

Empio - magazzinoStato $X = n^{\circ}$ pezzi nel magazzinocapacità massima: K $X \in \{0, 1, 2, \dots, K\}$ STATO DISCRETOeventi $\begin{cases} a: \text{arrivo di un pezzo} \\ d: \text{partenza di un pezzo} \end{cases}$

$$X_{k+1} = f(X_k, e_{k+1}) = \begin{cases} X_k + 1 & \text{se } e_{k+1} = a \text{ e } X_k < K \\ X_k - 1 & \text{se } e_{k+1} = d \text{ e } X_k > 0 \\ X_k & \text{altrimenti} \end{cases}$$

↑

funzione di transizione dello stato

(k: contatore del numero di eventi $\Rightarrow$ dinamica "event-driven")

—————↑

 Novità rispetto a Fondamenti di Automatica:

stato discreto vs stato continuo

dinamica event-driven vs dinamica time-driven

SISTEMI A STATO CONTINUO/DISCRETO

(2)

↓
quando il suo stato assume
~~una~~ valori in un insieme
continuo

↓
quando lo stato assume
valori in un insieme
numerabile
(di cardinalità finita
oppure infinita).

Si parla di sistemi ibridi nel caso di sistemi
con stato misto continuo e discreto.

SISTEMI TIME-DRIVEN/EVENT-DRIVEN

↓
sono sistemi nei quali
è il ~~tempo~~ trascorrere del
tempo a determinare la
dinamica
⇒ tutte le variabili (ingressi,
stato, uscite) possono cambiare
a ogni istante di tempo

↓
lo stato subisce
cambiamenti soltanto
in corrispondenza
dell'occorrenza
(tipicamente asincrona)
di eventi.

Def - Si definisce SISTEMA AD EVENTI DISCRETI (SED)
un sistema dinamico con stato discreto e
dinamica di tipo event-driven.

Formalmente un SED e' definito da:

(3)

- insieme discreto $\mathcal{E}$ degli eventi
- spazio di stato discreto $\mathcal{X}$ (+ spazio delle uscite discreto $\mathcal{Y}$)
- dinamica event driven

$$\begin{cases} X_{k+1} = f(X_k, e_{k+1}) & \text{eq. ne di transizione dello stato} \\ y_k = g(x_k) & \text{eq. ne di uscita} \end{cases}$$

$x \in \mathcal{X}$: STATO

$e \in \mathcal{E}$: EVENTO

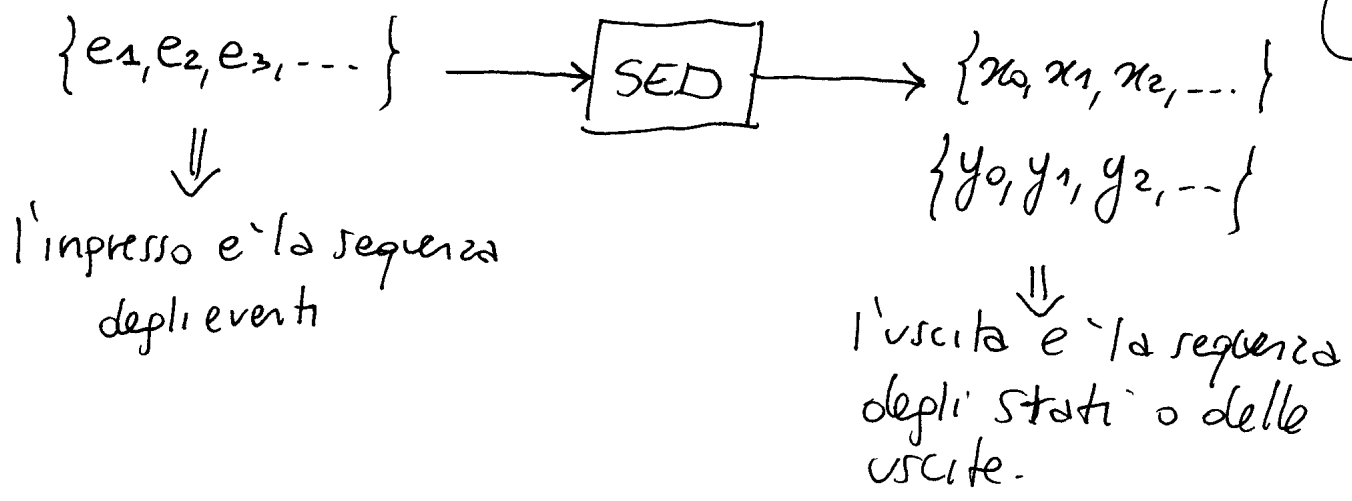
$y \in \mathcal{Y}$: USCITA

k : contatore del numero di eventi

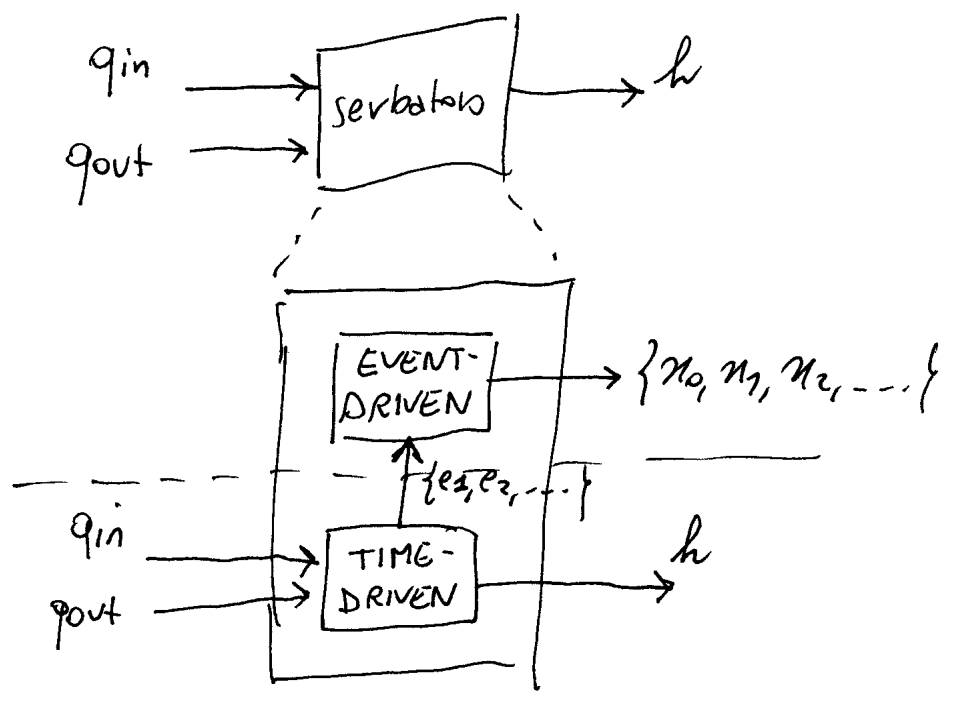
Nella dinamica event-driven non compare il tempo, essa descrive solo la logica di funzionamento del sistema, non la sua temporizzazione

↓
gli istanti di tempo in cui
accadono gli eventi

Per questo motivo, riferendoci a questi sistemi, parleremo di SED NON TEMPORIZZATI (o LOGICI)



esempio del serbatoio



$\{e_1, e_2, \dots\}$ → [] → $\{x_0, x_1, \dots\}$

per noi
 sarà il risultato
 della combinazione di
 processi stocastici concorrenti
 che sostituiscono la dinamica time-driven.

Esempi di eventi

- azioni specifiche (per esempio, pressione di un tasto su una tastiera, arrivo di un cliente in una coda, trasmissione di un pacchetto dati, completamento di una lavorazione, --)
- occorrenze spontanee (per esempio, guasti, interruzioni di servizio, --)
- soddisfacimento di una o più condizioni
(per esempio, il superamento di una soglia di allerta)



- In un SED nulla cambia fino al verificarsi del prossimo evento $\Rightarrow$ è per questo che si dice che la dinamica è "determinata dagli eventi".
- Ogni evento $e \in E$ ha associato un processo (che può essere deterministico o stocastico) attraverso il quale si determina l'istante in cui l'evento e si verifica.

La dinamica del SED è il risultato della combinazione di questi processi asincroni e concorrenti (non necessariamente indipendenti).

Esempio - macchina soggetta a guasti.

(6)

0: macchina libera
1: macchina occupata
2: macchina guasta.

stato X

Supponiamo di essere nello stato 1.

Nello stato 1 gli eventi di terminazione della lavorazione, e di guasto sono eventi CONCORRENTI, nel senso che il primo tra i due eventi ad accadere determina ~~l'ipotesi~~ la prossima transizione di stato.

⇒ Dal punto di vista modellistico, possiamo vedere i due eventi come il risultato finale di due processi (processo di lavorazione e processo di guasto)

Il processo che ha la durata inferiore determina il prossimo evento che dovrà accadere.

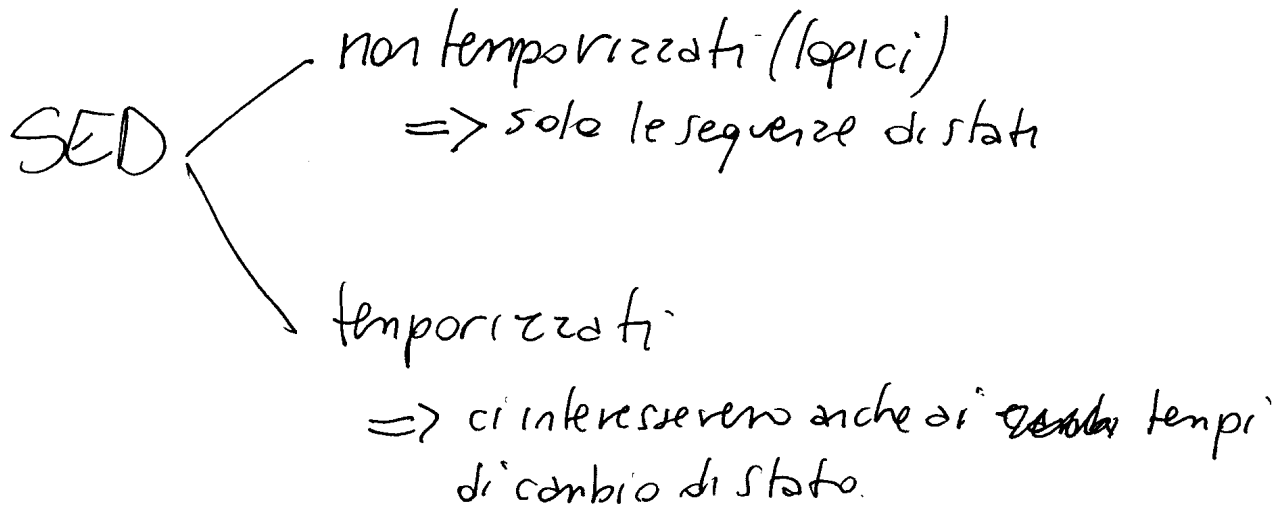
⇓

—+—

Nel seguito, astrarremo da questi processi e caratterizzeremo i tempi di occorrenza degli eventi in maniera deterministica oppure stocastica.

Quando fanno questo (cioè introducono i tempi di occorrenza degli eventi) parlano di SED temporizzati.

(7)

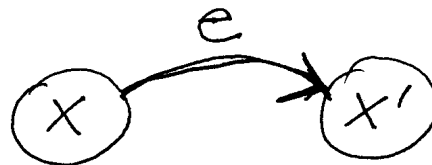


RAPPRESENTAZIONE GRAFICA DI SED

Grafo orientato e etichettato sugli archi (eventualmente anche sui nodi).

- NODI: stati
- ARCHI: funzione di transizione di stato
 $\Rightarrow$ metto un arco orientato dal nodo x al nodo x' con etichetta e se

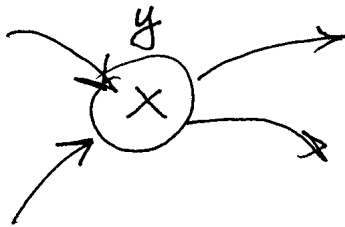
$$x' = f(x, e)$$



- ETICHETTE SUI NODI: funzione di uscita
 $\Rightarrow$ metto l'etichetta y sul nodo x se

$$y = g(x)$$

8

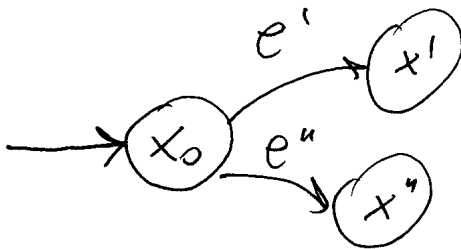


caso particolare: uscita binaria $y \in \{0, 1\}$

se $g(x) = 0$: $\textcircled{x}$

se $g(x) = 1$: $\textcircled{\textcircled{x}}$

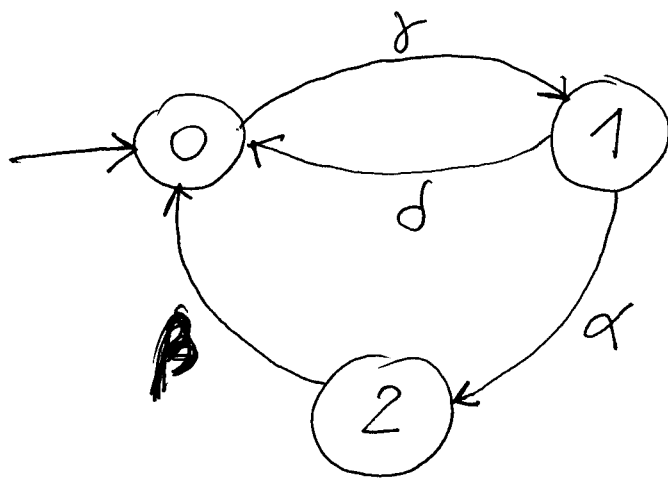
- La condizione iniziale è rappresentata da una freccia entrante senza etichetta e senza nodo di partenza.



ESEMPIO - macchina soppetta a guasti.

STATO $X = \begin{cases} 0 & \text{se la macchina è libera} \\ 1 & \text{se la macchina è occupata} \\ 2 & \text{se la macchina è guasta} \end{cases}$

Eventi $\left\{ \begin{array}{l} \alpha : \text{la macchina si guasta} \\ \beta : \text{la macchina viene riparata} \\ \gamma : \text{inizio lavorazione} \\ \delta : \text{fine lavorazione} \end{array} \right.$



ip- la macchina
si può guastare
solo quando è
in lavorazione

ip- il pezzo in
lavorazione ~~all~~
al momento del
guasto viene
scartato

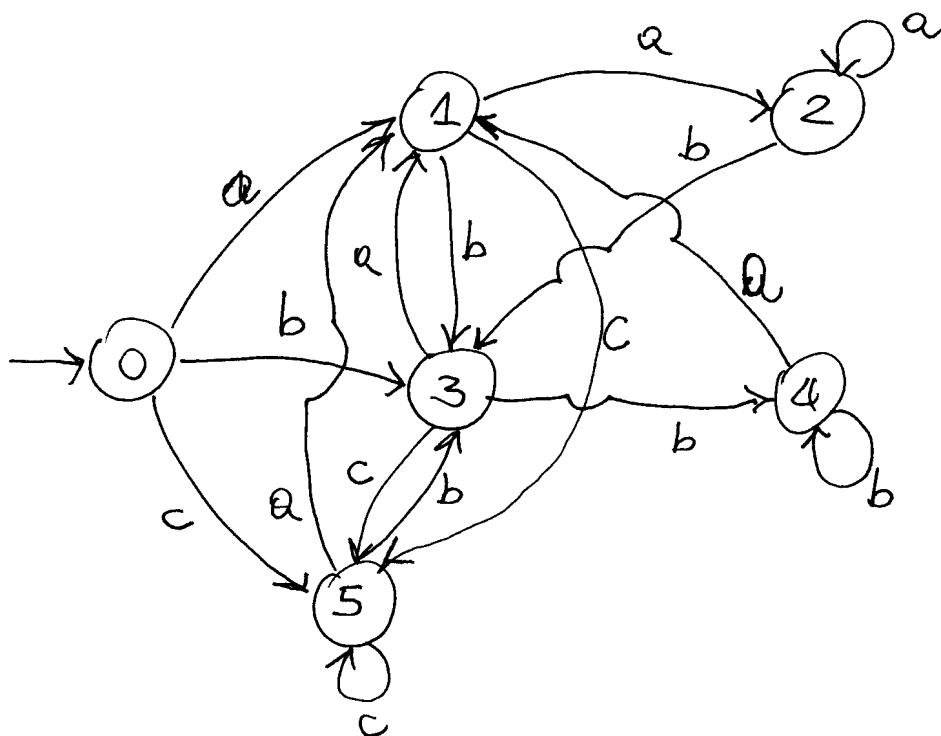
ip- all'inizio dell'
osservazione, la
macchina è libera.

ESEMPIO - Un macchinario esegue operazioni di
tre tipi, indicate con a , b , e c . Per motivi
tecnologici, un'operazione di tipo c non può essere
eseguita immediatamente dopo due operazioni di tipo a ,
oppure due operazioni di tipo b , consecutive.

Descrivere la dinamica del sistema attraverso un
modello ad eventi discreti.

EVENTI $\left\{ \begin{array}{l} a \\ b \\ c \end{array} \right.$

- 0 : nessuna operazione effettuata
- 1 : l'ultima operazione effettuata è di tipo a,
e la penultima non è di tipo a
- 2 : le ultime due operazioni effettuate sono di tipo a
- 3 : l'ultima op.ne effettuata è di tipo b,
e la penultima non è di tipo b.
- 4 : le ultime due op.ni effettuate sono di tipo b.
- 5 : l'ultima op.ne effettuata è di tipo c.



- fare: # stati non minimo
accorpare 2 e 4.

- fare modello da parte di chi deve progettare le operazioni?