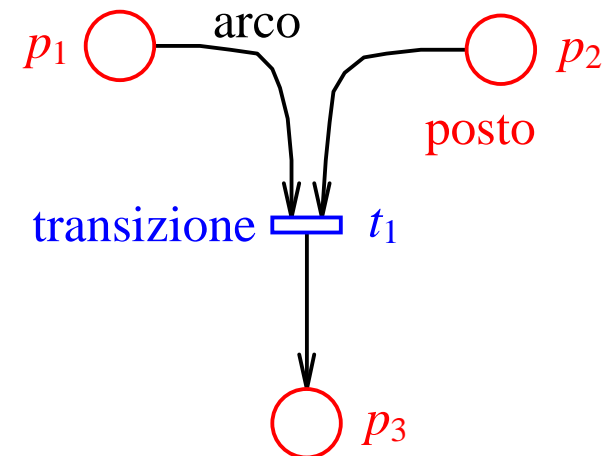
## Definizione di rete di Petri

Una rete di Petri è un *grafo* (orientato) *bipartito*, ovvero con due tipi di nodi:

- ▶ i *posti*, rappresentati con dei cerchi
- ▶ le *transizioni*, rappresentate con barre

I nodi sono collegati tra loro mediante

- ▶ *archi* orientati



Sono possibili solo collegamenti tra nodi di tipo diverso.

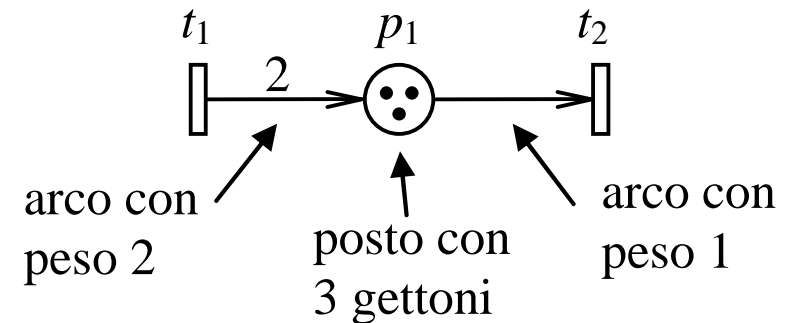
Il *pre-set* di un nodo della rete di Petri è l'insieme dei nodi (di tipo opposto) immediatamente a monte del nodo stesso. Si indica il pre-set di  $x$  con  $\bullet x$ .

Il *post-set* è l'insieme di nodi (di tipo opposto) immediatamente a valle. Si indica il post-set di  $x$  con  $x\bullet$ .

La definizione di rete che abbiamo fornito descrive la sua struttura topologica, ma non è sufficiente per definire un'evoluzione dinamica.

A questo scopo occorre aggiungere due elementi:

- ▶ la *marcatura*  
ogni posto contiene un numero intero non negativo di *gettoni* (o *marche* o *token*), rappresentati con pallini neri
- ▶ la funzione *peso*  
ad ogni arco è associato un numero intero non negativo, detto peso



L'idea è quella di associare ai posti il ruolo delle variabili di stato del sistema.

La *marcatura* dei posti della rete, in breve descritta da un vettore

$$M = [m_1 \ m_2 \ \dots \ m_{|P|}]^T$$

tale che  $m_i = k$  se il posto  $p_i$  contiene  $k$  gettoni, definisce lo stato globale della rete, che evolve (si modifica) come conseguenza dello *scatto* delle transizioni (associate agli eventi).

Una rete di Petri in cui tutti i pesi degli archi sono unitari viene anche detta *ordinaria* (altrimenti è una rete *generalizzata*).

Come in tutti i modelli a eventi discreti l'evoluzione di una rete di Petri è determinata dal verificarsi di eventi.

Un evento è caratterizzato da due attributi:

- ▶ la *possibilità* che esso accada → *abilitazione di una transizione*
- ▶ l'*effetto* del suo accadimento sul sistema → *scatto di una transizione* (abilitata)

Abilitazione di una transizione:

- ▶ affinché una transizione sia nella condizione di potere scattare (transizione *abilitata*) devono essere verificate alcune condizioni, che dipendono dal peso degli archi in ingresso alla transizione e dalla marcatura dei posti in ingresso alla transizione
- ▶ i posti in ingresso a una transizione sono associati a *pre-condizioni* per il verificarsi dell'evento associato alla transizione
- ▶ poiché la marcatura della rete ne rappresenta lo stato, l'abilitazione di una transizione dipende dalla marcatura corrente

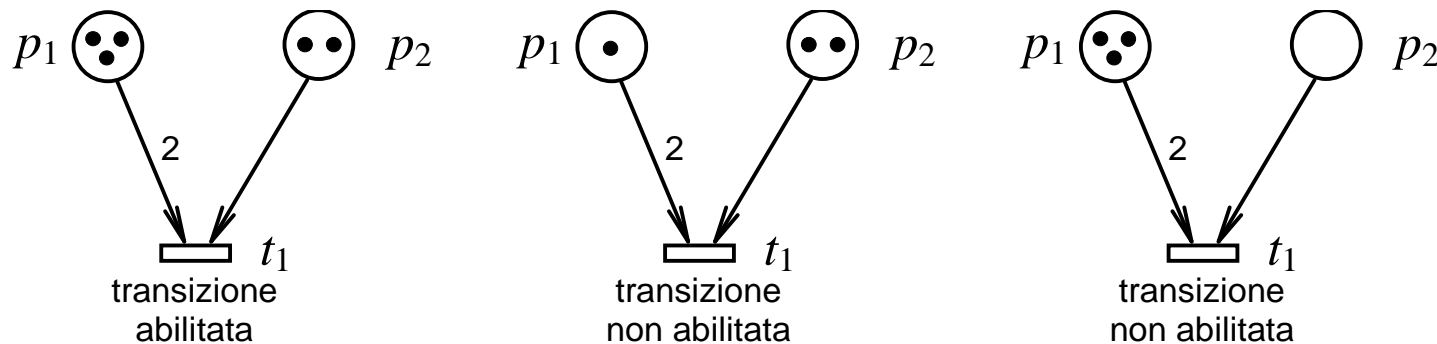
## Scatto di una transizione:

- ▶ lo scatto di una transizione abilitata determina il *consumo* di gettoni nei posti a monte e la *generazione* di gettoni nei posti a valle
- ▶ i posti in *uscita* sono associati a post-condizioni dell’evento
- ▶ l’effetto dello scatto di una transizione determina un cambiamento della marcatura della rete

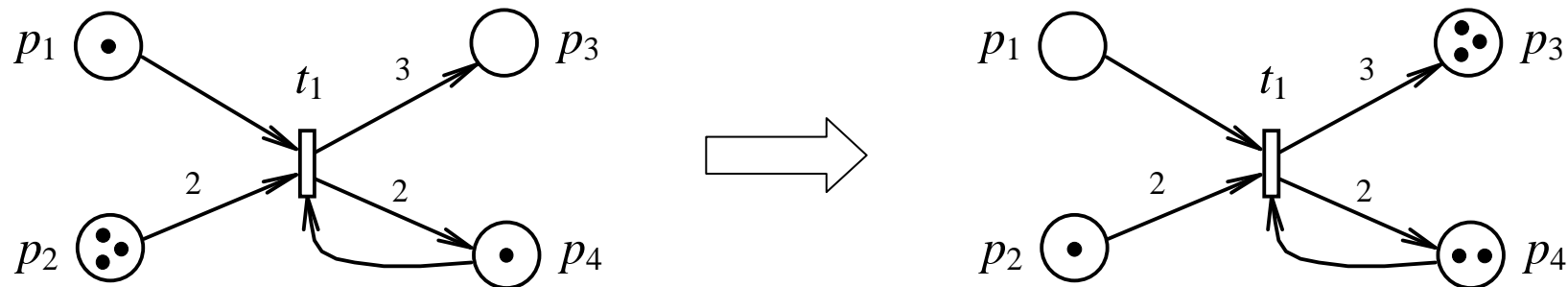
La definizione di opportune *regole di evoluzione* della rete rende quest’ultima un *modello formale*, cioè eseguibile e valutabile analiticamente.

## Evoluzione della rete

Una transizione  $t$  è *abilitata* in una certa marcatura  $M$  ( $M [ t >$ ) se ogni posto a monte di  $t$  ha un numero di gettoni almeno pari al peso dell'arco che lo collega a  $t$ .



Lo *scatto* di una transizione  $t$  abilitata in  $M$  porta la rete in uno stato nuovo  $M^*$  ( $M [ t > M^*$ ), ottenuto rimuovendo da [aggiungendo ad] ogni posto a monte [valle] di  $t$  un numero di gettoni pari al peso dei corrispondenti archi collegati.

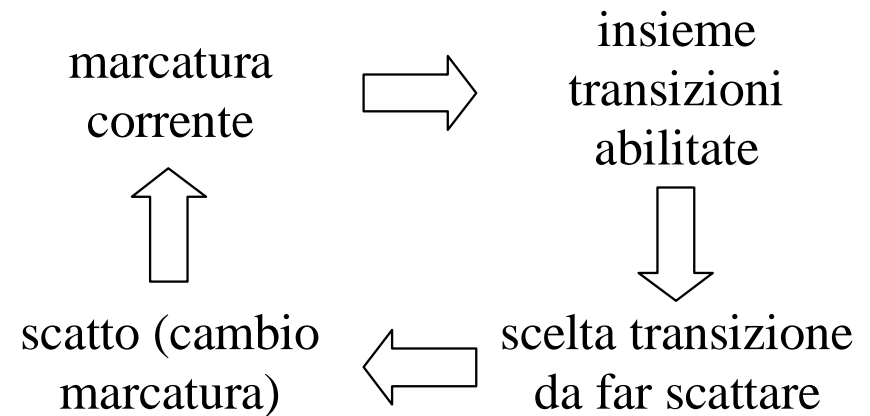


## Interpretazione del concetto di scatto:

- ▶ nel corso dell’evoluzione della rete, lo scatto delle transizioni provoca un “flusso” di gettoni, ma è più corretto interpretare il concetto di scatto affermando che i gettoni eliminati dai posti di pre-set *scompaiano* e che si *creino* gettoni nei posti di post-set
  - ▼ in generale, i gettoni creati possono essere in numero diverso rispetto a quelli eliminati
  - ▼ il numero di gettoni complessivo della rete non rimane necessariamente invariato!
- ▶ lo scatto di una transizione è un fenomeno *locale* della rete
  - ▼ l’abilitazione dipende solo dalla marcatura dei posti collegati a monte della transizione e lo scatto influenza la marcatura di tutti i posti collegati con essa
- ▶ lo scatto di una transizione è *indivisibile*
  - ▼ concettualmente si può immaginare che sia istantaneo

## Cosa succede se più transizioni sono abilitate in una certa marcatura?

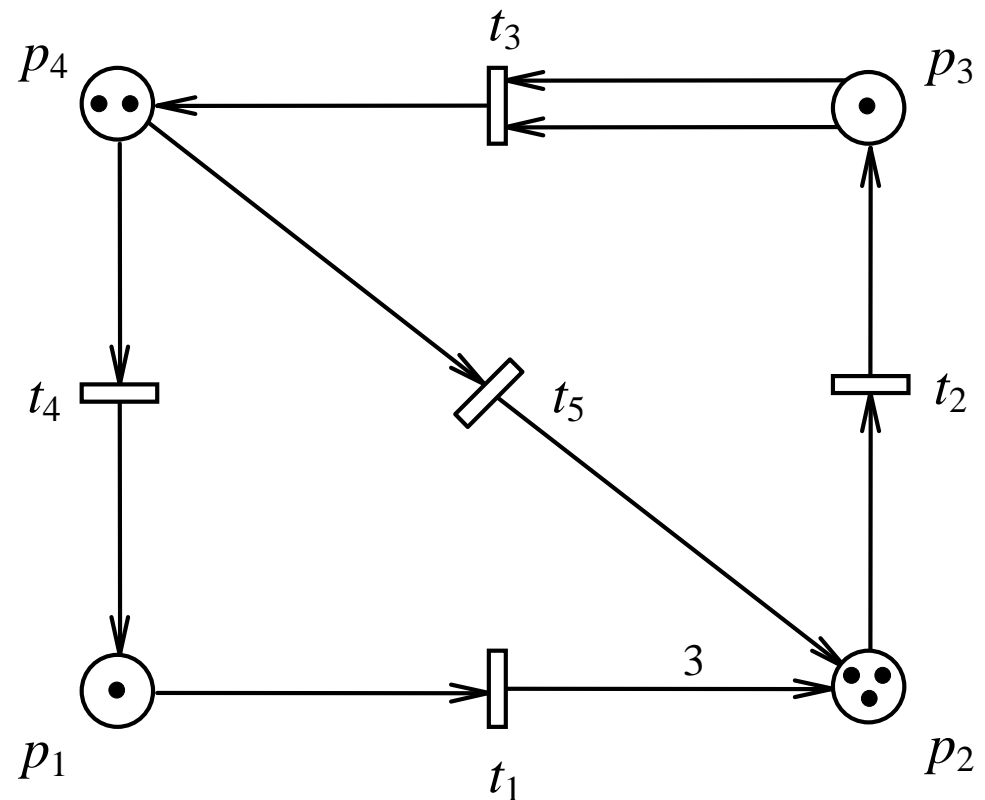
- ▶ nelle reti standard si assume, per convenzione, che scatti *una sola* transizione, scelta *a caso* (*non determinismo*)
- ▶ il fatto che scatti una transizione alla volta preserva la *località* dell’evoluzione della rete (del resto due eventi scorrelati non accadono mai contemporaneamente”)
- ▶ la scelta casuale della transizione che scatta ha senso in un modello concettuale di un problema, ma non va bene in fase di implementazione; si immagini ad esempio di dover progettare un controllore: occorre fornire regole certe di evoluzione!
- ▶ una regola alternativa (molto diffusa nei programmi per la simulazione di reti di Petri) stabilisce che transizioni abilitate ma non in conflitto scattino tutte; quanto a quelle in conflitto, vi sono molte convenzioni diverse:
  - ▼ scelta casuale
  - ▼ priorità
  - ▼ ecc.



## Esempio

Si consideri la rete posti-transizioni rappresentata in figura.

- ❶ Si determinino i pre-set e i post-set di  $p_2$ ,  $p_4$ ,  $t_3$ .
- ❷ Si dica quali sono le transizioni abilitate.
- ❸ Si determinino le marcature conseguenti allo scatto di ciascuna delle transizioni abilitate (trovate al punto precedente).
- ❹ Si trovi una sequenza ammissibile di transizioni che porti la rete di Petri dalla marcatura iniziale alla marcatura  $[0 \ 3 \ 3 \ 3]^T$ .
- ❺ Si verifichi che le sequenze  $(t_1, t_2, t_3, t_4, t_5)$ ,  $(t_2, t_3, t_5, t_4, t_1)$  e  $(t_4, t_2, t_1, t_5, t_3)$  sono ammissibili e portano tutte alla medesima marcatura finale. Commentare quest'ultimo fatto.



## Soluzioni:

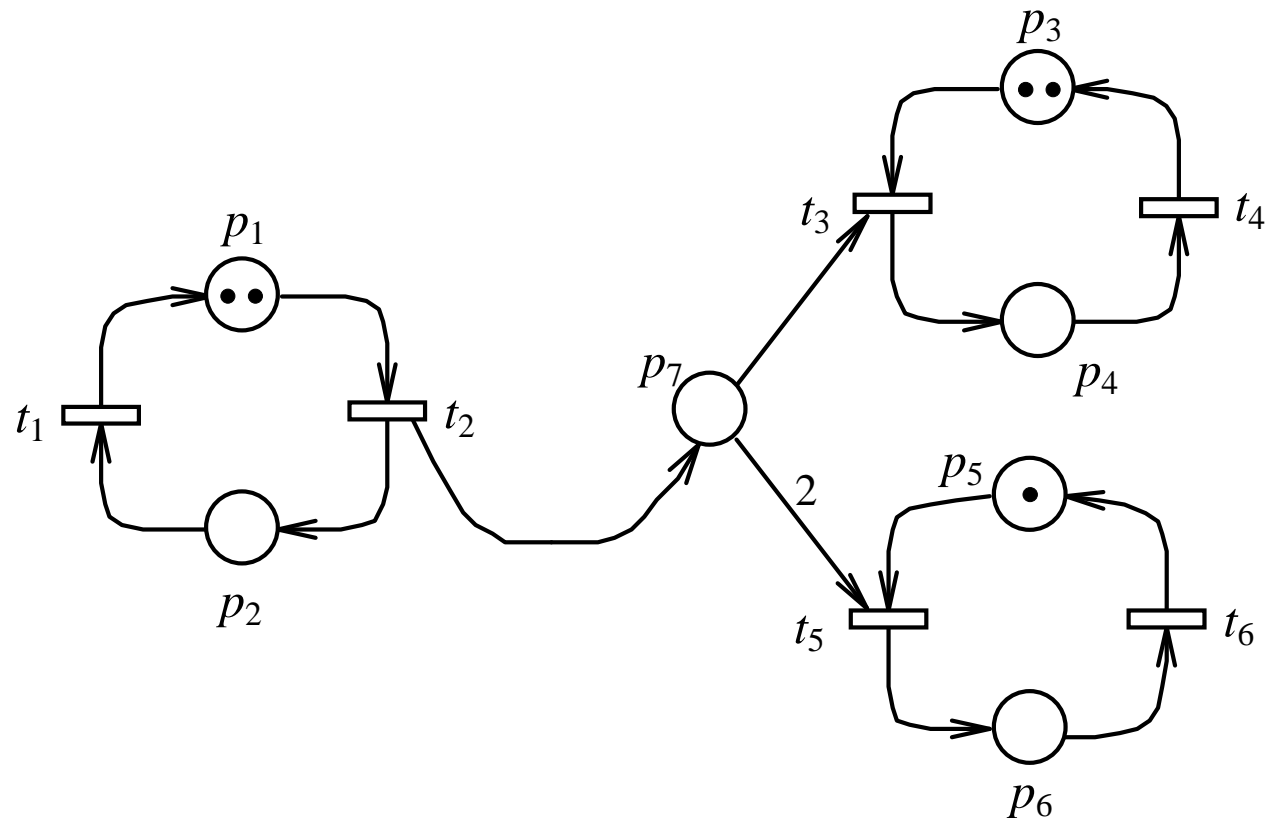
- ①  $\bullet p_2 = \{t_1, t_5\}$ ,  $p_2 \bullet = \{t_2\}$ ,  $\bullet p_4 = \{t_3\}$ ,  $p_4 \bullet = \{t_4, t_5\}$ ,  $\bullet t_3 = \{p_3\}$ ,  $t_3 \bullet = \{p_4\}$
- ②  $t_1, t_2, t_4$  e  $t_5$  sono abilitate, mentre  $t_3$  non è abilitata.
- ③  $M_0 = [1 \ 3 \ 1 \ 2]^T$ ,  $M_0 [t_1 > M_1 = [0 \ 6 \ 1 \ 2]^T$ ,  $M_0 [t_2 > M_2 = [1 \ 2 \ 2 \ 2]^T$ ,  
 $M_0 [t_4 > M_4 = [2 \ 3 \ 1 \ 1]^T$ ,  $M_0 [t_5 > M_5 = [1 \ 4 \ 1 \ 1]^T$ .
- ④ Una sequenza con le caratteristiche richieste è, per esempio:  
 $(t_1, t_4, t_1, t_2, t_2, t_2, t_2, t_2, t_2, t_3, t_3)$ .
- ⑤ Le tre sequenze portano alla marcatura finale  $[1 \ 6 \ 0 \ 1]^T$ .  
 Si ottiene la stessa marcatura perché in ogni sequenza ciascuna transizione scatta lo stesso numero di volte (ad esempio,  $t_1$  scatta una volta sola in ciascuna delle tre sequenze).  
 Poiché l'effetto netto dello scatto di una transizione, cioè la *variazione* della marcatura della rete, è indipendente dalla marcatura a partire dalla quale la si fa scattare (purché la transizione sia abilitata), le tre sequenze determinano la stessa variazione netta nella marcatura della rete.  
 Quindi, poiché sono applicate a partire dalla medesima marcatura e sono tutte ammissibili, non possono che portare al medesimo stato finale.

## Esempio: sistema produttori/consumatori

Il sistema è costituito da uno o più “produttori”, che hanno il compito di produrre “oggetti” e di metterli in un buffer condiviso, e uno o più “consumatori”, che prelevano gli “oggetti” dal buffer e li consumano.

E' un modello standard, che serve per modellizzare tanti casi reali, come ad esempio:

- ▶ due macchine in una linea di assemblaggio che si passano un prodotto
- ▶ due programmi di calcolo che si passano dati/parametri



Produttori = sotto-rete con  $p_1$  e  $p_2$

- ▶  $p_1$  = prodotto pronto (da consegnare)
- ▶  $p_2$  = produttore pronto a produrre
- ▶  $t_1$  = produzione
- ▶  $t_2$  = consegna prodotti al buffer

Consumatori = sotto-rete con  $p_3$  e  $p_4$  e sotto-rete con  $p_5$  e  $p_6$

- ▶  $p_3, p_5$  = consumatore in attesa di prodotti
- ▶  $p_4, p_6$  = consumatore pronto per consumare
- ▶  $t_3, t_5$  = prelievo prodotti dal buffer
- ▶  $t_4, t_6$  = consumo prodotti

I produttori sono 2 (v. gettoni in  $p_1$ ), mentre i consumatori sono 3 distinti in 2 gruppi (con profili di consumo diversi).

Il posto  $p_7$  rappresenta una coda o buffer di prodotti pronti per il consumo.

Gli archi di peso non unitario rappresentano la possibilità di trattare più prodotti contemporaneamente.

Inizialmente, è abilitata solo la transizione  $t_2$  (ci sono due prodotti finiti e i consumatori sono pronti a prelevarli, ma devono aspettare che i prodotti siano consegnati al buffer).

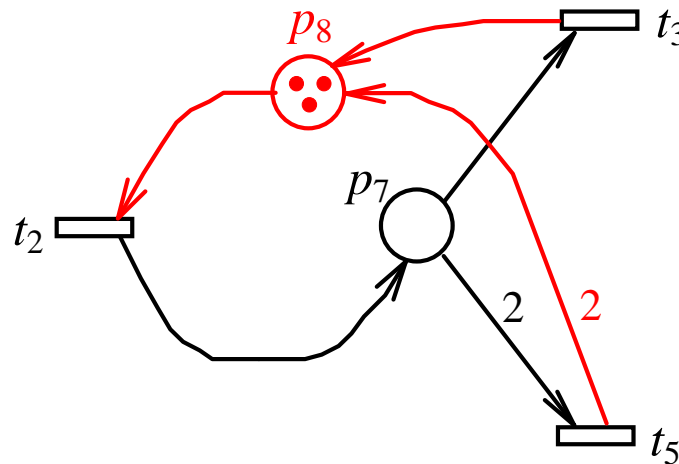
Quando  $t_2$  scatta, si raggiunge una marcatura in cui sono abilitate  $t_1$ ,  $t_2$  e  $t_3$ .

Supponendo che scatti  $t_3$  si ottiene la marcatura finale  $M = [1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0]^T$ .

Osservazioni:

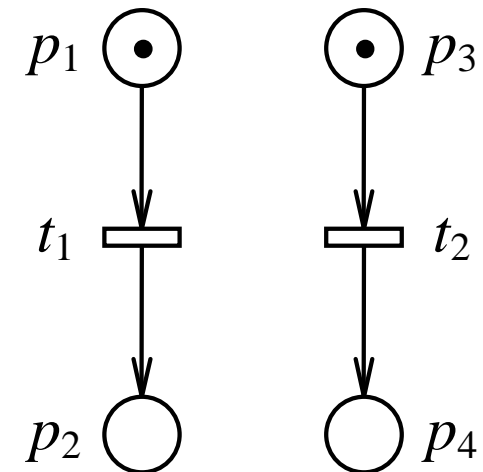
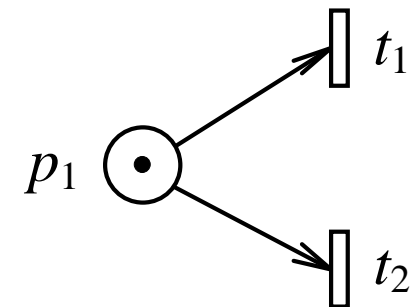
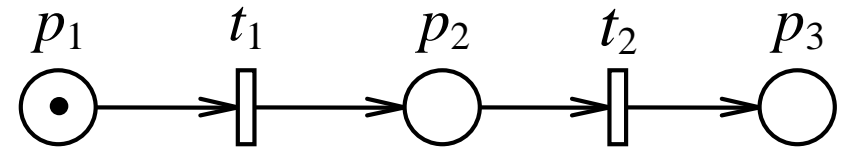
- ▶ le attività di produzione e consumo sono rappresentate da cicli in cui i posti schematizzano gli unici stati logici possibili del processo (i gettoni sono sempre due nella sotto-rete di produzione e la loro localizzazione in un posto piuttosto che nell'altro rappresenta lo stato logico dei due produttori)
- ▶ i cicli di produzione e consumo sono indipendenti tra loro (i cicli di produzione non sono condizionati dai cicli di consumo e anche questi ultimi possono continuare ad evolvere purché ci sia un numero sufficiente di prodotti nel buffer)
- ▶ anche se si assume che due scatti di transizioni non siano “contemporanei”, si comprende come questo modello realizzi la concorrenza tra processi indipendenti
- ▶ non si distingue tra i due produttori (i gettoni sono uguali!)  
→ attenzione nell'implementazione

- ▶ se cambiasse il numero di produttori o consumatori, basterebbe modificare il numero di gettoni nella rete; se si aggiungessero altri tipi di produttori o consumatori, basterebbe duplicare le relative sotto-reti e collegarle opportunamente al posto buffer → *modularità* (cfr. automi!)
- ▶ con un’opportuna sequenza di scatti di transizioni è sempre possibile ritornare allo stato iniziale (*reversibilità*)  
→ proprietà tipica dei processi produttivi ciclici
- ▶ in questo modello il buffer è illimitato (se il produttore è più veloce del consumatore, si accumulano prodotti nel buffer); per limitarne la capacità si può inserire un *posto complementare* (marcato con un numero di gettoni pari alla capacità del buffer)



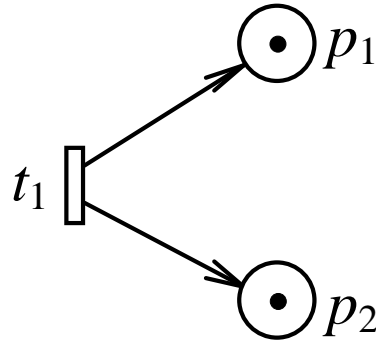
## Strutture modellistiche fondamentali

- ▶ due transizioni  $t_1$  e  $t_2$  si dicono in *sequenza*, e  $t_1$  precede  $t_2$  in una data marcatura, se con  $t_1$  abilitata e  $t_2$  no, lo scatto di  $t_1$  abilita  $t_2$
- ▶ due transizioni  $t_1$  e  $t_2$  si dicono in
  - ▼ *conflitto strutturale*  
se hanno almeno un posto d'ingresso in comune
  - ▼ *conflitto effettivo*  
se sono in conflitto strutturale e se sono entrambe abilitate in una marcatura  $M$ , ma lo scatto di una disabilita l'altra
- ▶ due transizioni  $t_1$  e  $t_2$  si dicono in
  - ▼ *concorrenza strutturale*  
se non condividono nessun posto d'ingresso
  - ▼ *concorrenza effettiva*  
se sono in concorrenza strutturale e se sono entrambe abilitate in una marcatura  $M$  e lo scatto di una non disabilita l'altra

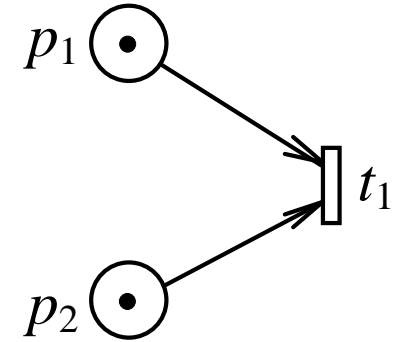


- ▶ due strutture legate al concetto di concorrenza sono

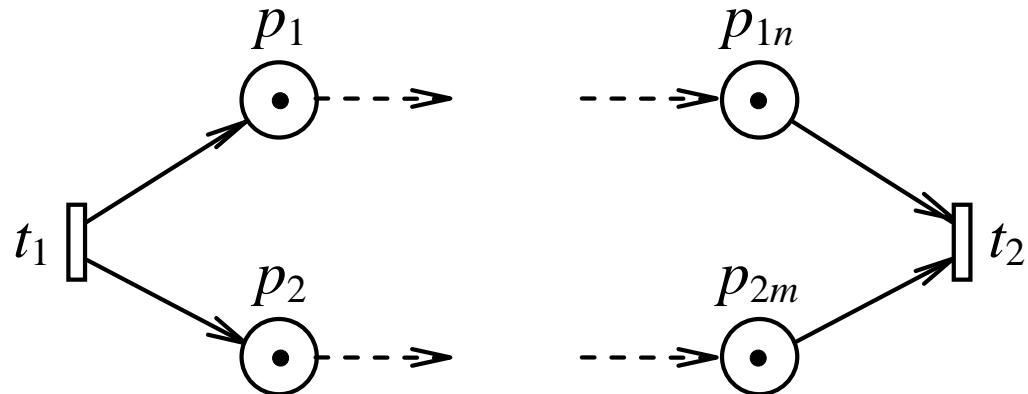
- ▼ la transizione di *inizio concorrenza* (transizione con più di un posto a valle)



- ▼ e quella di *sincronizzazione* (transizione con più di un posto a monte)



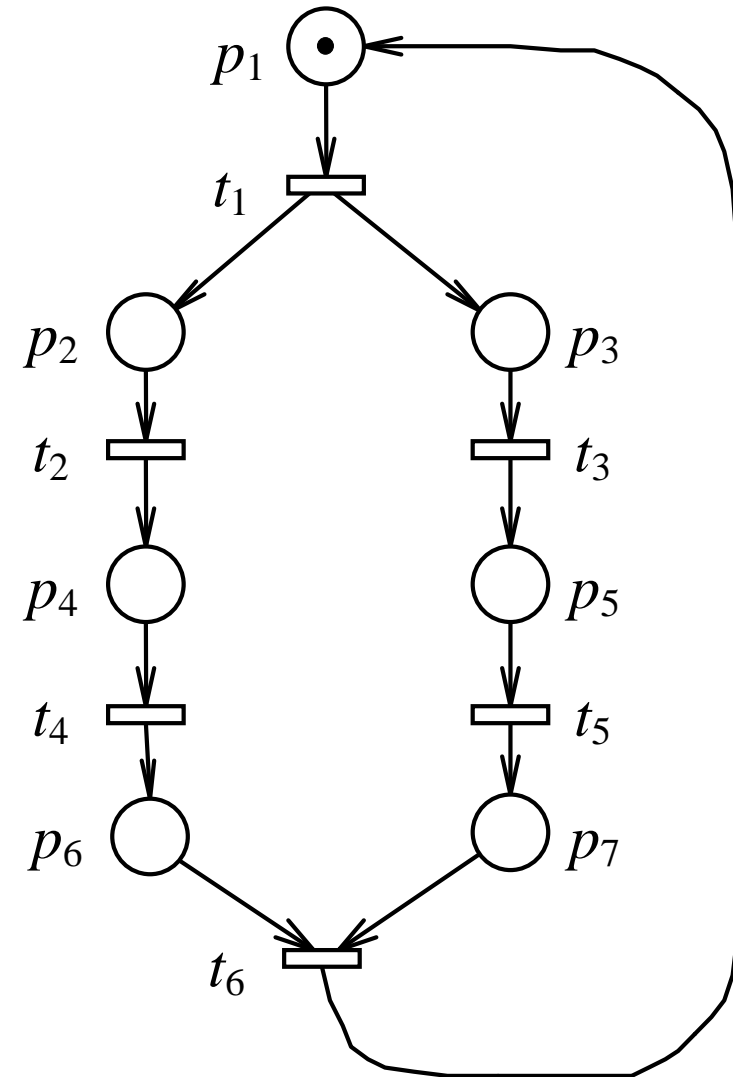
- ▶ tipicamente, due sequenze concorrenti sono attivate dallo scatto di una transizione del primo tipo e la fase di concorrenza termina con lo scatto di una transizione del secondo



## Esempio

Nella rete si riconoscono diverse strutture modellistiche di base:

- ▶ le transizioni  $t_6$   $t_1$   $t_2$   $t_4$  e  $t_6$   $t_1$   $t_3$   $t_5$  costituiscono due sequenze
- ▶ la transizione  $t_1$  è una transizione di concorrenza
- ▶ la transizione  $t_6$  è una transizione di sincronizzazione



## Interpretazione del modello:

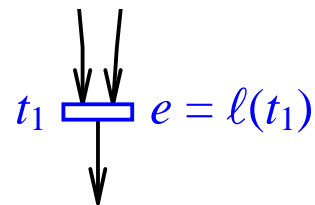
- ▶ se ad ogni posto della rete si associa il significato di esecuzione di un'operazione di un processo produttivo, intendendo che l'operazione  $o_i$  è in esecuzione se il posto  $p_i$  è marcato, la rete rappresenta un processo in cui al completamento dell'operazione  $o_1$  vengono attivate contestualmente le due operazioni  $o_2$  e  $o_3$
- ▶ successivamente, si svolgono in modalità concorrente due sequenze di operazioni,  $o_2 o_4 o_6$  e  $o_3 o_5 o_7$
- ▶ quando *entrambe* le sequenze sono completate, il processo può ricominciare dall'operazione  $o_1$
- ▶ la transizione  $t_6$  opera una sincronizzazione tra le due sotto-sequenze, nel senso che se una delle due termina prima dell'altra, deve comunque aspettarne la conclusione, prima che il processo possa proseguire con  $o_1$

## Rete di Petri etichettata

Per descrivere con una rete di Petri un sistema a eventi discreti il cui insieme degli eventi sia  $E$ , occorre dotare la definizione di un'ulteriore funzione, detta *funzione di etichetta delle transizioni*:

$$\ell : T \rightarrow E$$

che associ gli eventi alle transizioni.

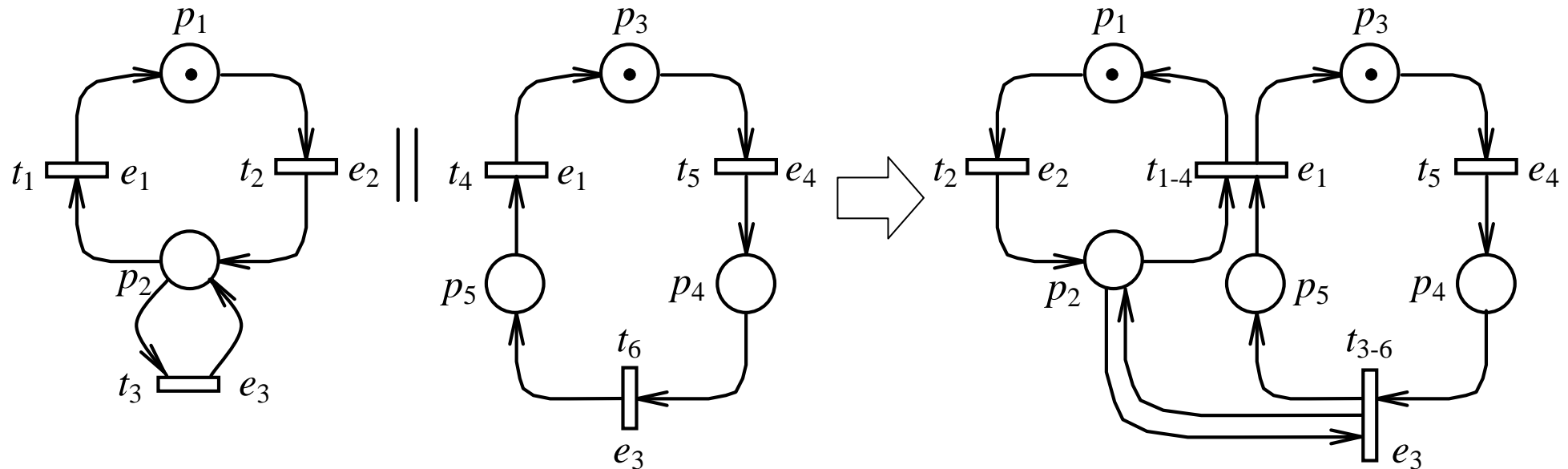


In questo modo lo scatto di una transizione corrisponde fisicamente ad un evento.

Si noti che lo stesso evento può essere associato a più transizioni.

La rete così ottenuta prende il nome di *rete di Petri etichettata*.

## Composizione sincrona di reti di Petri (etichettate)



Diversamente dal caso degli automi, la composizione sincrona di reti di Petri determina una crescita *lineare* del numero dei posti.

Nell'esempio,  $N_1$  ha 2 posti e  $N_2$  ne ha 3.

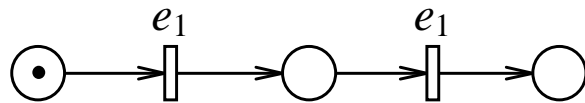
La rete risultante dalla composizione sincrona ha  $2+3 = 5$  posti.

Una rete di Petri progettata per composizione sincrona di moduli semplici prende il nome di *rete di Petri sincronizzata*.

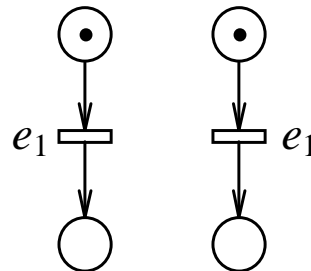
## Reti di Petri sincronizzate

La simulazione di reti sincronizzate (con eventi) richiede qualche cautela aggiuntiva nell'interpretazione delle regole di evoluzione. In alcuni casi, non è chiaro quale marcatura venga raggiunta dopo l'accadimento di un evento:

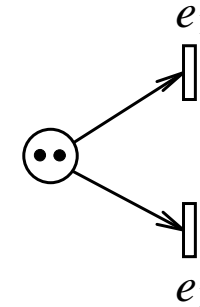
scatti multipli concatenati



scatti multipli indipendenti



scatti di transizioni in conflitto



### Algoritmi di interpretazione:

- ▶ approccio *sincrono*:  
scattano tutte le transizioni che possono scattare quando accade un evento
- ▶ approccio *asincrono*:  
scatta una sola transizione (l'evento è consumabile solo per uno scatto)

## Proprietà di base delle reti di Petri

Le proprietà di base hanno interpretazioni molto utili ai fini della modellizzazione e del controllo.

La prima proprietà che ci interessa è se sia possibile ottenere una data marcatura (cioè un determinato stato) a partire da un'altra.

Ad esempio può essere richiesto di verificare se una marcatura che causa un blocco del sistema (deadlock) sia raggiungibile dalla marcatura corrente.

Raggiungibilità:

- ▶ una marcatura  $M^*$  si dice *raggiungibile* a partire da una data marcatura  $M$  se esiste almeno una sequenza di transizioni tali che facendole scattare nell'ordine a partire da  $M$  si ottenga  $M^*$  (devono essere tutte abilitate al momento giusto)
- ▶ si definisce *insieme di raggiungibilità*  $R(N, M_0)$  di una rete  $N$  con marcatura iniziale  $M_0$  l'insieme tale che:
  - ▼  $M_0 \in R(N, M_0)$
  - ▼ se  $M^* \in R(N, M_0)$  ed  $\exists t \in T$  tale che  $M^* [t > M^{**}$ , allora  $M^{**} \in R(N, M_0)$

Nei processi produttivi è importante poter ripristinare un funzionamento nominale in ogni situazione (conseguente a guasti, errori, ecc.).

Inoltre, spesso tali processi consistono di sequenze cicliche di operazioni, al termine delle quali si torna in uno stato iniziale.

Reversibilità:

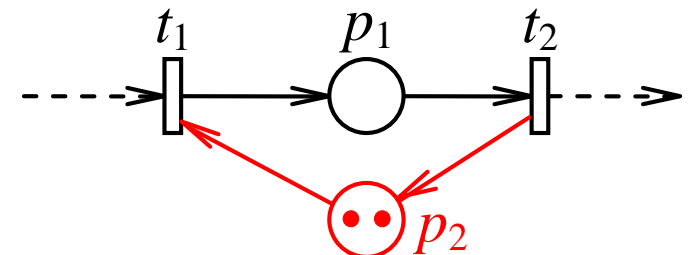
- ▶ una rete di Petri  $N$  con marcatura iniziale  $M_0$  è detta essere *reversibile* se  $\forall M \in R(N, M_0), M_0 \in R(N, M)$ , ovvero se per ogni marcatura  $M$  raggiungibile da  $M_0$  si ha che  $M_0$  è raggiungibile da  $M$
- ▶ una marcatura  $M$  della rete è detta essere uno *home state* se  $\forall M^* \in R(N, M_0), M \in R(N, M^*)$ , ovvero se per ogni marcatura  $M^*$  raggiungibile da  $M_0$  si ha che  $M$  è raggiungibile da  $M^*$

I gettoni nei posti sono spesso usati per rappresentare variabili legate al conteggio di oggetti (pezzi che transitano in una macchina, posti di un magazzino, ecc.).

E' allora importante accertare che il numero di gettoni che può venirsi a trovare in un posto a seguito dell'evoluzione della rete non superi una soglia limite.

Limitatezza:

- ▶ un posto  $p$  di una rete si dice *k-limitato* se in tutte le marcature raggiungibili a partire dalla marcatura iniziale il numero di gettoni presenti nel posto non supera mai un valore prefissato  $k$
- ▶ una rete si dice *k-limitata* se tutti i suoi posti sono *k-limitati*
- ▶ una rete si dice *limitata* se è *k-limitata* per qualche valore di  $k$
- ▶ se una rete è limitata, non può avere un numero illimitato di marcature distinte; quindi, è equivalente ad un automa a stati finiti
- ▶ si può sempre imporre la limitatezza di un posto aggiungendo un posto *complementare*
- ▶ una rete è illimitata se e solo se  $\exists \sigma_1, \sigma_2$  t.c.  
 $M_0 [ \sigma_1 > M_1 [ \sigma_2 > M_2$  con  $M_1 \leq M_2$  e  $M_1 \neq M_2$ .
- ▶ una rete si dice *strutturalmente limitata* se è limitata per ogni marcatura iniziale



## Binarietà o sicurezza:

- ▶ una rete limitata con  $k = 1$  si dice *binaria* o *sicura*
- ▶ in una rete binaria ogni posto può contenere al più un gettone
  - ▼ tutte le marcature, compresa naturalmente quella iniziale, contengono solo ‘0’ o ‘1’
- ▶ tali reti si prestano bene al coordinamento di unità semplici
  - ▼ un gruppo di posti rappresenta tutti gli stati possibili di un dispositivo e solo uno dei posti alla volta è marcato con un gettone, a seconda dello stato in cui si trova il dispositivo

## Conservatività:

- ▶ nelle reti di Petri si modellizzano sistemi di allocazione di risorse: poiché i gettoni vengono usati per rappresentare le risorse è importante che si *conservino*
- ▶ una rete con marcatura iniziale  $M_0$  si dice *conservativa con riferimento ad un vettore peso*  $W = [w_1 \ w_2 \ \dots \ w_{|P|}] \geq 0$ , se  $\forall M \in R(N, M_0)$  vale l'equazione:

$$\sum_i w_i m_i = \sum_i w_i m_{0i}$$

ovvero

$$WM = WM_0$$

- ▶ una rete si dice *conservativa* se è conservativa con riferimento ad un vettore peso strettamente positivo ( $W > 0$ )
- ▶ una rete si dice *strettamente conservativa* se è conservativa con riferimento ad un vettore peso  $W = [1 \ 1 \ \dots \ 1]$
- ▶ una rete conservativa è anche strutturalmente limitata

Altre importanti proprietà di una rete di Petri hanno a che fare con la vivezza: la rete può sempre continuare ad evolvere a partire da qualunque stato in cui si possa venire a trovare, oppure si blocca (tutta o in parte)?

Vivezza:

- ▶ una *transizione*  $t$  si dice *viva* se e solo se per ogni marcatura  $M$  raggiungibile dalla marcatura iniziale, esiste una marcatura  $M^*$  raggiungibile da essa, tale che  $t$  è abilitata in  $M^*$
- ▶ una *rete* si dice *viva* se e solo se tutte le sue transizioni sono vive
- ▶ se una transizione  $t$  è viva, esiste una sequenza di transizioni che porta da  $M_0$  ad  $M^*$  in cui essa è abilitata; ora, se  $t$  scatta ci si troverà in una marcatura  $M^{**}$ , che per definizione è raggiungibile da  $M_0$ ; per definizione di transizione viva, esisterà anche per  $M^{**}$  una marcatura raggiungibile in cui  $t$  risulta nuovamente abilitata, e così via
  - ▼ una transizione viva può scattare *infinite* volte!
  - ▼ in una rete viva tutte le transizioni possono scattare *infinite* volte!
- ▶ la vivezza è quindi una condizione estremamente forte; per questo motivo non esistono condizioni necessarie e sufficienti per la vivezza di reti generiche

La vivezza si può definire anche in termini di marcature:

- ▶ una *marcatatura*  $M$  si dice *viva* se e solo se per ogni transizione  $t$  esiste una marcatura  $M^*$  raggiungibile da  $M$  tale che  $t$  è abilitata in  $M^*$
- ▶ a partire da una marcatura viva è *possibile* far scattare una qualunque delle transizioni, ma può anche darsi che si raggiungano marcature in cui alcune o tutte le transizioni della rete non siano abilitate
- ▶ una *marcatatura*  $M$  si dice *morta* (*deadlock*) se e solo se nessuna transizione è abilitata in  $M$
- ▶ una *rete* è *viva* se e solo se tutte le sue marcature raggiungibili a partire dalla marcatura iniziale sono vive
  - ▼ se una rete ammette una marcatura raggiungibile morta, non è viva (né reversibile), ma
  - ▼ l'assenza di marcature morte (rete *deadlock-free*) non è sufficiente a garantire la vivezza
  - ▼ in effetti, una rete non deve essere necessariamente viva per continuare ad evolvere: un blocco di parte delle transizioni della rete può consentire ugualmente al resto del sistema di evolvere indefinitamente (rete non viva, ma deadlock-free)

NB. Reversibilità, limitatezza e vivezza sono proprietà *indipendenti*.

## Classi particolari di reti di Petri

Si introducono varie restrizioni della definizione di rete di Petri, con vincoli sulla topologia del grafo.

Le classi di reti risultanti hanno un minor potere rappresentativo, ma per esse si riescono a dimostrare più risultati analitici.

Classi di reti:

- ▶ macchina a stati (*state machine*, SM)
- ▶ grafo marcato (*marked graph*, MG)
- ▶ rete a scelta libera (*free choice Petri net*, FC)
- ▶ rete a scelta libera estesa (*extended free choice Petri net*, EFC)
- ▶ rete a scelta libera asimmetrica (*asymmetric free choice Petri net*, AFC)

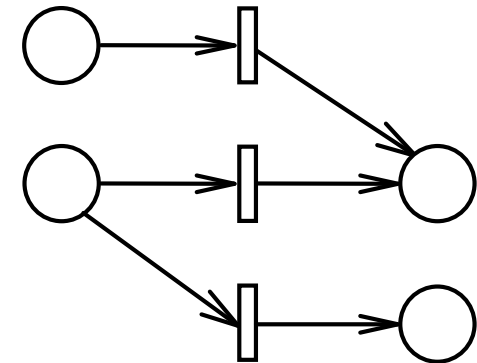
In questo contesto, considereremo solo reti di Petri ordinarie (rete con solo archi di peso 1). Le reti non ordinarie sono chiamate *reti generalizzate*.

## Macchina a stati

Una *macchina a stati* (*state machine*, SM) è una rete di Petri ordinaria in cui ogni transizione è limitata ad avere un solo posto in ingresso ed un solo posto in uscita.

Proprietà:

- ▶ il numero di gettoni nella rete non cambia mai, quindi la macchina a stati è una rete di Petri strettamente conservativa
- ▶ l'insieme di raggiungibilità è finito, quindi una macchina a stati è equivalente ad un *automa a stati finiti*
- ▶ una macchina a stati contenente inizialmente solo un gettone è una rete *binaria*
- ▶ una macchina a stati è viva se e solo se è *fortemente connessa* (cioè se è possibile andare da un qualunque nodo ad un altro, seguendo la relazione di flusso) e se ha almeno un gettone



Una macchina a stati può rappresentare conflitti (posto con più transizioni in uscita), ma non può modellizzare la creazione e la distruzione di gettoni necessaria per rappresentare la concorrenza e la sincronizzazione.

## Grafo marcato

Un *grafo marcato* (*marked graph*, MG) è una rete di Petri ordinaria in cui ogni posto ha una sola transizione di ingresso ed una sola transizione di uscita.

Proprietà:

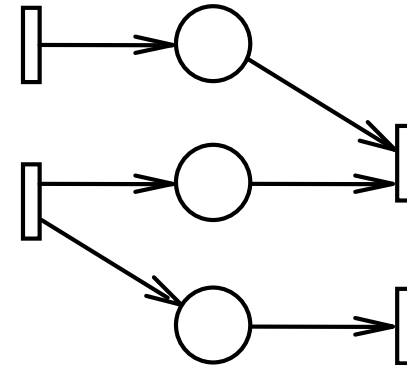
- ▶ un grafo marcato è vivo se e solo se ogni suo ciclo contiene almeno un posto marcato
- ▶ un grafo marcato è strutturalmente limitato se e solo se la rete è fortemente connessa
- ▶ un ciclo è una sequenza di nodi  $x_1 x_2 \dots x_n$ ,  $n \geq 2$ , tale che

$$\nabla x_i \bullet \supseteq \{x_{i+1}\}, i = 1, \dots, n-1, \text{ e}$$

$$\nabla x_n \bullet \supseteq \{x_1\},$$

ovvero tale che il successore dell'ultimo nodo della sequenza è il primo nodo

Un grafo marcato può modellizzare la concorrenza e la sincronizzazione, ma non può rappresentare conflitti.



## Rete a scelta libera

Due transizioni  $t_1$  e  $t_2$  si dicono in conflitto (effettivo) se solo una delle due può scattare, ma non tutte e due, e concorrenti se possono scattare entrambe in qualunque ordine senza conflitti.

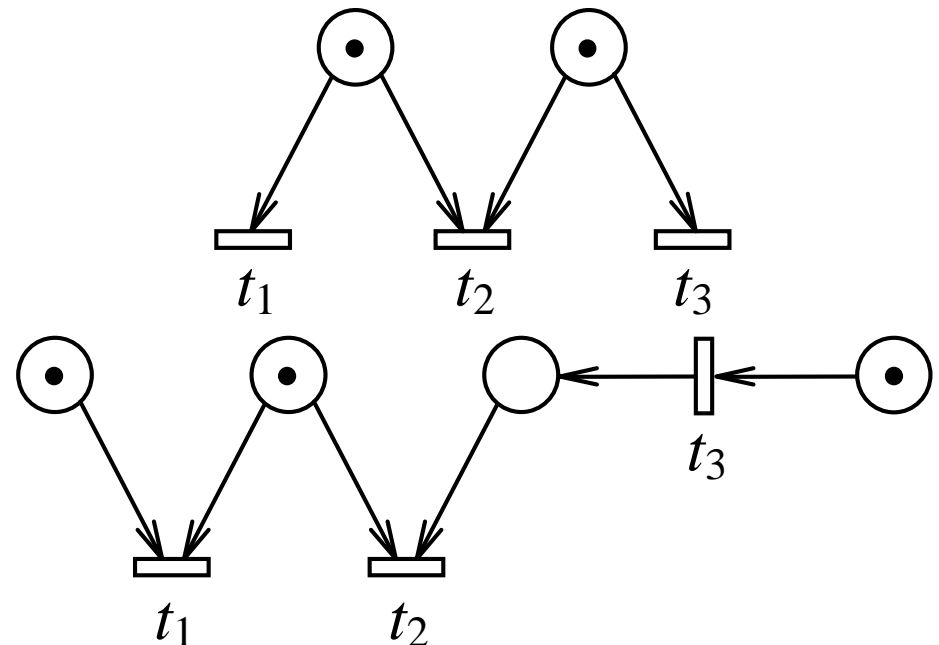
Se conflitto e concorrenza sono mescolati si ha una situazione di *confusione*.

► *confusione simmetrica:*

$t_1$  e  $t_3$  sono concorrenti e, nello stesso tempo, sono entrambe in conflitto con  $t_2$

► *confusione asimmetrica:*

$t_1$  è concorrente con  $t_3$ , ma andrà in conflitto con  $t_2$  se  $t_3$  scatterà prima di  $t_1$

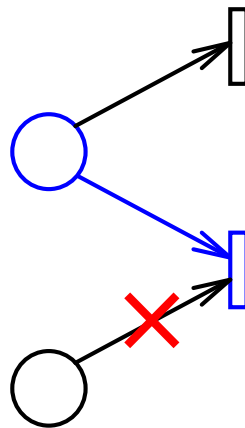


In entrambi i casi possono esistere delle marcature in cui di una coppia di transizioni in conflitto (p.es.  $t_1$  e  $t_2$ ), una è abilitata e l'altra no.

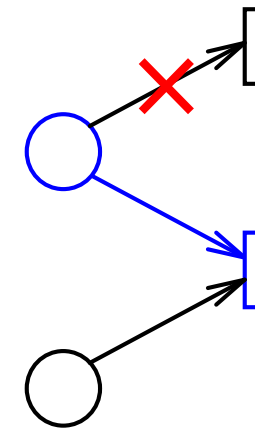
In altre parole, la scelta di quale transizione di un conflitto far scattare non è libera, ma dipende anche dalla marcatura di altri posti della rete (diversi da quelli nel pre-set di entrambe le transizioni in conflitto).

Una *rete a scelta libera* (*free choice Petri net*, FC) è tale che per ogni arco da un posto ad una transizione:

- ▶ o quel posto è l'*unico posto in ingresso* alla transizione (non c'è sincronizzazione)



- ▶ o quella transizione è l'*unica transizione in uscita* da quel posto (non ci sono conflitti)



In sintesi,  $\forall p$  tale che  $|p\bullet| > 1$  (conflitto) deve valere la seguente condizione:

$$\bullet(p\bullet) = \{p\}$$

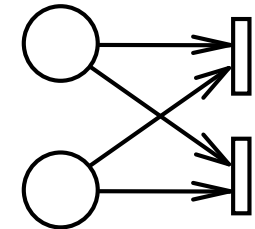
che impone che non ci possa essere una sincronizzazione sulle transizioni nel post-set di  $p$ .

## Osservazioni:

- ▶ nelle reti a scelta libera è possibile rappresentare sia conflitti che concorrenza, ma con alcune limitazioni
- ▶ in particolare, non si possono generare marcature in cui transizioni in conflitto non siano entrambe abilitate o entrambe disabilitate
- ▶ ciò implica che la risoluzione dei conflitti effettivi è sempre libera, ovvero è sempre possibile far scattare una qualunque delle transizioni in conflitto
- ▶ macchine a stati e grafi marcati sono reti a scelta libera, ma non vale il viceversa

## Rete a scelta libera estesa e a scelta libera asimmetrica

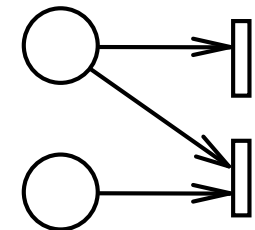
Una *rete a scelta libera estesa* (*extended free choice Petri net*, EFC) è tale che se due posti hanno una transizione di uscita in comune, allora quei posti hanno le *stesse* transizioni d'uscita.



In altre parole, se  $p_i \bullet \cap p_j \bullet \neq \emptyset$ , allora deve valere  $p_i \bullet = p_j \bullet$ .

Una rete a scelta libera è anche una rete a scelta libera estesa, ma non vale il viceversa.

Una *rete a scelta asimmetrica* (*asymmetric free choice Petri net*, AFC) è una rete di Petri tale che se due posti hanno delle transizioni di uscita in comune, l'insieme di transizioni di uscita di uno dei due posti contiene quello dell'altro.



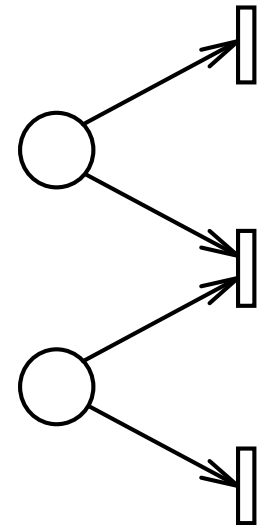
In altre parole, se  $p_i \bullet \cap p_j \bullet \neq \emptyset$ , allora deve valere  $p_i \bullet \supseteq p_j \bullet$  oppure  $p_i \bullet \subseteq p_j \bullet$ .

Una rete a scelta libera estesa è anche una rete a scelta libera asimmetrica, ma non vale il viceversa.

## Rete ordinaria

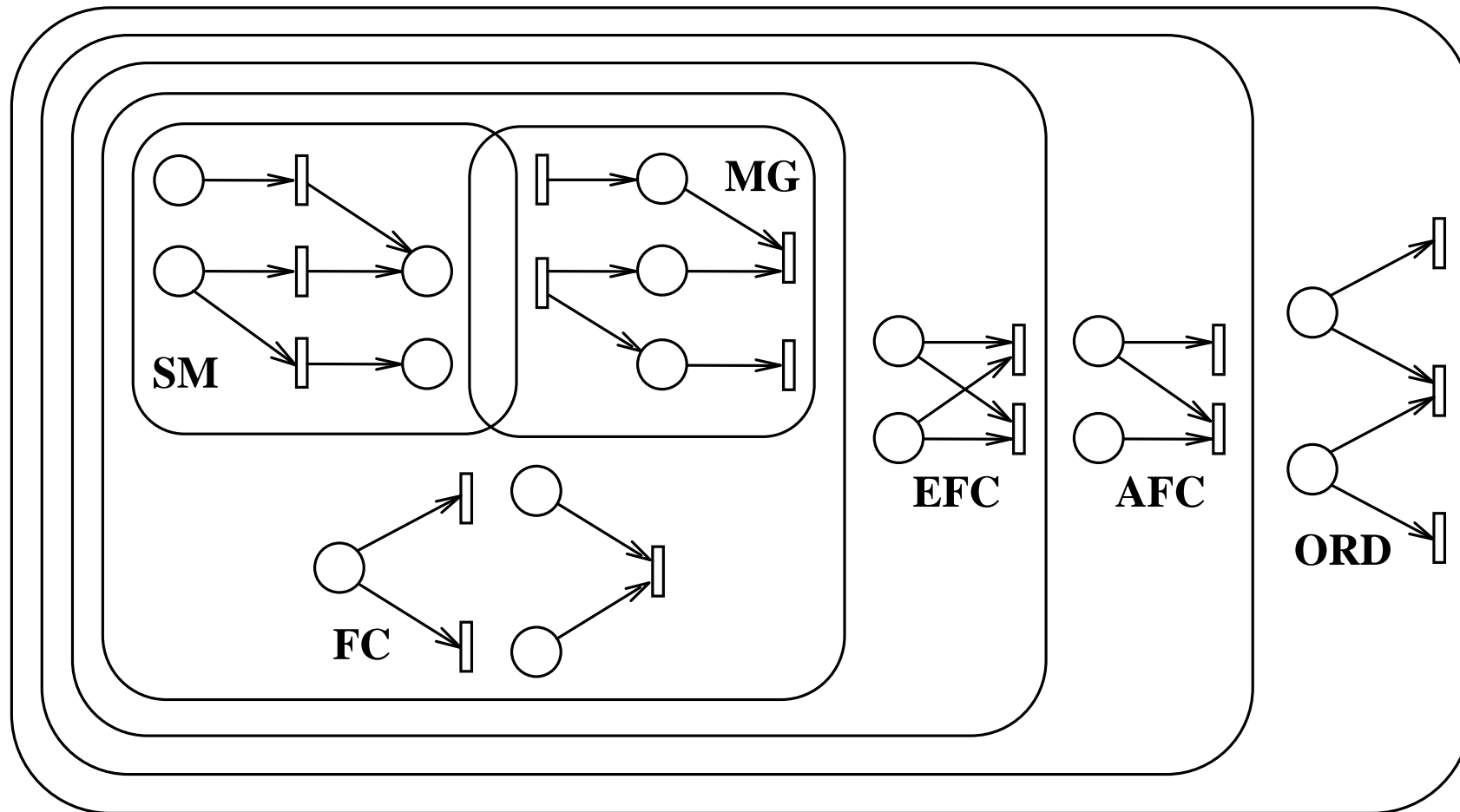
In una rete di Petri ordinaria è ammissibile anche una struttura come quella della figura a lato, che è incompatibile con tutte le definizioni precedenti.

Una rete a scelta libera asimmetrica è anche una rete ordinaria, ma non vale il viceversa.



## Legami tra le varie tipologie di reti

Reti di Petri ordinarie  $\supset$  AFC  $\supset$  EFC  $\supset$  FC  $\supset$  (SM  $\cup$  MG)



## Estensioni delle reti di Petri

Le reti di Petri ordinarie non includono alcun concetto di tempo: con esse si può descrivere la struttura logica di un sistema, ma non la sua evoluzione temporale.

Esistono vari tipi di *reti di Petri temporizzate*, a seconda delle ipotesi modellistiche adottate. Le estensioni più utilizzate sono le seguenti:

- ▶ ogni transizione rappresenta un evento, il cui verificarsi è perciò istantaneo; ad ogni transizione si associa un intervallo di tempo  $(T_{min}, T_{max})$ , dove  $T_{min}$  [ $T_{max}$ ] rappresenta il tempo minimo [massimo] che deve intercorrere tra l’abilitazione e lo scatto (a meno che la transizione non venga disabilitata nel frattempo)
- ▶ ogni transizione rappresenta un’attività del sistema, che richiede un tempo non nullo per poter essere completata; ad ogni transizione viene associata una durata e lo scatto avviene nei seguenti passi:
  - ❶ i gettoni vengono rimossi dai posti d’ingresso non appena la transizione è abilitata,
  - ❷ la transizione permane nella fase di scatto per tutta la durata,
  - ❸ alla fine della fase di scatto, si ha la produzione di gettoni nei posti d’uscita.
- ▶ ogni posto rappresenta un’attività del sistema; ad ogni posto viene associata una durata, pari al tempo richiesto per completare l’attività

Si parla di *reti di Petri stocastiche* se:

- ▶ si modellizzano gli intervalli di tempo come variabili casuali, oppure se
- ▶ si aggiungono al modello delle distribuzioni probabilistiche per la risoluzione di conflitti.

Solitamente si utilizza la convenzione di associare gli intervalli di tempo alle transizioni.

Nelle reti di Petri standard i gettoni sono completamente indistinguibili, cioè non portano con sé alcuna informazione.

Nelle *reti di Petri ad alto livello*, essi sono associati ad altre informazioni (v. reti di Petri colorate).

Oltre a ciò alle transizioni sono connesse delle condizioni logiche che influenzano lo scatto delle transizioni stesse.