Lezione 8 30/10/2023

Reti di Petri

Introduzione

obiettivi:
"sviluppo di una teoria matematica fondata sui principi della fisica moderna (teoria della relatività e fisica quantistica), che sia una **teoria dei sistemi** in grado di descrivere il **flusso di informazione** e permetta di analizzare **sistemi con organizzazione complessa**"

La concorrenza è data dall'indipendenza di processi diversi, non dal poterli eseguire in modo sparso.

Una rete di Petri può essere rappresentata come una matrice, quindi il comportamento può essere anche simulato in questo modo. (non lo vedremo?)

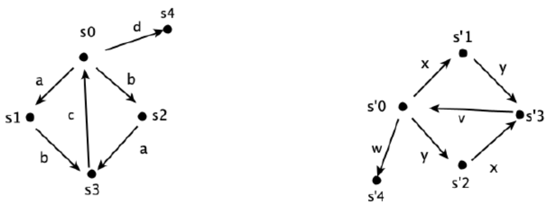
La teoria delle reti di Petri è usata in diversi ambiti applicativi come la specifica di protocolli di comunicazione, il disegno di circuiti asincroni, algoritmi e programmi concorrenti, modelli di sistemi organizzativi, etc.

---

isomorfismo tra LTS

Due sistemi di transizioni etichettati (S_1, E_1, T_1, s_{01}) e (S_2, E_2, T_2, s_{02}) sono **isomorfi** ((S_1, E_1, T_1, s_{01}) **iso** (S_2, E_2, T_2, s_{02})) sse esistono due mappe biunivoche $\alpha : S_1 \rightarrow S_2$ e $\beta : E_1 \rightarrow E_2$ tali che:

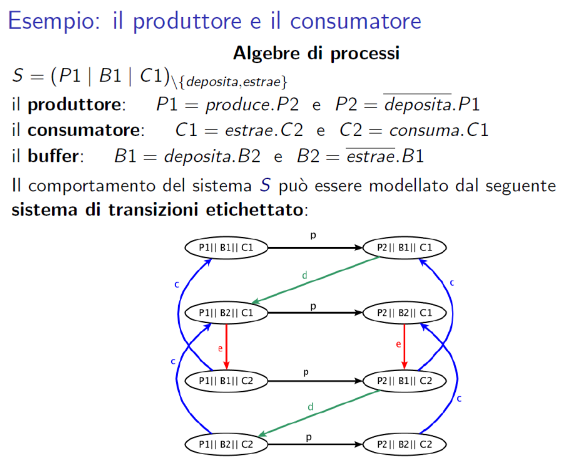
- $\alpha(s_{01}) = s_{02}$
- $\forall s, s' \in S_1, \forall a \in E_1$: $(s, a, s') \in T_1 \Leftrightarrow (\alpha(s), \beta(a), \alpha(s')) \in T_2$



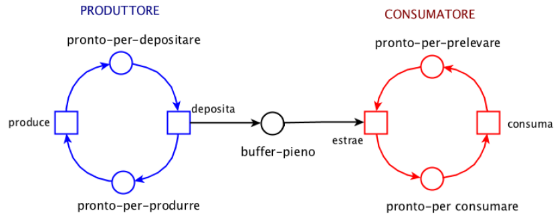
Aspetti critici nei sistemi di transizioni:

- nei sistemi distribuiti lo stato globale non e osservabile
- nella realtà fisica non esiste un sistema di riferimento temporale unico
- la simulazione sequenziale non deterministica (semantica a "interleaving") dei sistemi distribuiti è una forzatura, non ne rappresenta le caratteristiche reali del

comportamento

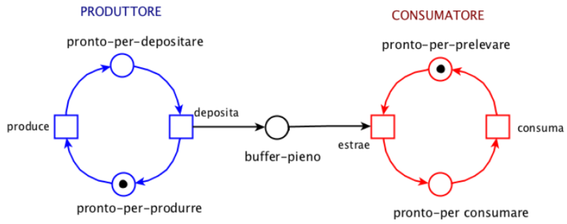


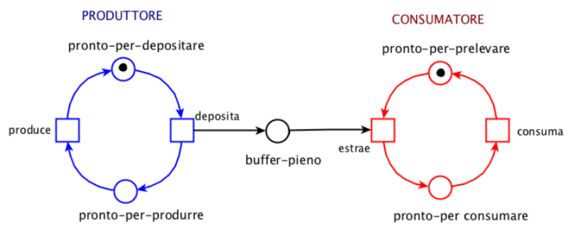
Proviamo a creare una soluzione con una rete di Petri:

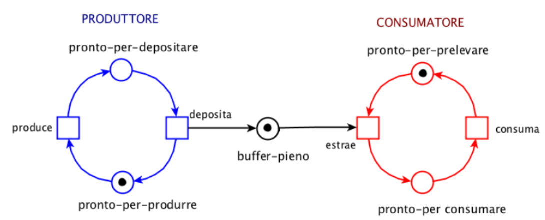


Il buffer non deve essere pieno e il produttore deve essere pronto per depositare. I cerchi rappresentano delle condizioni che possono essere vero oppure false. Deposita rappresenta l'evento che sarà possibile se le precondizioni sono vere e se le post condizioni sono false. Quando l'evento avviene, le precondizioni varranno impostate false e le post condizioni verranno impostate vere.
Precondizione: pronto-per-depositare; post condizione: pronto-per-produrre.
Stessa cosa per il consumatore che deve essere pronto per prelevare ma non pronto per consumare.

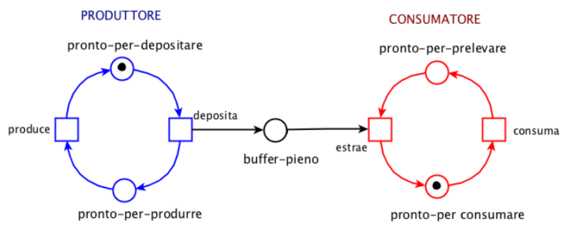
Segno con un pallino le condizioni vere e lascio vuote quelle false. In questo caso l'unico evento abilitato è l'evento produce, tutti gli altri non hanno le precondizioni necessarie.



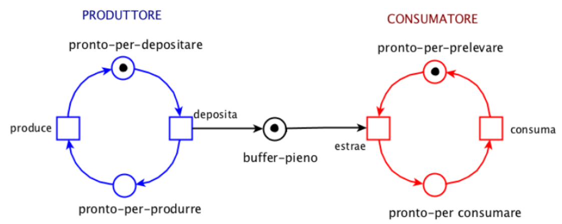




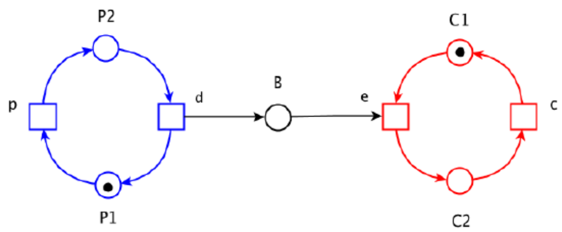
Ora sia “produce” che “estrae” sono abilitati. Questi possono occorrere in maniera concorrente.

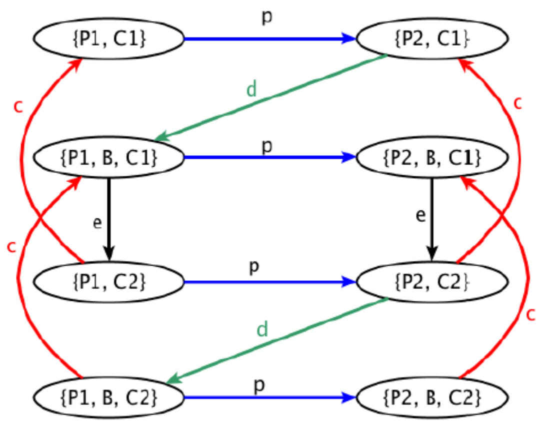


Anche qui il produttore può depositare in maniera indipendente da quando il consumatore consuma. I due eventi non interferiscono l’uno con l’altro e quindi possono accadere in maniera concorrente.



In questa configurazione deposita non è abilitato perché la post condizione non è falsa, mentre estrae è abilitato. Il deposito quindi può avvenire solo dopo l’estrazione del consumatore. Quindi c’è una relazione di dipendenza temporale tra i due eventi.





Questo è il comportamento come sistema di transizioni etichettato. Registro le condizioni vere nelle graffe.

Sistemi elementari: reti elementari

Formalizzazione di quanto appena visto:

$N = (B, E, F)$ é una **rete** se e solo se:

- B insieme finito di **condizioni** (**stati locali**, proposizioni/proprietá vere o false) rappresentate da ◯

- E insieme finito di **eventi** (trasformazioni locali di stato, **transizioni locali**) rappresentati da □

tali che : $B \cap E = \emptyset$ e $B \cup E \neq \emptyset$

- $F \subseteq (B \times E) \cup (E \times B)$ **relazione di flusso** rappresentata da $\longrightarrow$ tale che : $dom(F) \cup ran(F) = B \cup E$ (no elementi isolati)

Sia $x \in X,\ X = B \cup E$,

${}^\bullet x = \{y \in X : (y, x) \in F\}$ *pre-elementi* di x (precondizioni o pre-eventi)
$x^\bullet = \{y \in X : (x, y) \in F\}$ *post-elementi* di x (postcondizioni o post-eventi)

Sia $A \subseteq B \cup E$, ${}^\bullet A = \cup_{x \in A}\ {}^\bullet x$ e $A^\bullet = \cup_{x \in A}\ x^\bullet$

In questo modello non è previsto che possano esistere elementi isolati.

La rete $N = (B, E, F)$ descrive la *struttura statica del sistema*, il comportamento é definito attraverso le nozioni di *caso* e di *regola di scatto* (o di transizione).

Un **caso** (o *configuarazione*) é un insieme di condizioni $c \subseteq B$ che rappresentano l'insieme di condizioni vere in una certa configurazione del sistema, un *insieme di stati locali* che collettivamente individuano lo *"stato globale"* del sistema.

◯ condizione falsa ⊙ condizione vera

La regola di scatto

Sia $N = (B, E, F)$ una rete elementare e $c \subseteq B$.
L'evento $e \in E$ é **abilitato** (puó occorrere) in c, denotato $c[e >$, sse

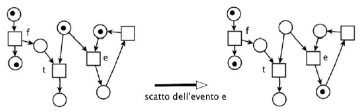
$${}^\bullet e \subseteq c \text{ and } e^\bullet \cap c = \emptyset$$

c è l'insieme delle condizioni vere. Il pallino prima di e significa pre condizioni, mentre la e con il pallino dopo sono le post condizioni.

Se $c[e >$, allora quando e **occorrere** in c genera un nuovo caso c', denotato $c[e > c'$: $\quad c' = (c - {}^\bullet e) \cup e^\bullet$

Quando l'evento occorre allora trasforma le pre condizioni in false e le post condizioni in vere.

principio di estensionalitá *(il cambiamento di stato é locale):*
"un evento é completamente caratterizzato dai cambiamenti che produce negli stati locali, tali cambiamenti sono indipendenti dalla particolare configurazione in cui l'evento occorre"



Tutto il resto può evolvere differentemente, non ci interessa se le altre condizioni esterne siano vere o false, rimarranno invariate.

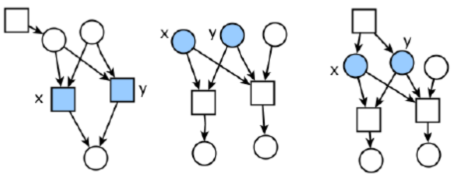
Reti semplici e pure

Sia $N = (B, E, F)$ una rete elementare

- N è **semplice** sse

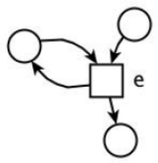
$$\forall x, y \in B \cup E,\ {}^\bullet x = {}^\bullet y \wedge x^\bullet = y^\bullet \Rightarrow x = y$$

N è **pura** sse $\forall e \in E : {}^\bullet e \cap e^\bullet = \emptyset$



reti non semplici

Se x e y sono eventi elementari, li posso descrivere con le loro pre e post. Quindi viene rappresentato un evento unico, x e y sono la stessa cosa. Penso che tutti debbano essere così perché la rete sia semplice.



rete non pura

Le pre condizioni e le post condizioni di un dato evento devono essere diverse. In questo caso l'evento non potrà mai occorrere.

Eventi indipendenti e concorrenti

Sia $N = (B, E, F)$ una rete elementare, $U \subseteq E$ e $c, c_1, c_2 \subseteq B$.

- U è un **insieme di eventi indipendenti** sse
$$\forall e_1, e_2 \in U: \; e_1 \neq e_2 \;\Rightarrow\; ({}^\bullet e_1 \cup e_1{}^\bullet) \cap ({}^\bullet e_2 \cup e_2{}^\bullet) = \emptyset$$

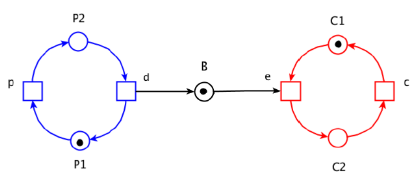
Se per ogni coppia di eventi in U, questi non hanno in comune nessuna precondizione e nessuna post condizione.

U è un **passo abilitato** (insieme di eventi concorrenti) in c $(c[U>)$ sse
$$U \text{ insieme di eventi indipendenti} \quad \wedge \quad \forall e \in U: c[e>$$

Con U insieme di eventi indipendenti, questi sono un passo abilitato se possono essere eseguiti contemporaneamente. Questo è diverso dalla sincronizzazione tra eventi indipendenti.

U è un **passo da** c_1 **a** c_2, $(c_1[U> c_2)$ sse
$$c_1[U> \quad \wedge \quad c_2 = (c_1 - {}^\bullet U) \cup U^\bullet$$

c1 e c2 vengono eseguiti contemporaneamente



- $\{p, e\}$, $\{p, c\}$, $\{d, c\}$ sono esempi di insiemi di eventi indipendenti
- $\{p, e\}$ è un passo abilitato in $\{P_1, B, C_1\}$
- $\{P_1, B, C_1\}$ $[\{p, e\} > \{P_2, C_2\}$

Un **sistema elementare** $\Sigma = (B, E, F; c_{in})$ è definito da una *rete* $N = (B, E, F)$ e da $c_{in} \subseteq B$ un *caso iniziale*.

L'insieme dei **casi raggiungibili** (C_Σ) del sistema elementare $\Sigma = (B, E, F; c_{in})$ è il più piccolo sottoinsieme di 2^B tale che:

- $c_{in} \in C_\Sigma$
- se $c \in C_\Sigma$, $U \subseteq E$, $c' \subseteq B$ sono tali che: $c[U > c'$, allora $c' \in C_\Sigma$

U_Σ è l'**insieme dei passi** di Σ: $U_\Sigma = \{U \subseteq E \mid \exists c, c' \in C_\Sigma : \; c[U > c'\}$

Definisco i casi raggiungibili in modo induttivo. Il caso iniziale appartiene all'insieme dei casi raggiungibili. Se poi ho un caso c che fa parte dei casi raggiungibili, se ho un evento che ha come post condizione c' allor anche c' è un caso raggiungibile.

c[U > c' significa che applico un insieme di eventi U ad un caso c raggiungendo il caso c'. Penso che un passo sia un insieme di eventi che ti portano da un caso c ad un caso c'.

Comportamento dei sistemi elementari

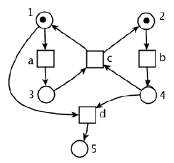
Sia $\Sigma = (B, E, F; c_{in})$ un sistema elementare, $c_i \in C_\Sigma$, $e_i \in E$, $U_i \subseteq E$

- comportamento sequenziale - sequenze di occorrenze / di eventi ("interleaving", simulazione sequenziale non deterministica)
  $c_{in}[e_1 > c_1[e_2 > \ldots [e_n > c_n$ oppure $c_{in}[e_1 e_2 \ldots e_n > c_n$
- comportamento non sequenziale - sequenze di passi ("step semantics")
  $c_{in}[U_1 > c_1[U_2 > \ldots [U_n > c_n$ oppure $c_{in}[U_1 U_2 \ldots U_n > c_n$
- comportamento non sequenziale - processi non sequenziali ("partial order semantics" - "true concurrency")
  ...

Si considerano sia sequenze finite che sequenze infinite (di eventi o di passi)

Esempio:

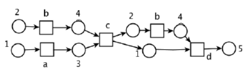
Dato il seguente sistema elementare Σ



- una possibile sequenza di occorrenze di eventi:
$\{1, 2\}[a > \{3, 2\}[b > \{3, 4\}[c > \{1, 2\}[b > \{1, 4\}[d > \{5\}$;

- una possibile sequenza di passi:
$\{1, 2\}[\{a, b\} > \{3, 4\}[\{c\} > \{1, 2\}[\{b\} > \{1, 4\}$.

- un processo non sequenziale di Σ:



Nel primo caso quella è una sequenza di eventi singoli.

Dato che a e b sono indipendenti possono anche essere eseguiti contemporaneamente come un passo, creando una sequenza di passi.

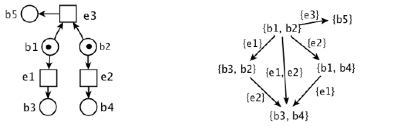
Nel terzo caso mostro la dipendenza e l'indipendenza tra le diverse condizioni.

Grafo dei casi raggiungibili

Il comportamento di un sistema elementare può essere rappresentato dal suo grafo dei casi.

Il **grafo dei casi** di Σ è il *sistema di transizioni etichettato* $CG_\Sigma = (C_\Sigma, U_\Sigma, A, c_{in})$ dove:

- C_Σ è l'insieme dei nodi del grafo (gli stati globali),
- U_Σ è l'alfabeto,
- A è l'insieme di archi etichettati:
  $A = \{(c, U, c') \mid c, c' \in C_\Sigma,\ U \in U_\Sigma,\ c[U > c'\},$

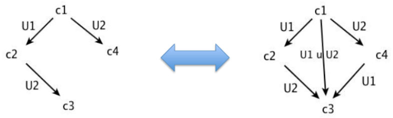


un sistema Σ e il suo grafo dei casi CG_Σ

Il grafo dei casi raggiungibili è un sistema di transizioni etichettato dove i nodi del grafo sono gli stati e gli archi sono eventi.

dyamond property

Sia $\Sigma = (B, E, F; c_{in})$ un sistema elementare, $CG_\Sigma = (C_\Sigma, U_\Sigma, A, c_{in})$ il suo grafi dei casi, $U_1, U_2 \in U_\Sigma$: $U_1 \cap U_2 = \emptyset$, $U_1 \neq \emptyset$, $U_2 \neq \emptyset$, e $c_1, c_2, c_3, c_4 \in C_\Sigma$, allora vale:



La vedremo bene la prossima volta