1. Introduction

1.1 Discrete Event System

signal: map $y: T \rightarrow Y$

Time → maybe continuous or discrete

value set

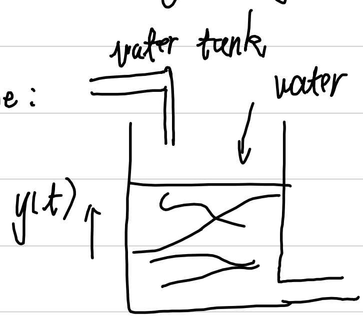
$y(t) \in Y$

$\forall t \in T$

in DES: consider discrete-valued signals

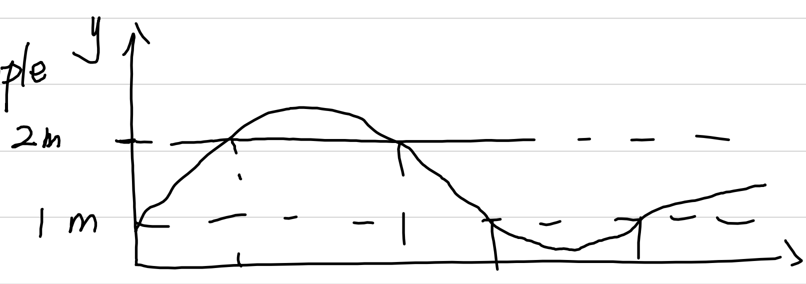
· often generated by quantization process　　欧州 gs200

Example:

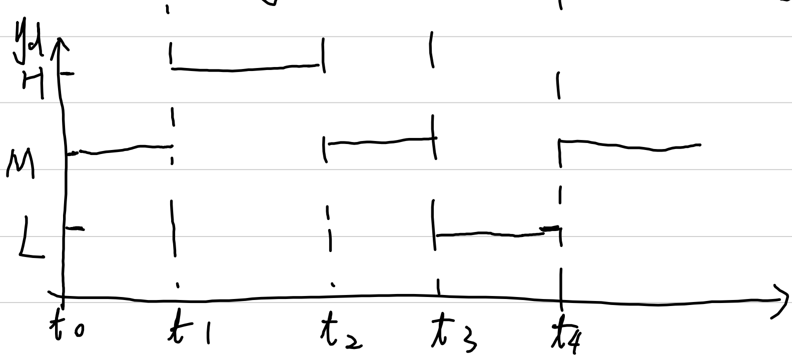


$y: \mathbb{R}^+ \rightarrow \mathbb{R}^+$

Example



quantized signal: $y_d: \mathbb{R}^+ \rightarrow \{Hi, Me, Lo\}$



$\rightsquigarrow$ sequence of logical events

$\mathbb{N}_0 \longrightarrow \Sigma$

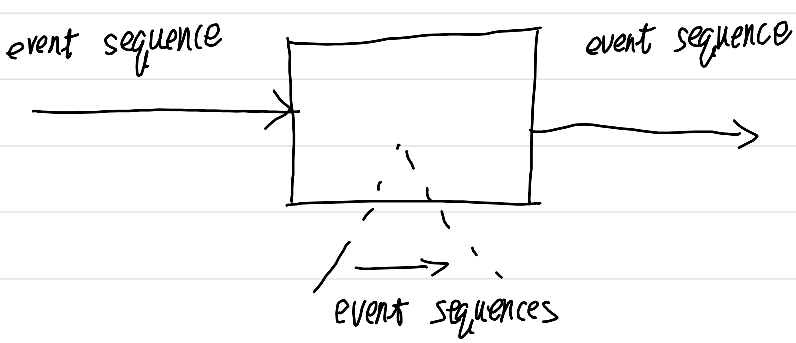
continuous-valued signal — quantilize

discrete-valued signal = sequence of timed event — protract our time

sequence of logic event

signal aggregation: increasing level of abstraction (loss of information)

Discrete Event Sytem (DES): All signals are discrete-valued (sequence of timed / logical discrete event): input, output, internal signals.



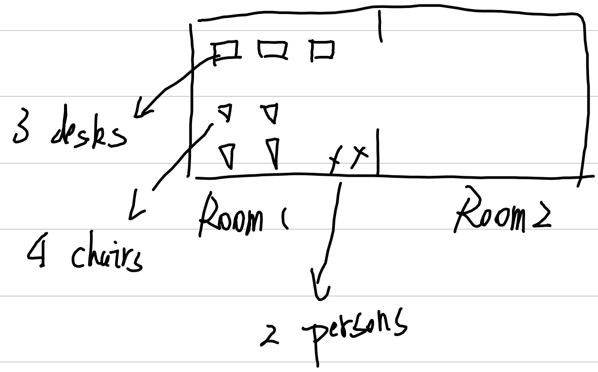
\# timed DES

\# untimed / logical DES

- Petri net method
- method based on formal (language / finite automate)

2. Petri Net

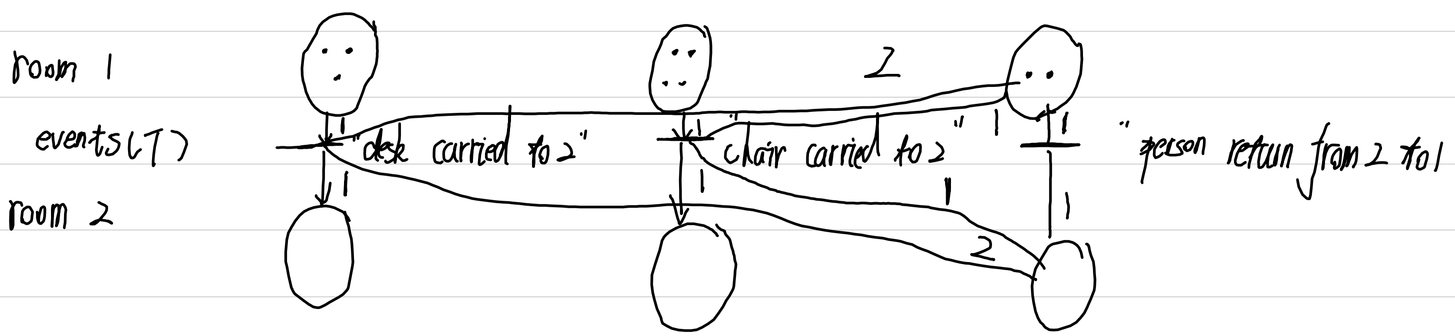
Introductory example:



2 person carry 1 desk

1 person carry 1 chair

(P) number of desks   number of chairs   number of persons



2.1 Petri Net Graph

Def: A Petri net graph is a direct bipartite graph

→ node set partioned into 2 distinct subsets.

$$N = (P, \underset{\text{}}{T}, \overset{\text{arc}}{E}, \overset{\text{weight}}{W})$$

When $P = \{p_1, \dots p_n\}$ finite set of places

$T = \{t_1, \dots t_n\}$ finite set of transitions.

$E \subseteq (P \times T) \cup (T \times P)$ set of direct arcs / edges.

$W: E \to \mathbb{N} \dots$ weight function

$\{1, 2, 3 \dots\}$

$I(t_j) := \{p_i \in P \mid (p_i, t_j) \in E\}$

$O(t_j) := \{p_j \in P \mid (t_j, p_i) \in E\}$

$I(p_i) := \{t_j \in T \mid (t_j, p_i) \in E\}$

$O(p_i) := \{t_j \in T \mid (p_i, t_j) \in E\}$

graphical representation:

$\bigcirc$ ... places

$|$ ... transitions

$\xrightarrow{\alpha}$ ... arc with weight α

Remark: often alternative definition of weight function

$W: (P \times T) \cup (T \times P) \longrightarrow \mathbb{N}_0 = \{0, 1, 2 \dots\}$ corresponding to W

Then: E determined by W.

$E = \{(p_i, t_j) \mid W(p_i, t_j) \geq 1\} \cup \{(t_j, p_i) \mid W(t_j, p_i) \geq 1\}$

Interpretation: transition corresponds to event

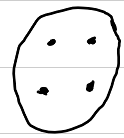
places are used to formulate conditions for event to happen

2.2 Petri Net Dynamics

Definition: A Petri Net is a pair (N, x^0), where

1. $N = (P, T, E, W)$ graph
2. $x^0 = \mathbb{N}_0^n$, $n = |P|$ vector of initial markings.

cardinality of set P 初始状态

graphical representation: e.g. $x_i^0 = 4$  We have 4 tokens initial P_i

Interpreted: a dynamic system with state signal $x: \mathbb{N}_0 \to \mathbb{N}_0^n$
and initial states $x(0) = x^0$

(at least 1 desk, 2 persons in room 1 for event 1 taking place)

dynamic determined by two rules: 1. whether transition can actually occur

$\iff x_i(k) \geq$ the $W(p_i, t_i)$ $\forall p_i \in I(t_j)$

2. occurance ("firing") of transition t_j changes the number of tokens in place p_i according to:

$$x_i = \begin{cases} x_i(k) - W(p_i, t_j) + W(t_j, p_i), & \text{if } p_i \in I(t_j) \cap O(t_j) \\ x_i(k) - W(p_i, t_j), & \text{if } p_i \in I(t_j) \setminus \overset{\text{not}}{O(t_j)} \\ x_i(k) + W(t_j, p_i), & \text{if } p_i \in O(t_j) \setminus I(t_j) \\ x_i(k) & \text{else} \end{cases}$$

**

Formally: transition function:

$$f: \mathbb{N}_0^n \times T \longrightarrow \mathbb{N}_0^n$$

old state   Transition   new state

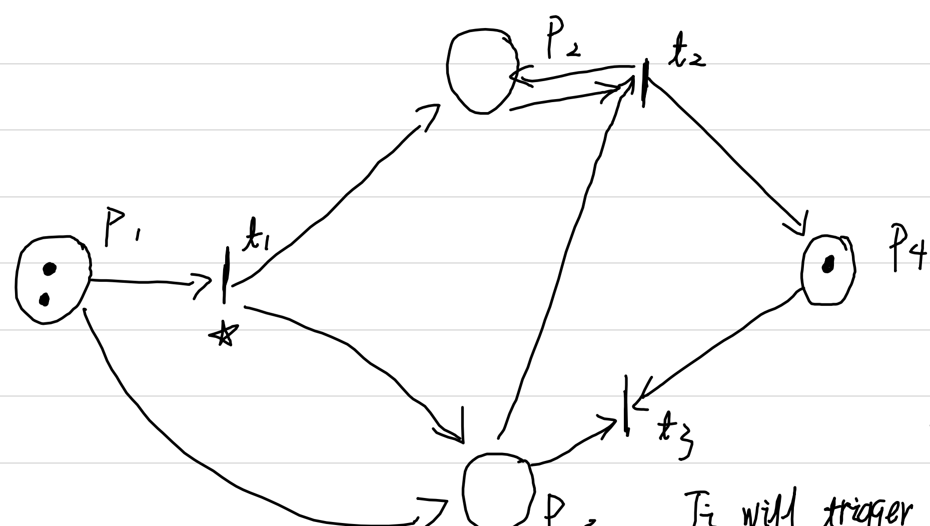
f is a partial function

$f(x(k), t_j)$ is defined iff rule1 holds

If $f(x(k), t_j)$ is defined, then $f_i(x(k), t_j) = x_i(k)$ is given by rule 2

$(i = 1, 2, \dots)$

Example:



(N, x^0)

Ti will trigger all related pi

$$x(0) = x^0 = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} \longrightarrow$$

t_1 enabled, $p_1 \geq w(p_1, t_1)$

t_2 not enabled (p_2, t_2) (p_3, t_2)

t_3 not enabled (p_1, t_3) (p_3, t_3) (p_4, t_3)

$\Uparrow$

p_1 and p_4 is sufficient but p_3 not

$$x' = f(x_0, t_1) = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \longrightarrow$$

t_1 enabled

t_2 enabled

t_2 eablod

t_4 enabled

$$x(0) = x^0 = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} \xrightarrow{t_1} x(1) = x^1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{tikzcd}
 & x^2 = \begin{bmatrix} 0 \\ 2 \\ 2 \\ 1 \end{bmatrix} \arrow[r, "t_2"] & x^3 \begin{bmatrix} 0 \\ 2 \\ 1 \\ 2 \end{bmatrix} \\
x^1 \arrow[ur, "t_1"] \arrow[r, "t_2"] \arrow[dr, "t_3"'] & x^2 = \begin{bmatrix} 0 \\ 3 \\ 1 \\ 2 \end{bmatrix} \arrow[ur, "t_1"'] & \vdots \\
 & x^2 = \begin{bmatrix} 1 \\ 3 \\ 2 \\ 3 \end{bmatrix} &
\end{tikzcd}$$

in the following steps, only t_2 can happen. dead lock

Lecture 2 N ... Petri net graph

$PN: (\underbrace{P, T, E, W}_{N}, x^0)$ $\leftarrow x^0 \in \mathbb{N}_0^n$ $n = |P|$

P: place, T: transition, $E \subseteq (P \times T) \cup (T \times P)$

$W: E \to \mathbb{N}$

Interpret Petri Net as a Dynamical System

state space $\mathbb{N}_0^n$, $n = |P|$
initial state $x(0) = x^0$
state evolution governed by 2 rules

(1) t_j is enabled in state $x(k)$ → the state after firing of a transition

$\iff x_i(k) \geq w(p_i, t_j)$, $\forall p_i \in I(t_j)$

(2) If t_j fires

$$x_i(k) \begin{cases} x_i(k) + W(t_j, p_i) - W(p_i, t_j), & \text{if } p_i \in I(t_j) \cap O(t_j) \\ x_i(k) + W(t_j, p_i), & \text{if } x_i(k) \in O(t_j) \setminus I(t_j) \\ x_i(k) - W(p_i, t_j), & \text{if } x_i(k) \in I(t_j) \setminus O(t_j) \\ x_i(k) \end{cases}$$

partial transition function

$f: \mathbb{N}_0^n \times T \longrightarrow \mathbb{N}^n$

$A^- \in \mathbb{N}_0^{n \times m}$, n is the number of places, m is the number of transitions

with element $a^-_{ij} = \begin{cases} w(p_i, t_j), & \text{if } (p_i, t_j) \in E \\ 0, & \text{else} \end{cases}$ number of token that p_i lose.

$A^+ \in \mathbb{N}_0^{n \times m}$ with element $a^+_{ij} = \begin{cases} w(t_j, p_i), & \text{if } (t_j, p_i) \in E \\ 0, & \text{else} \end{cases}$

... number of token that p_i gains, if t_j fired

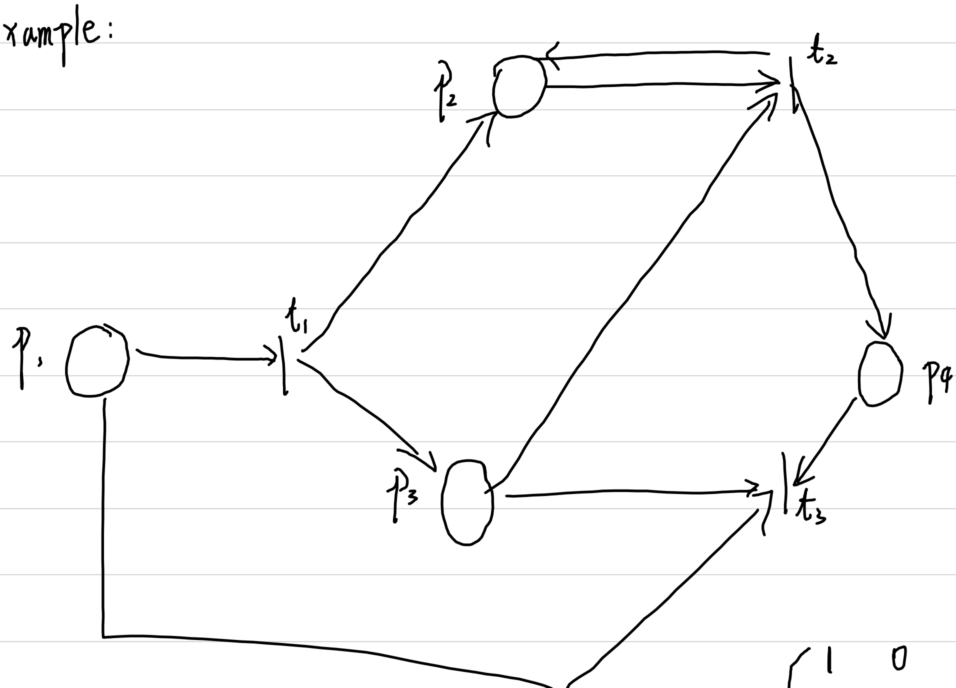
$A := A^{+} - A^{-} \in \mathbb{Z}^{n \times m}$ ... incidence matrix of the Petri Net with element $a_{ij} = a_{ij}^{+} - a_{ij}^{-}$

- t_j can fire a certain state $x(n) \iff x(n) \geq \mathrm{col}_j(A^{-}) = A^{-} u_j$

  $x(n)$: element vector; u_j: j-th unit vector

- If t_j is enabled in $x(k)$, then the new state $x(k_n) = f(x(k), t_j) = x(k) + \mathrm{col}_j(A) = x(k) + A u_j$

Example:



$n = 4$, $m = 3$, $x^0 = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix}$. $A^{-} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$ $\quad x^0 \geq A^{-}$

$$A^{+} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$A = A^+ - A^- = \begin{bmatrix} -1 & 0 & -1 \\ 1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & -1 \end{bmatrix}$$

$x^0 \geq col_1(A^-) \rightarrow t_1$ enabled.

$x^0 \ngeq col_2(A^-) \rightarrow t_2$ not enabled

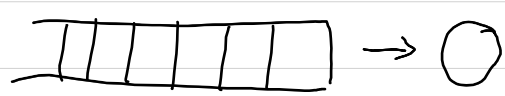
$x^0 \ngeq col_3(A^-) \rightarrow t_3$ not enabled

$$x(1) = f(x(0), t_1) = x(0) + col_1(A^-) = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$N = (P, T, E, w) \Longleftrightarrow A^+, A^- \Longleftrightarrow A^-, A$

Example: $\boxed{\text{Queing system}}$

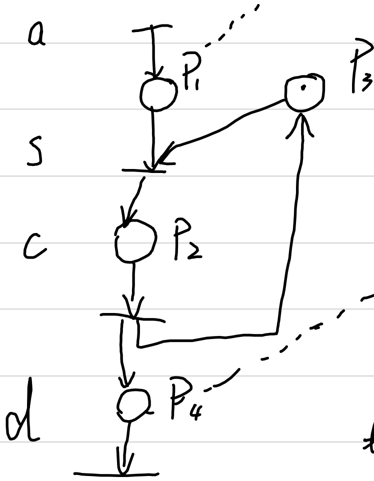
"customer arrives" (event a) →



"service starts" (event S)

"service completed" (event c)

Petri net



number of waiting customers

token in P_3 means "service idle"

token in P_2 means "service busy"

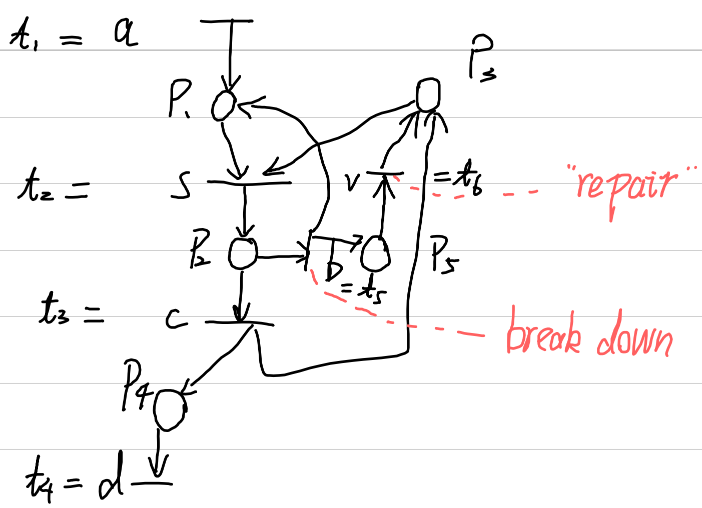
customer leave after the service have been completed.

a new transition D

token in P_4 means customers having been served loitered to post office.

"customer depeot / leave"

extension "server break down"



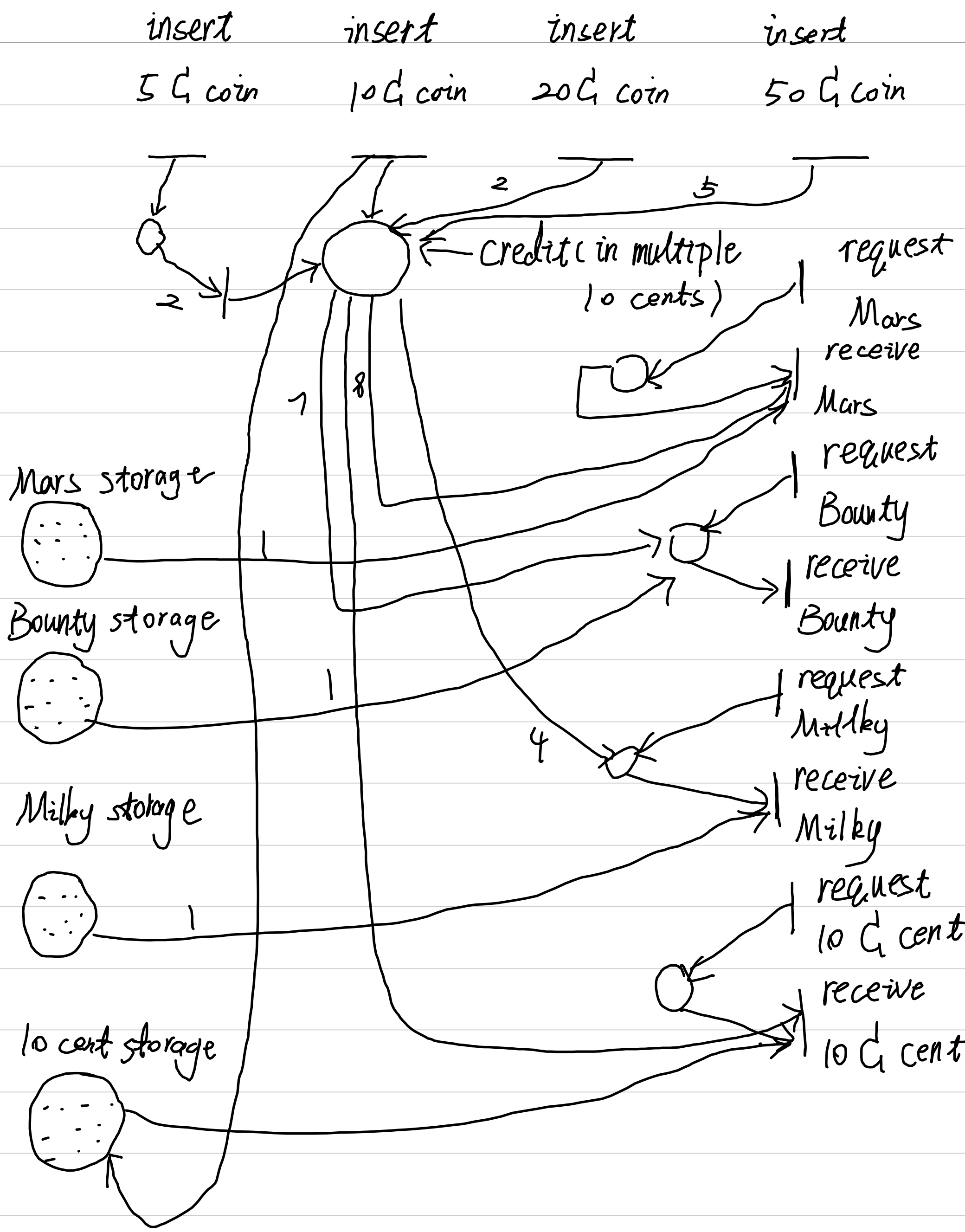
$$A^- = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A^+ = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$A = A^+ - A^- = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 & 0 \\ 0 & 1 & -1 & 0 & -1 & 0 \\ 0 & -1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

Example: candy machine that sells different kinds of sweets.

- accept the following: 5 ¢, 10 ¢, 20 ¢, 50 ¢
- it returns change
- sells
  - Mars (80 ¢)
  - Bounty (70 ¢)
  - Milky Way (40 ¢)



insert 5¢ coin
insert 10¢ coin
insert 20¢ coin
insert 50¢ coin
2
5
2
credit (in multiple 10 cents)
request Mars
receive Mars
7
8
request Bounty
receive Bounty
Mars storage
1
Bounty storage
1
request Milky
4
receive Milky
Milky storage
1
request 10¢ cent
receive 10¢ cent
10 cent storage

2.3 Special Classes of Petri Nets

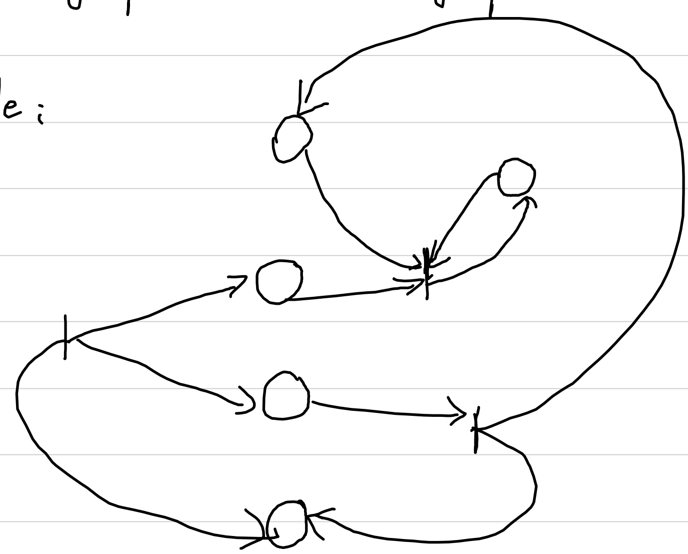
- synchronization graph (event graph) → several inputs at the same time

  # $|I(p_i)| = |O(p_i)| = 1 \quad \forall p_i \in P$ — the point P_i only have one input and one output

  # $W(p_i, t_j) = 1 \quad \forall (p_i, t_j) \in E$

  $W(t_j, p_i) = 1 \quad \forall (t_j, p_i) \in E$ — a cycle

Example:



cannot model conflicts / decision

("decision free" Petri Net)

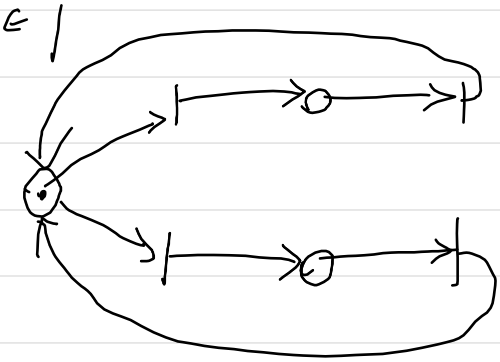
model synchronization may have several input places (enable several t fire at the same time)

- state machine

  # $|I(t_j)| = |O(t_j)| = 1 \quad \forall t_j \in T$

  # $W(p_i, t_j) = 1 \quad \forall (p_i, t_j) \in E$

  $W(t_j, p_i) = 1 \quad \forall (t_j, p_i) \in E$



cannot model synchronization

model conflict

Lecture 3

Recap: $PN = (\underbrace{\overbrace{P, T, E, w}^{\text{graph}}}, x^0)$

P ← places, T ← transition, $\underbrace{\text{places} \quad \text{transition}}_{\text{Nodes}}$

$E \subseteq (P \times T) \cup (T \times P)$

$w: E \to \mathbb{N}$

The condition of dynamical system is just the number of token in all places

state $x(k) \in \mathbb{N}_0^n$ ← $|P|$

↳ how often transitions have fired so far.

initial state $x(0) = x^0$

t_j enabled in $x(k)$

$\Leftrightarrow x_i(k) \geq w(p_i, t_j) \quad \forall p_i \in I(t_j)$

$\Leftrightarrow x(k) \geq col_j(A^-)$ *

if t_j fires

$x(k+1) = x(k) + col_j\underbrace{(A^+ - A^-)}_{A \text{ incidence matrix}}$ ** (what will happen)

$f: \mathbb{N}_0^n \times T \to \mathbb{N}_0^n$

$\mathbb{N}_0^n$ ↓ state, T ↓ transition

2.4 Analysis of Petri Nets.

2.4.1 Petri Net Properties

$T = \{t_1, \dots t_m\}$ transition net

T^* .... set of all finite strings from T ("Kleene closure")

$$T^* = \{\underset{\substack{\uparrow \\ \text{string of} \\ \text{length } 0}}{\varepsilon}, \underbrace{t_1, \dots, t_m}_{\substack{\text{string of} \\ \text{length } 1}}, \underbrace{t_1 t_2, t_1 t_2 \dots t_m t_n}_{\substack{\text{string of} \\ \text{length } 2}}, t_1 t_2 t_3 \dots t_m t_n t_k, \dots\}$$

$$f : \mathbb{N}_0^n \times T^* \rightarrow \mathbb{N}_0^n$$

$$f(x^0, \varepsilon) = x^0$$

$$f(x^0, t_{i_1} \dots t_{i_k}) = f(\dots(f(f(x_0, t_{i_1}), t_{i_2}), t_{i_3}), \dots t_{i_n})$$

a) Reachability

Def: A state $x^1 \in \mathbb{N}_0^n$ of Petri net (N, x^0) is reachable, if there exists a string $s \in T^*$ such that $x^1 = f(x^0, s)$

$R(N, x^0)$: the set of reachable state of (N, x^0)

b) Boundedness.

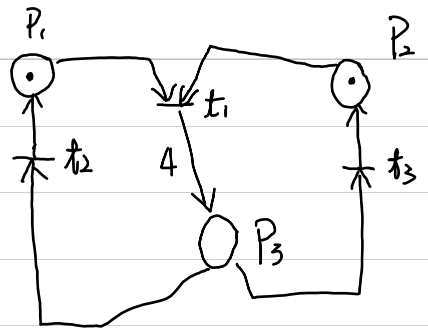
Def: $p_i \in P$ is bounded if $\exists k \in \mathbb{N}_0$ such that $x_i^1 \leq k \quad \forall x^1 \in R(N, x^0)$

↳ a property refer to one or several places

(N, x^0) is bounded if all the places are bounded

obvious: (N, x^0) bounded $\Longleftrightarrow R(N, x^0)$ is finite.

Example:



... unbounded

c) Coverability

recall: t_j enabled in $\underset{\uparrow}{x^1}$ $\Longleftrightarrow$ $x^1 \geq \underbrace{Col_j(A^-)}_{:=\xi^j}$

$f(x_0, s)$

" x^1 covers ξ^j "

Def: $\xi \in \mathbb{N}_0^n$ is coverable by (N, x^0) if there exists $x^1 \in R(N, x^0)$ s.t. $x^1 \geq \xi_i$, $i = 1, \ldots, n$

for every transition, x^1 in future that x^1 is available for firing t_j $\longrightarrow$ every transition can occur sometime in the future

ξ is not coverable means that there exists transition can not be fired at any possible condition x^1 in the future and at the present.