Lecture 30
Chapter # 8 (contd.)
PETRI NET-BASED PERFORMANCE MODELING

Example Problem 1

Consider the Petri-Network in Fig 25.

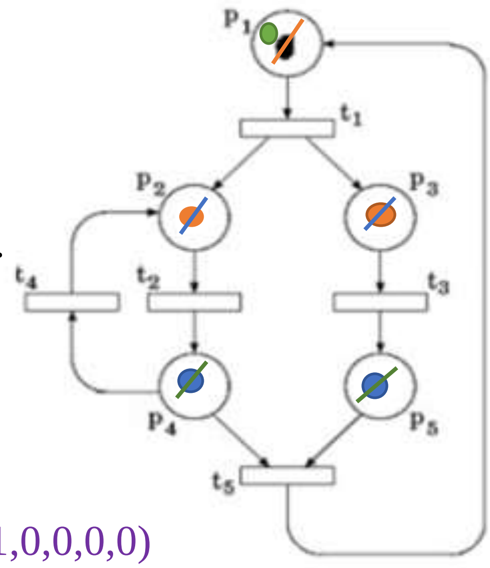


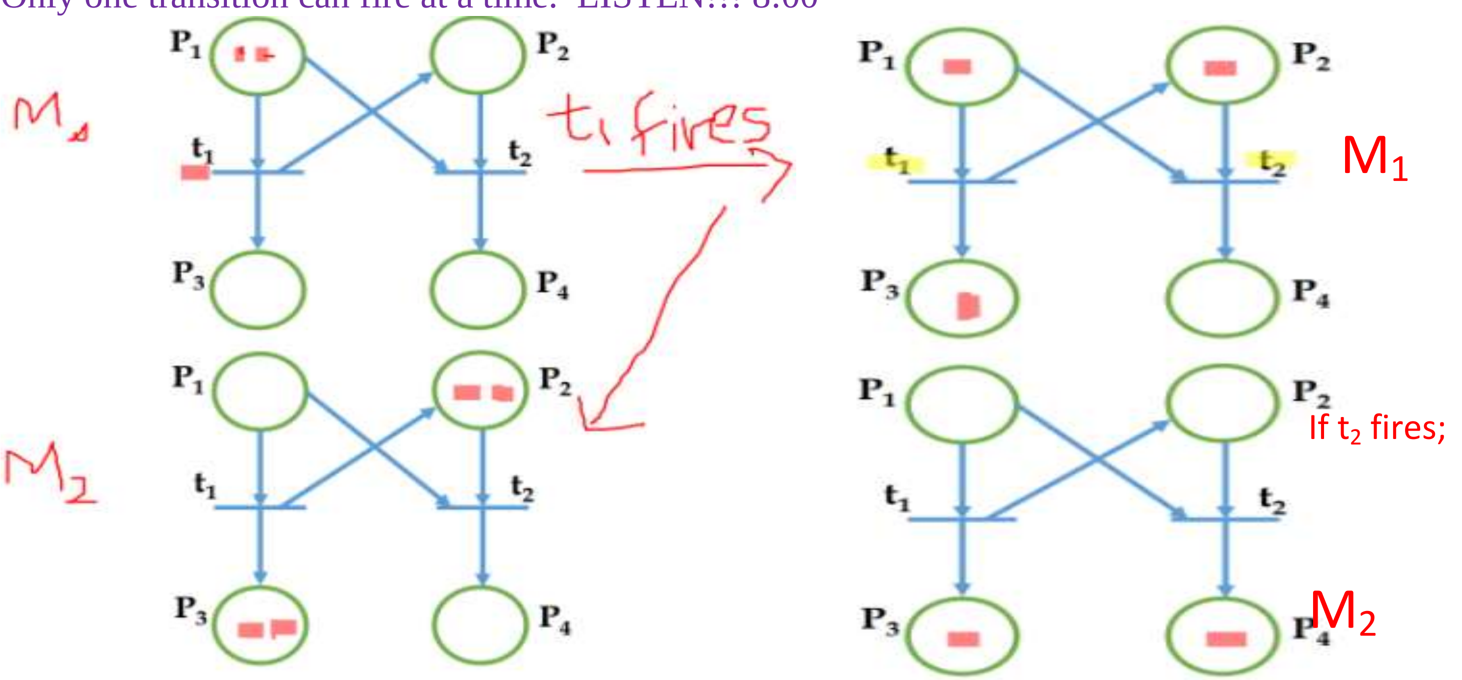
Fig 25

a) Provide the sets of places, transitions & initial marking.
b) Provide the sets of all the input and output functions w.r.t the transitions given in the figure.
c) Suppose at time T_1, transition t_1 is able to fire. What would be the marking M_1?
d) Identify M_2, M_3, M_4 and M_5 at times T_2, T_3 and T_4.

a) P = {P1, P2, P3, P4, P5}, T = {t1, t2, t3, t4, t5}, $\mu 0 = (1,0,0,0,0)$
b) $I(t_1) = (P1)$, $I(t_2) = (P2)$, $I(t_3) = (P3)$, $I(t_4) = (P4)$, $I(t_5) = (P4, P5)$
$O(t_1) = (P2, P3)$, $O(t_2) = (P4)$, $O(t_3) = (P5)$, $O(t_4) = (P2)$, $O(t_5) = (P1)$.
c) M1 = (0,1,1,0,0), M2 = (0,0,0,1,1), M3 = (1,0,0,0,0)
M4 = (0,1,1,0,0), M5 = (0,0,0,1,1)

Example Problem 2

A transition is enabled when each of it input places has at least as many tokens as the multiplicity of the corresponding input arcs. An enabled transition may fire & in a result of firing each output place of transition fired receives as many tokens as the multiplicity of the corresponding output arcs.
On the other hand, each input place of the transition fired releases as many tokens as the multiplicity of the corresponding input arcs.
Only one transition can fire at a time. LISTEN!!! 8:00



MATRIX ANALYSIS

• The input and output functions of a PN can be equivalently defined using a matrix notation.

• Let D^- denotes the *input matrix*. D^- is a (n_t x n_p) matrix, whose generic element d_{ij}^- is equal to the number of arcs connecting place p_j with transition t_i.
• Similarly, the *output matrix* D^+ is a (n_t x n_p) matrix, whose generic element d_{ij}^+ is equal to the number of arcs connecting transition t_i with place p_j.
• The *incidence matrix* D is defined by the following relation:

$$\mathbf{D = D^+ - D^-}$$

Input matrix is also called the pre-incidence matrix & output matrix is also called the post-incidence matrix.

Example

Following are the matrices D^-, D^+ and D for the PN of Fig 26:

$$D^- = \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{array} \begin{array}{c} \begin{array}{ccccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 \end{array} \\ \left| \begin{array}{ccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array} \right| \end{array}$$

$$D^+ = \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{array} \begin{array}{c} \begin{array}{ccccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 \end{array} \\ \left| \begin{array}{ccccccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{array} \right| \end{array}$$

$$D = \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{array} \begin{array}{c} \begin{array}{ccccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 \end{array} \\ \left| \begin{array}{ccccccc} -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & -1 \end{array} \right| \end{array}$$

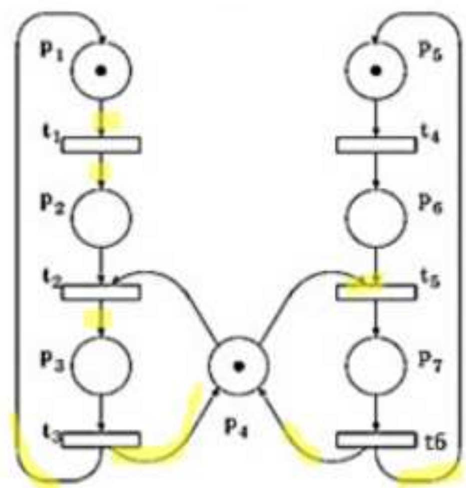


Fig 26

• Introducing the vector $\underline{e}_j$ which is a n_t-dimensional row vector with all the entries equal to 0 except entry j equal to 1.
•With this notation the execution rules of a PN becomes:

- a transition t_j is enabled in marking M iff $\mathbf{M \geq \underline{e}_j D^-}$ (note that $e_j D^-$ is the j-th row of D^-);
- firing of t_j in M produces a marking M' given by:

$$\mathbf{M' = M - e_j D^- + e_j D^+ = M + e_j D}$$

• Given a PN with initial marking $\mathbf{M_1}$ and a firing sequence $t_i \rightarrow t_j \rightarrow t_k \rightarrow t_j \rightarrow t_i$, the marking obtained at the end of the sequence is given by the following matrix equation:

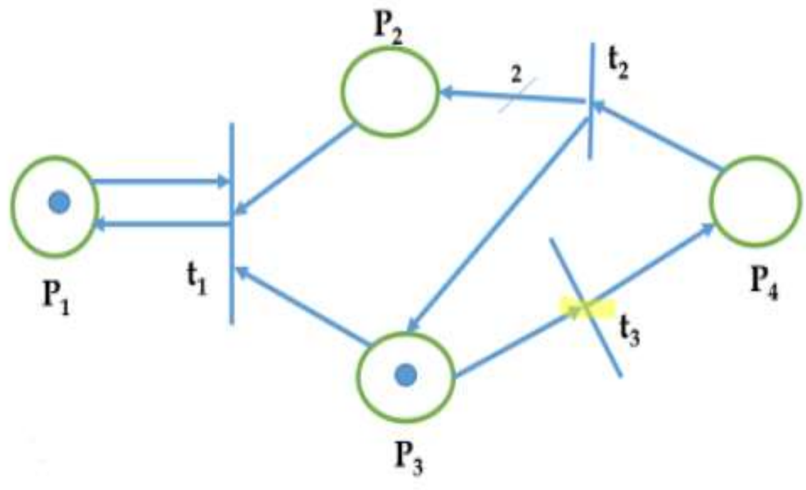
$$\mathbf{M_{fin} = M_1 + (\underline{e}_i + \underline{e}_j + \underline{e}_k + \underline{e}_j + \underline{e}_i)\, D}$$

Example Problem 3

Q) Consider the figure, find M' if t_3 is enabled to fire.

Input or Pre-Incidence Matrix:

$$D^- = \begin{array}{c} t_1 \\ \\ \\ \end{array} \begin{vmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{vmatrix}$$



Fig

Output or Post-Incidence Matrix:

$$D^+ = \begin{vmatrix} 1 & 0 & 0 & 0 \\ \underline{0} & 2 & 1 & \underline{0} \\ 0 & 0 & 0 & 1 \end{vmatrix} \begin{array}{c} \\ t_2 \\ \\ \end{array}$$

Incidence Matrix:

$$\underline{D} = \begin{vmatrix} 0 & -1 & -1 & 0 \\ 0 & 2 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{vmatrix}$$

Initial Marking:

$M_0 = [1\ 0\ 1\ 0]$

From the Fig, t_3 is enabled:

$\mathbf{e_j = (0, 0, 1)}$ as (t_1, t_2, t_3) & only t3 is enabled

$$M' = M + e_j D$$

$$M' = [1\ 0\ 1\ 0] + [0\ 0\ 1] \begin{bmatrix} 0 & -1 & -1 & 0 \\ 0 & 2 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

$$M' = [1\ 0\ 0\ 1]$$

Q) Let the firing sequence be: $t_3t_2t_3t_2t_1$

$\Longrightarrow e_j = [1\ 2\ 2]$

If $M = [1\ 0\ 1\ 0]$,

$$M' = [1\ 0\ 1\ 0] + [1\ 2\ 2] \begin{bmatrix} 0 & -1 & -1 & 0 \\ 0 & 2 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

M’ = [1 3 0 0]

Timed Petri Nets

• An activity in a real system takes finite time to perform its operation.
• e.g., to read a file from a disk, to execute a program, or to communicate with some other machine.
• Adding time provides Petri net modeler with another *powerful tool* to study the performance of computer systems.
• Time can be associated with transitions, selection of paths, waiting in places, inhibitors, and with any other component of the Petri net.
• The most typical way that time used in Petri nets is with transitions.

- the firing of a transition can be viewed as the execution of an event being modeled – e.g., a CPU execution cycle.

• These timed transitions are represented graphically as a *rectangle* or *thick bars*.

The Semantics of the Firing

• When a transition becomes enabled, its clock timer is set and begins to count down.
• Once the timer reaches 0, the transition fires.
• In Fig 27, when token arrives at p_1, the timer for t_1 set to τ_1 and begins to count down.
When the transition is enabled, the timer will start counting down.
• The decrement of the timer must be at a constant fixed transitions in the Petri net model.

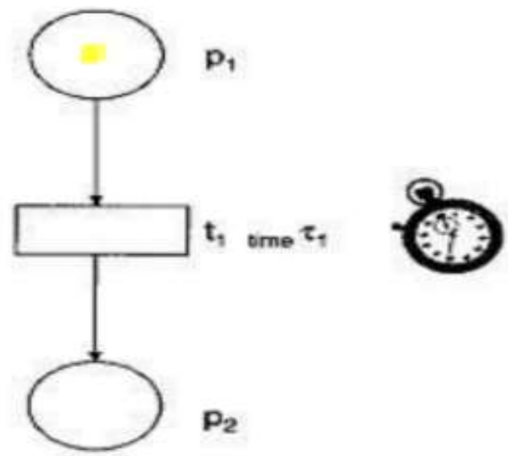


Fig 27: Timed Petri net

• A consideration to think about

- what occurs when a transition becomes non-enabled due to the initial enabling token being used to ultimately fire a competing transition.

• This condition shown in Fig 28.
• If we assume the time for t_1 is less than that for t_2, then, when p_1 receives a token, the two timers would begin counting down.
• At some time (τ_1) in the future, the timer for t_1 would reach its zero value, resulting in the firing of t_1.
• Since the token enabling t_2 is now gone, t_2 is

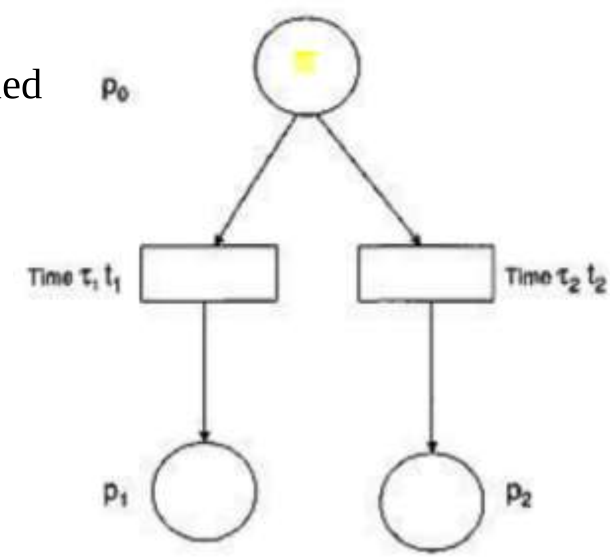


Fig 28: Timed Petri net with conflict

no longer enabled and, therefore, its timer (τ_2) would stop ticking down.

• The question now is what to do with transition t_2's timer.

Two possibilities

1) The first is to simply reset the timer on the next cycle.

• In this case, unless place $\mathbf{p_l}$ has a state where it has more than one token present, transition $\mathbf{t_2}$ will never fire.

2) Allow transition t_2's timer to maintain the present clock timer value $(\boldsymbol{\tau_2 - \tau_1})$.

• When the next token is received at p_l, if the remaining time in t_2's timer is less than t_1's timer, then $\mathbf{t_2}$ will fire, leaving $\mathbf{t_1}$ with the remaining time $(\boldsymbol{\tau_1 - (\tau_2 - \tau_l)})$.

The choice of which protocol to use will depend on the system one wishes to model.

FINALLY