# Petri Net System Modeling and Analysis

報告人:沈榮麟
國立臺北大學電機資訊學院資訊工程學系
(Ph. D., Professor, *Fellow*, IET;
*Senior Member*, IEEE, ACM, IEICE)

# Abstract

- Petri nets are a promising tool for describing and studying information processing systems that are characterized as being concurrent, asynchronous, distributed, parallel, nondeterministic, and/or stochastic. This talk starts with a brief review of the history and the application areas. It then proceeds with introductory modeling examples, behavioral and structural properties, three methods of analysis. Also, it presents introductory discussions on high-level Petri nets with applications to logic programming and fuzzy logic systems.

# Outlines

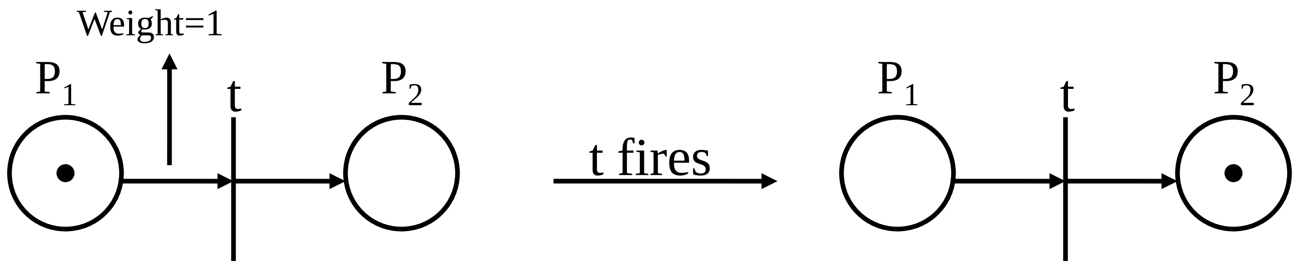
- Introduction
- Properties
- Methods of Analysis
- Predicate/ Transition Net- Logic Programs
- High-Level Fuzzy Petri Net- Fuzzy Logic
- Applications
- Conclusion

# 1. *Introduction*

- Brief History – 1962, Dr. Carl Adam Petri, Univ. of Bonn.
- *Definition 1.1:* Let $P$, $T$, $F$ be finite sets. The triple $PN = (P, T, F)$ is called a *Petri net structure* if and only if the following conditions hold: (a *bipartite* graph)
- (1) $P \cap T = \varnothing$,
- (2) $P \cup T \neq \varnothing$,
- (3) $F \subseteq (P \times T) \cup (T \times P)$,
- (4) $\text{domain}(F) \cup \text{codomain}(F) = P \cup T$.
- $P$ and $T$ are called the sets of *places* and *transitions* of $N$, respectively. $F$ is called the *flow relation* and the elements of $F$ are called the *arcs* of $PN$.

- A graphical and mathematical model, or data structure

Exp. 1：




P1 P2

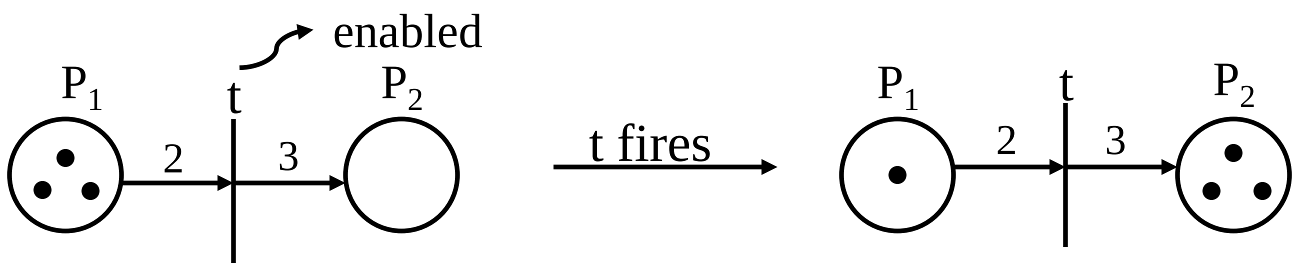
$M_0$=〔1 0〕= Initial Marking

P1 P2

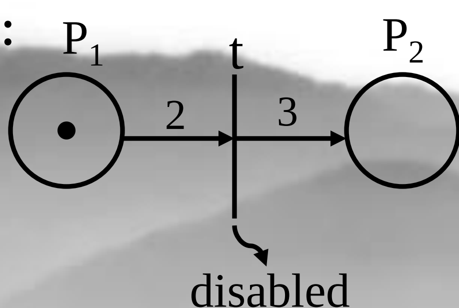
$M_f$=〔0 1〕= Final Marking

● Token (表fact or data)

Exp. 2：




Exp .3：




t cannot fire

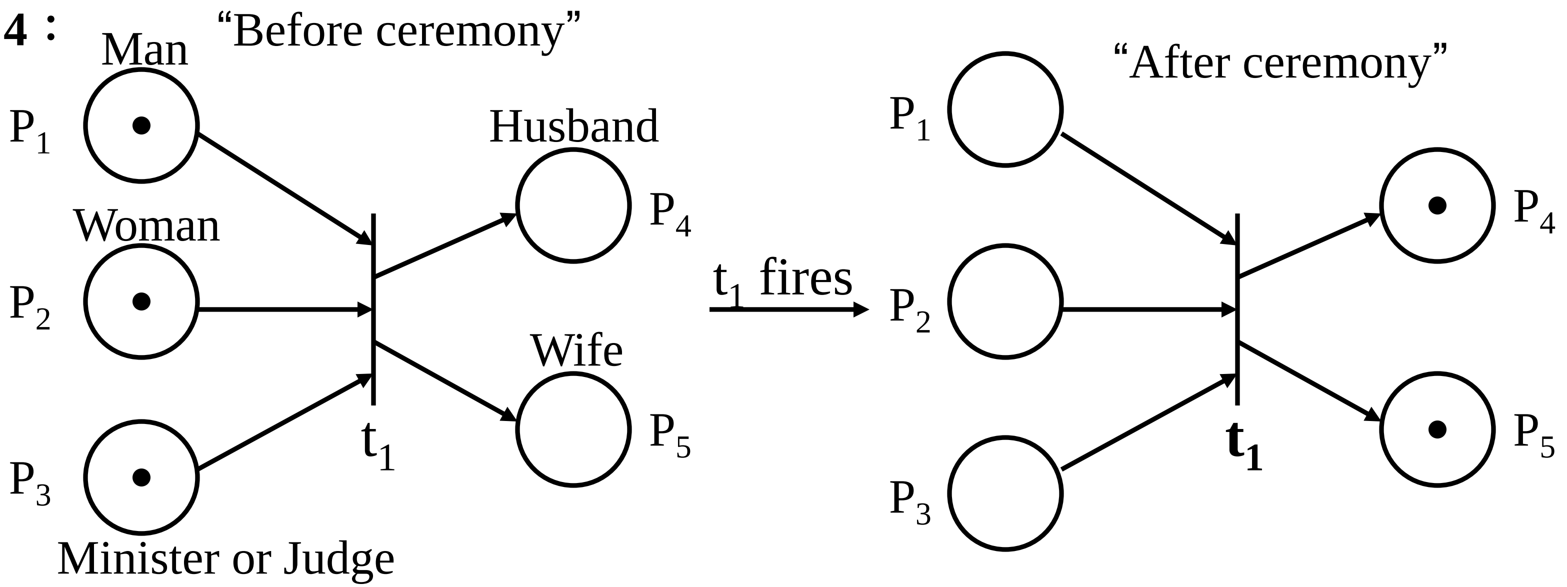
# 1. *Introduction (continued)*

- **Some Typical Interpretations of Transitions and Places**

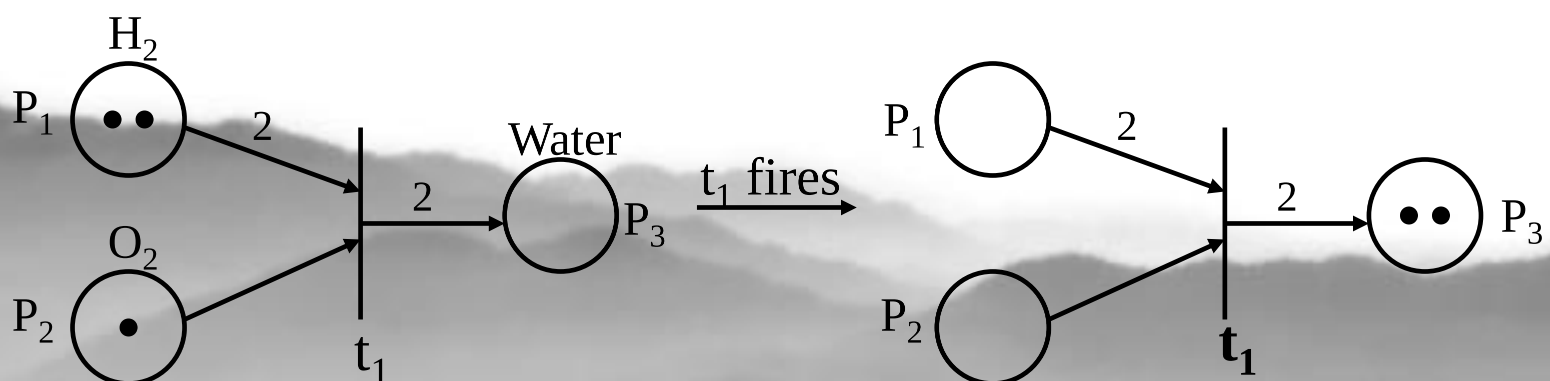
| Input Places | Transition | Output Places |
|---|---|---|
| Preconditions | Event | Postconditions |
| Input data | Computation step | Output data |
| Input signals | Signal processor | Output signals |
| Resources needed | Task or job | Resources released |
| Conditions | Clause in logic | Conclusion(s) |
| Buffers | Processor | Buffers |

- **Examples:**

**Exp. 4：**




**Exp. 5：** $2H_2 + O_2 \rightarrow 2H_2O$



# 2. Properties

- Behavioral Properties-
- 1) Reachability: $M \in R(M_0)$ ? $M = A^T x + M_0$
- 2) Boundedness: $M(p) < \infty$, $\forall p \in P$, $\forall M \in R(M_0)$ (or Safeness)
- 3) Liveness: different levels of liveness, $L$0, 1, 2, 3, 4
- 4) Reversibility and Home State: $\forall M \in R(M_0)$, $M_0 \in R(M)$
- 5) Coverability: $\forall M_1 \in R(M_0)$, $M_1(p) \geqq M(p)$, $\forall p \in P$
- 6) Persistence: any enabled transition can be disabled by its own firing.
- 7) Synchronic Distance: between two transitions $t_1$ and $t_2$.
- $d_{12} = \max \{\sigma(t_1) - \sigma(t_2)\}$

## "Reachability"

$$A = \text{Incidence Matrix} = \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \end{array} \begin{array}{c} \begin{array}{cccc} P_1 & P_2 & P_3 & P_4 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 0 \\ 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & -1 \end{pmatrix} \end{array}$$

$\sigma_1$ = Firing Sequence : $t_1 t_2 t_3 t_4$

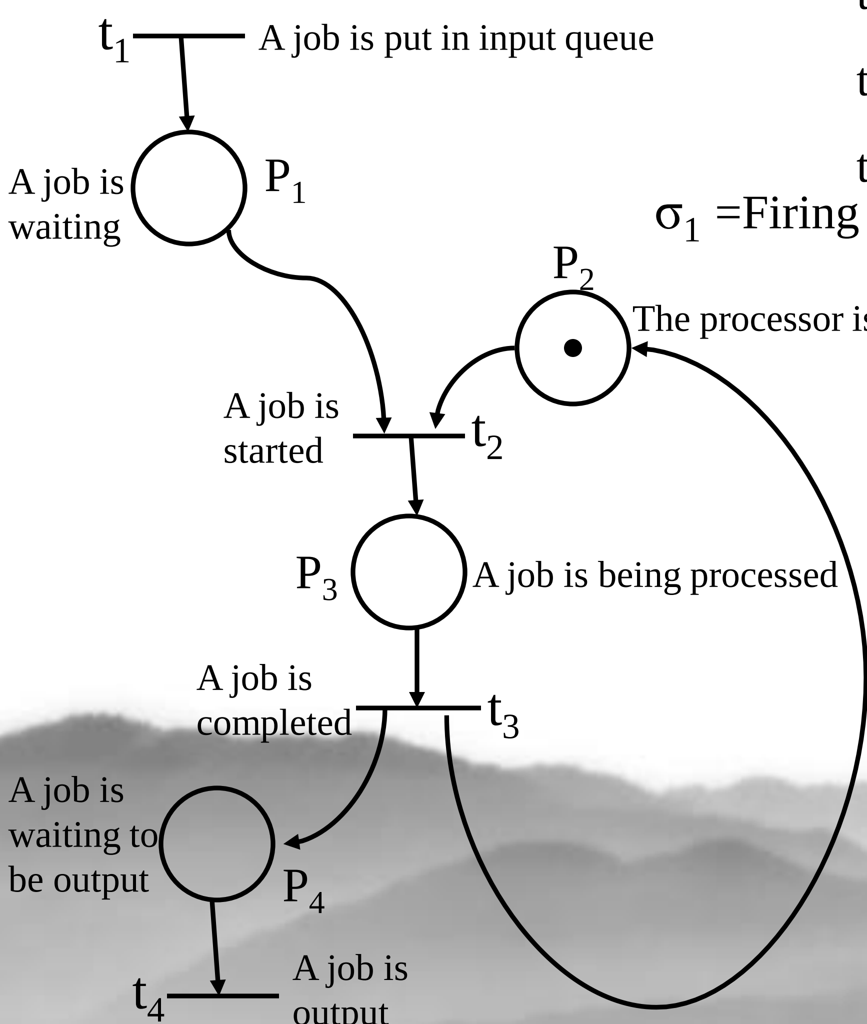



**Figure 3.3** Modeling of a simple computer system.

## "Reachability Graph"

| | $P_1$ | $P_2$ | $P_3$ | $P_4$ |
|---|---|---|---|---|
| $M_0=$ | [ 0 | 1 | 0 | 0 ] |
| $\downarrow t_1$ | | | | |
| $M_1=$ | [ 1 | 1 | 0 | 0 ] |
| $\downarrow t_2$ | | | | |
| $M_2=$ | [ 0 | 0 | 1 | 0 ] |
| $\downarrow t_3$ | | | | |
| $M_3=$ | [ 0 | 1 | 0 | 1 ] |

($t_4$: $M_3 \to M_0$)

Exp. $M_2 \overset{?}{=} A^T \cdot X + M_0$

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

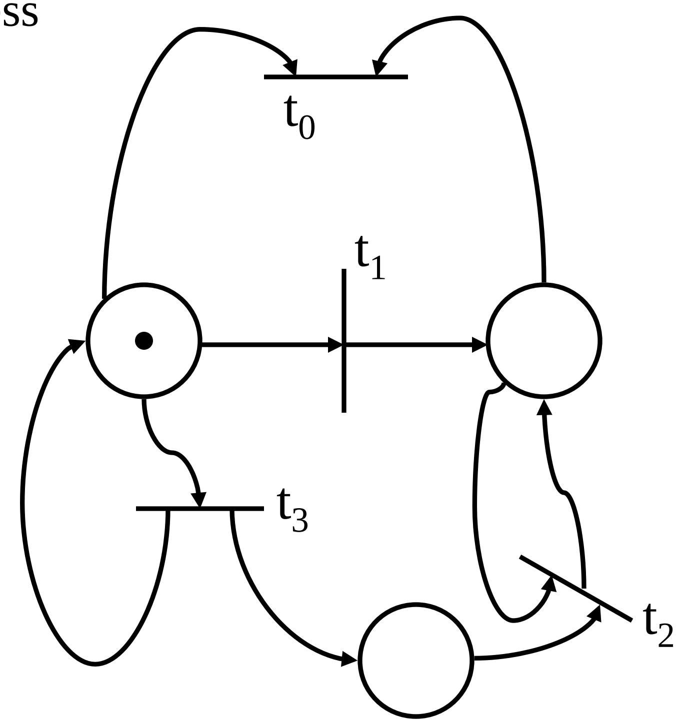
$$= \begin{bmatrix} 0 \\ -1 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$\therefore \sigma_2 = t_1 t_2$

$\therefore$ $M_2$ is reachable from $M_0$.

**"Liveness"** — different levels of liveness

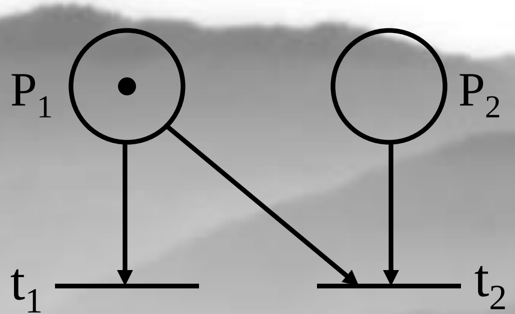
- L0 : dead
- L1 : at least once
- L2 : at least k times for a given $k \in N$
- L3 : $\infty$ - times
- L4 : L1 for all $M \in R(M_0)$




L4 ⟹ L3 ⟹ L2 ⟹ L1 ⟹ L0

| L4 | L3 | L2 | L1 | L0 |
|---|---|---|---|---|
| ($\forall M \in R(M_0)$, once) | ($\infty$ - times) | (k - times) | (once) | (dead) |
| | $t_3$ | $t_2$ | $t_1$ | $t_0$ |

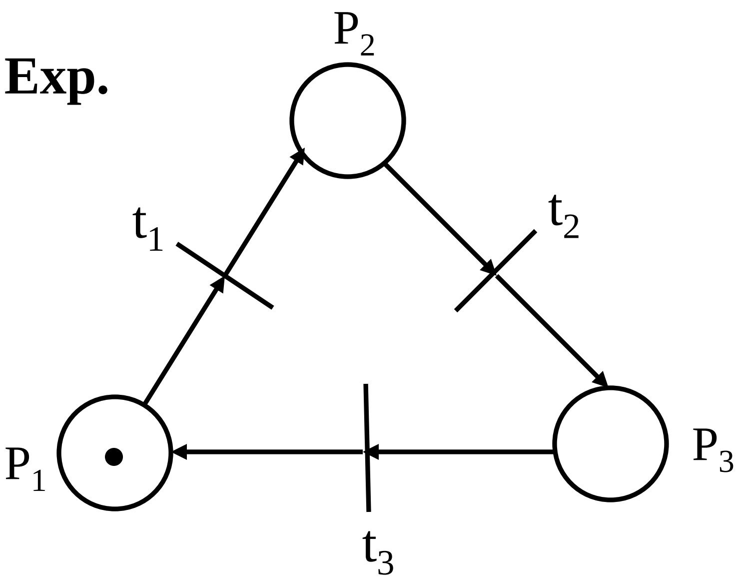
**"Persistence"**



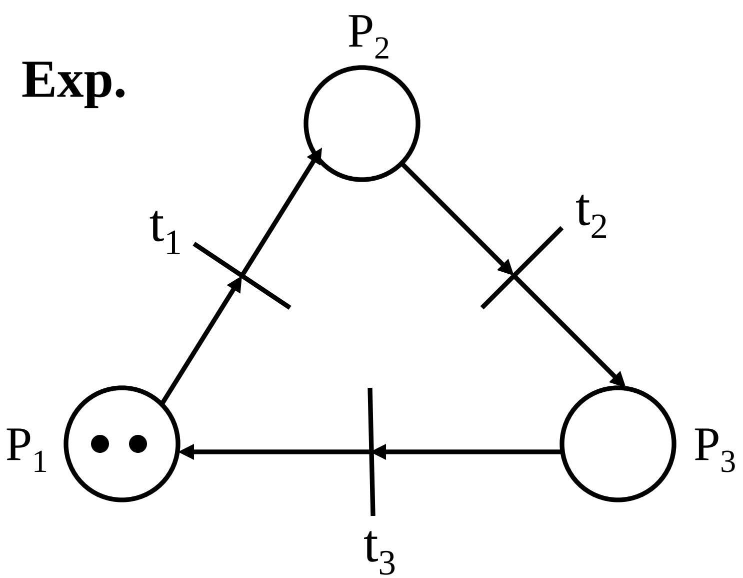

**a persistent PN.**

# "Synchronic Distance"

**Exp.**




**Exp.**




$\because \sigma = t_1 t_2 t_3 t_1 \ldots$

$\therefore d_{12} = \max| 2-1 | = 1$

$\because \sigma = t_1 t_1 t_2 t_2 t_3 t_3 t_1 t_1 \ldots$

$\therefore d_{12} = \max| 4-2 | = 2$

$$\mathbf{d_{12} = \max\{\sigma(t_1) - \sigma(t_2)\}}$$

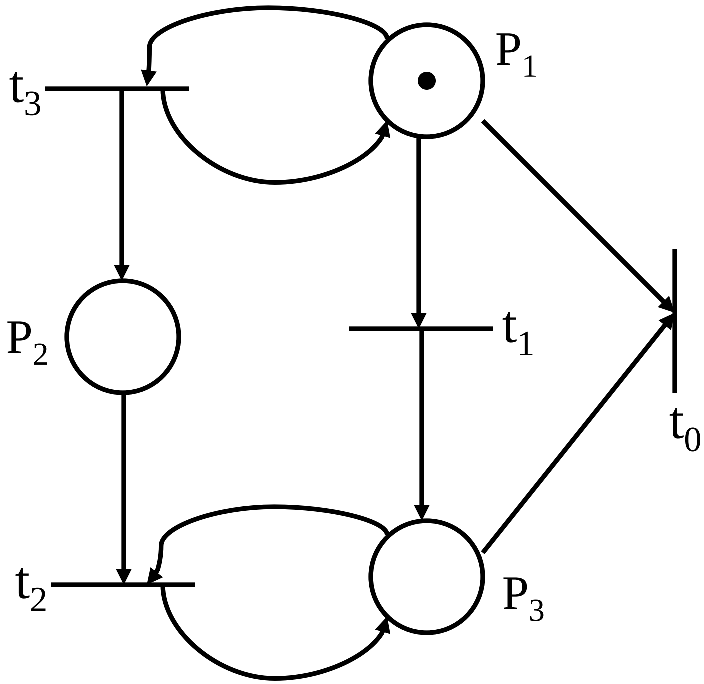
- **Structural Properties**
- **1) Structural Liveness: if there exists a live initial marking.**
- **2) Controllability: if any marking $M_n$ in PN is reachable from any initial marking $M_0$, Rank A=|p|**
- **3) Structural Bouundedness: if it exists > 0, Ay $\leqq$ 0**

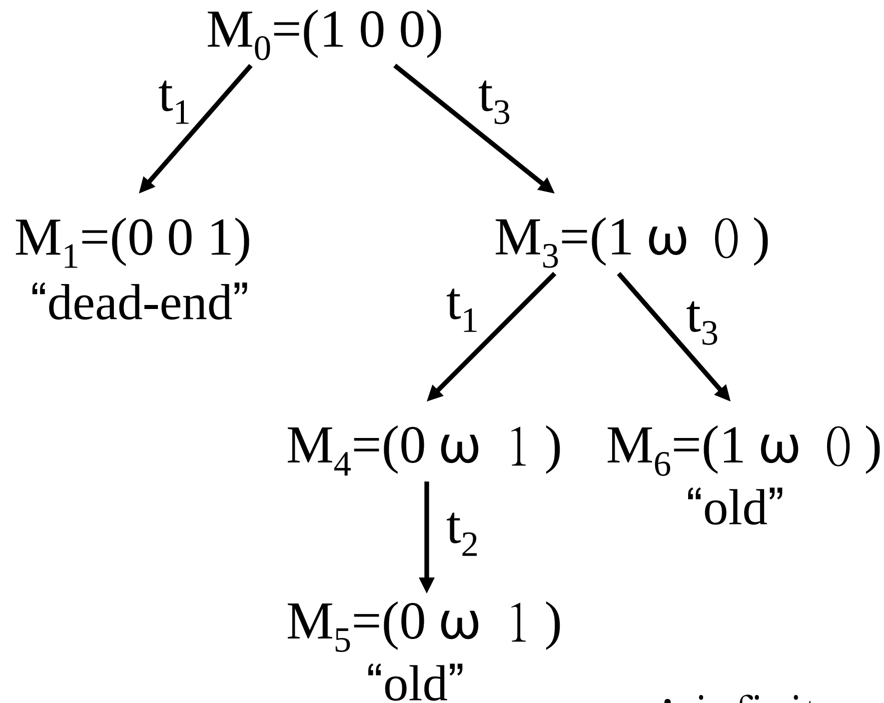
  **Y: place-vector**

# 3. Methods of Analysis

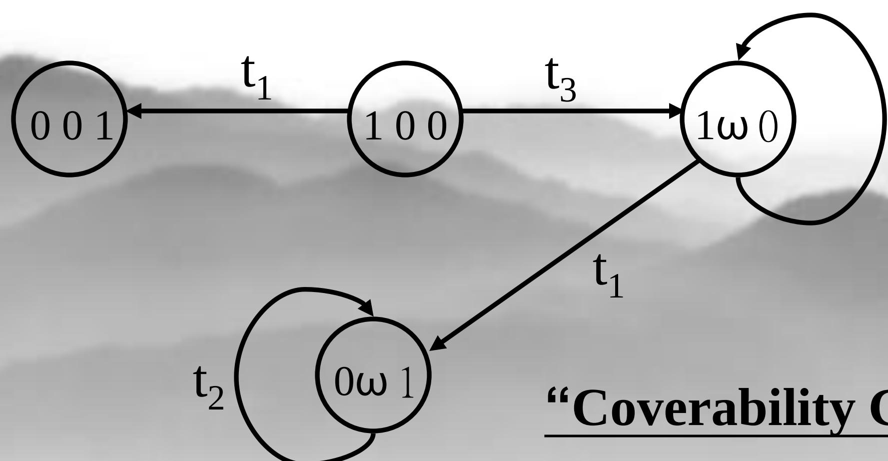
- Coverability Tree
- Incidence Matrix and State Equation
- Reduction Rules for Analysis

## “Coverability Tree”



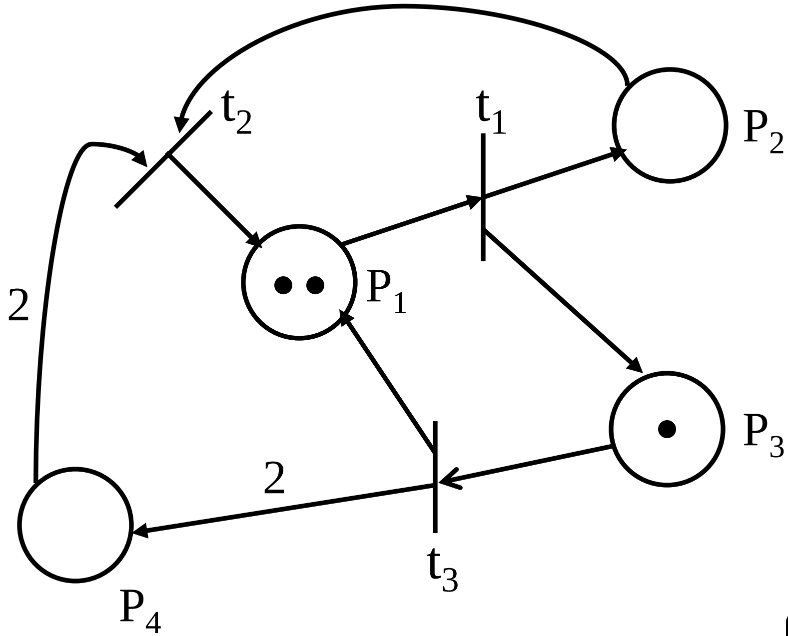




ω : infinity




## “Coverability Graph”

# "Incidence Matrix and State Equation"




$M_k = M_{k-1} + A^T \cdot X$ , k= 1,2,…

X : transition firing count vector

$A^T \cdot X = M_k - M_{k-1} = \triangle M$

$\triangle M = 0 \Rightarrow A^T \cdot X = 0$ …T-invariant

$A \cdot y = 0$ …S-invariant

$$
\begin{array}{c} \\ M_0 = \end{array}
\begin{array}{cccc} P_1 & P_2 & P_3 & P_4 \\ (\,2 & 0 & 1 & 0\,) \end{array}
$$

$$\downarrow t_3$$

$$M_1 = (\,3 \quad 0 \quad 0 \quad 2\,)$$

$$\downarrow$$

$$\vdots$$

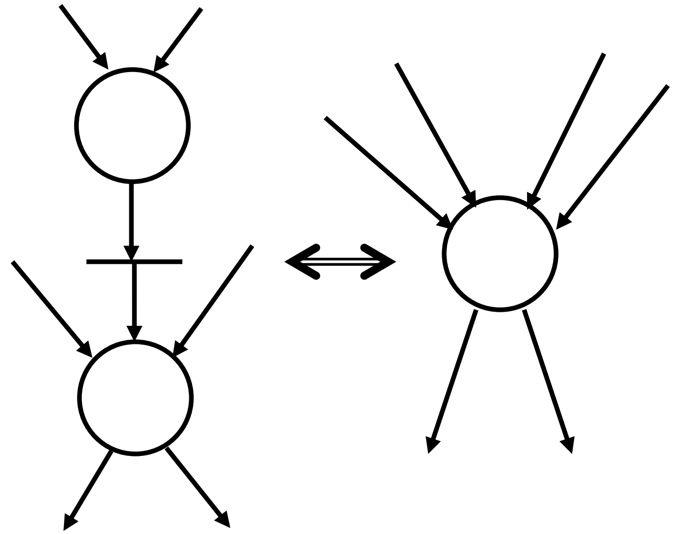
$M_1 = M_0 + A^T \cdot X$

$$
\begin{pmatrix} 3 \\ 0 \\ 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \end{array} \begin{array}{c} \begin{array}{ccc} t_1 & t_2 & t_3 \end{array} \\ \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & -2 & 2 \end{pmatrix} \end{array} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \begin{array}{c} t_1 \\ t_2 \\ t_3 \end{array}
$$

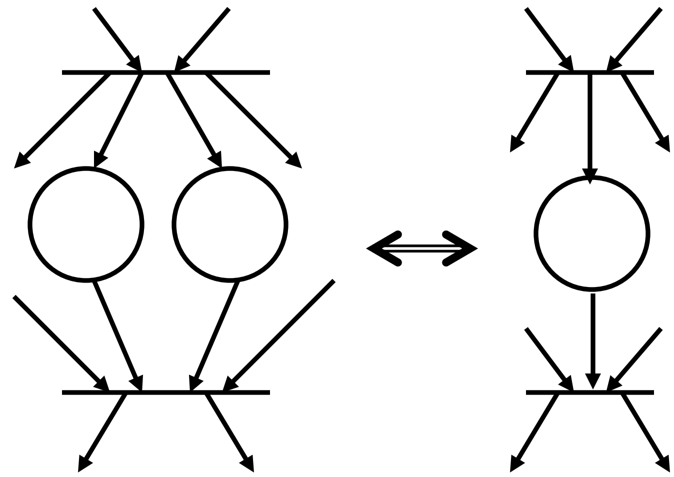
$\therefore M_1$ is reachable from $M_0$ .

## "Reduction Rules for Analysis"：preserving liveness, safeness, and boundedness

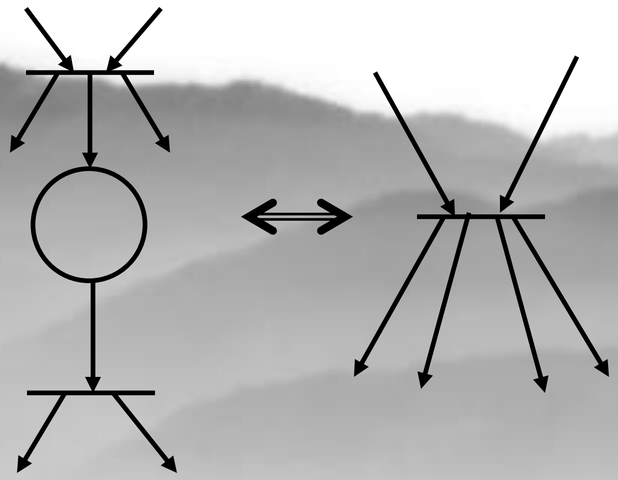
### 1)Fusion of Serial Places (FSP)



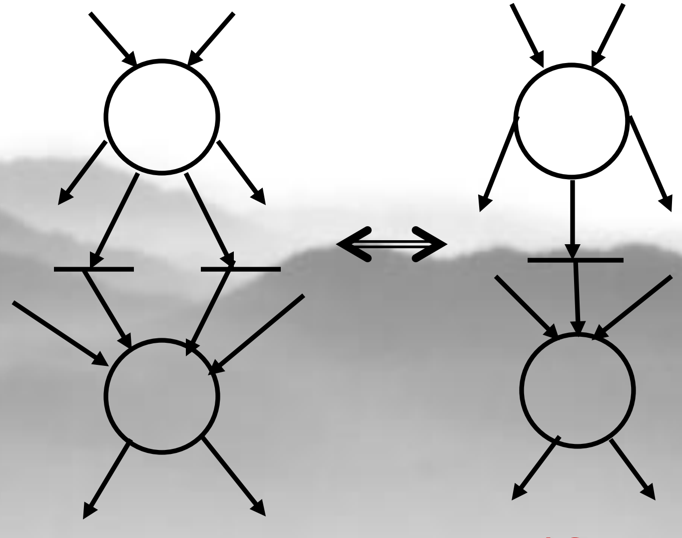
### 3)Fusion of Parallel Places (FPP)



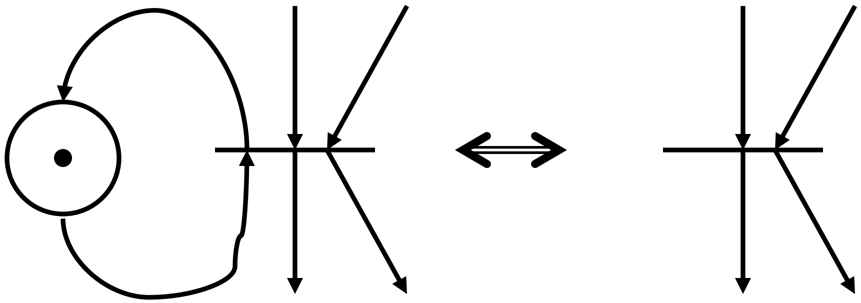
### 2)Fusion of SerialTransitions(FST)



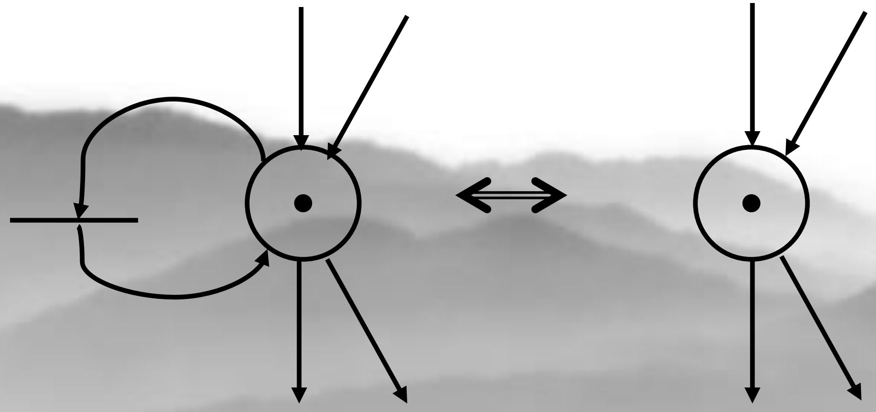
### 4)Fusion of Parallel Transitions(FPT)



## 5)Elimination of Self-loop Places(ESP)



## 6)Elimination of Self-loop Transitions(EST)



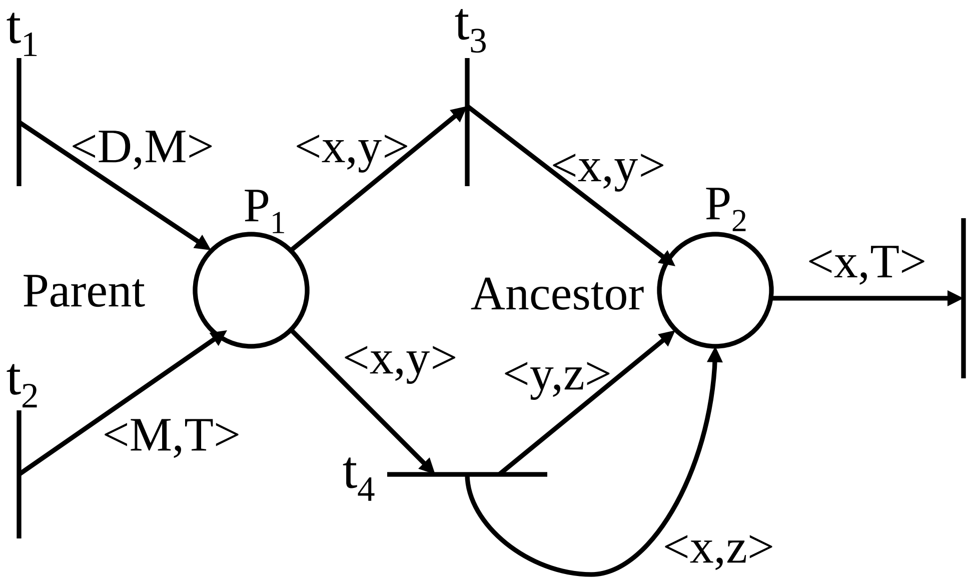
# 4. Predicate/Transition Net ( PrT Net )

- **Definition 4.1:* Let $\Omega = (P, T, F, L)$ be a 4-tuple such that
- (1) $(P, T, F)$ is a Petri net structure,
- (2) $L : F \rightarrow W \cup \ldots \cup W\ n$, where $W$ is a set of terms and $n$ is some integer value,
- (3) $\forall p \in P$ and $\forall x, y\ \ \{L((a,b)) \mid \{p\} \cap \{a, b\} \neq \varnothing\}$, we have $\propto(x) = \propto(y)$,
- then $\Omega$ is called a predicate/transition net (PrT net). The elements of $P$, $T$, $F$, and $L$ are called $\Omega$-predicates, $\Omega$-transitions, $\Omega$-arcs and $\Omega$-arc-labels, respectively. $\propto(x)$ in (3) defines the arity of the $\Omega$-predicate $p \in P$.
- In particular, a place can be viewed as a *precondition* or *postcondition*; a transition as an *event*; and an arc-label as a *data item*.

# Example 4.1 :

- Parent (David, Mary) ←
- Parent (Mary, Tom) ←
- Ancestor (x, y) ← Parent (x, y)
- Ancestor (x, z) ← Parent (x, y), Ancestor (y, z)
- ← Ancestor (x, Tom)

## Exp.4.1-PrT Net




$A^T \cdot X=0 \ldots$ T-invariant

$$X_1=\begin{array}{c} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{array}\begin{pmatrix} \psi \\ \{\} \\ \{M|x,T|y\} \\ \psi \\ \{M|x\} \end{pmatrix} \quad \sigma_1=t_2t_3t_5$$

$$A=\begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{array}\begin{array}{cc} \text{Parent}(P_1) & \text{Ancestor}(P_2) \\ \hline \langle D,M\rangle & 0 \\ \langle M,T\rangle & 0 \\ -\langle x,y\rangle & \langle x,y\rangle \\ -\langle x,y\rangle & -\langle y,z\rangle+\langle x,z\rangle \\ 0 & -\langle x,T\rangle \end{array}$$

$$X_2=\begin{array}{c} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{array}\begin{pmatrix} \{\ \} \\ \{\ \} \\ \{M|x,T|y\} \\ \{D|x,M|y,T|z\} \\ \{D|x\} \end{pmatrix} \quad \sigma_2=t_1t_2t_3t_4t_5$$

∴Answer：x=Mary&David

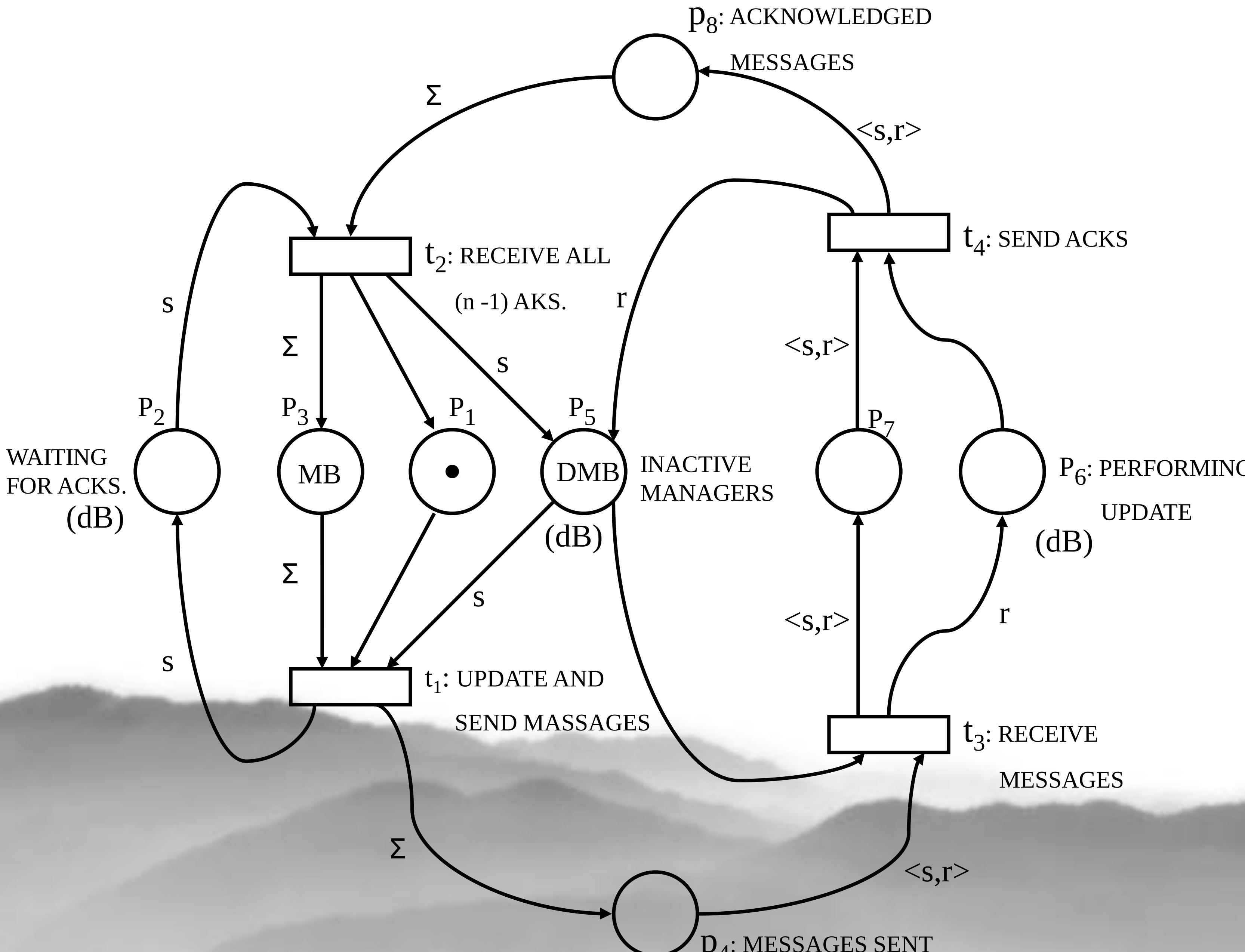



**Figure 3.21. A distributed database system model.**

# 5. High-Level Fuzzy Petri Net

- Definition 5.1: *HLFPN*
- The high-level fuzzy Petri net (HLFPN) is defined as an 8-tuple:
- $HLFPN = (P, T, F, C, V, \alpha, \beta, \delta)$,
- where
- $P = \{p_1, p_2, p_3, ..., p_k\}$ is a finite set of places;
- $T = \{t_1, t_2, t_3, ..., t_l\}$ is a finite set of transitions;
- $P \cup T \neq \emptyset$;
- $F \subseteq (P \times T) \cup (T \times P)$ is called the *flow relation* and is also a finite set of arcs, each representing the fuzzy set (i.e. fuzzy term) for an antecedent or a consequent; where the positive arcs (i.e. *THEN* parts) are denoted by $\rightarrow$;
- $C =$ a finite set of linguistic variables, e.g. $X$, $Y$, and $Z$;
- where $X = \{x_1, x_2, ..., x_n\}$, $Y = \{y_1, y_2, ..., y_m\}$, $Z = \{z_1, z_2, ..., z_q\}$;
- $V$ is known as the *fuzzy relational matrix* between the antecedent and the consequent of each rule;
- $\alpha : P \rightarrow C$ is an association function mapping from places to linguistic variables,

# 5. High-Level Fuzzy Petri Net (continued)

- $\alpha(p_i) = c_i$, $i = 1, ..., n$, where $C = \{ c_i \}$ is the set of linguistic variables in the knowledge base ( KB ) and $n$ denotes the number of linguistic variables in KB;
- $\beta : F \rightarrow [0, 1]$ is an association function mapping from the flow relation to the fuzzy truth values between zero and one (i.e. fuzzy sets);
- $\delta : T \rightarrow V$ is an association function mapping from the transition to the fuzzy relational matrix.

- * *R1*: IF it($X_1$) is hot($A_1$) AND the sky($X_2$) is cloudy($A_2$) THEN the humidity($Y$) is high($B$).
- $R_1$': IF $X_1(A_1)$ AND $X_2(A_2)$ Then $Y(B)$.

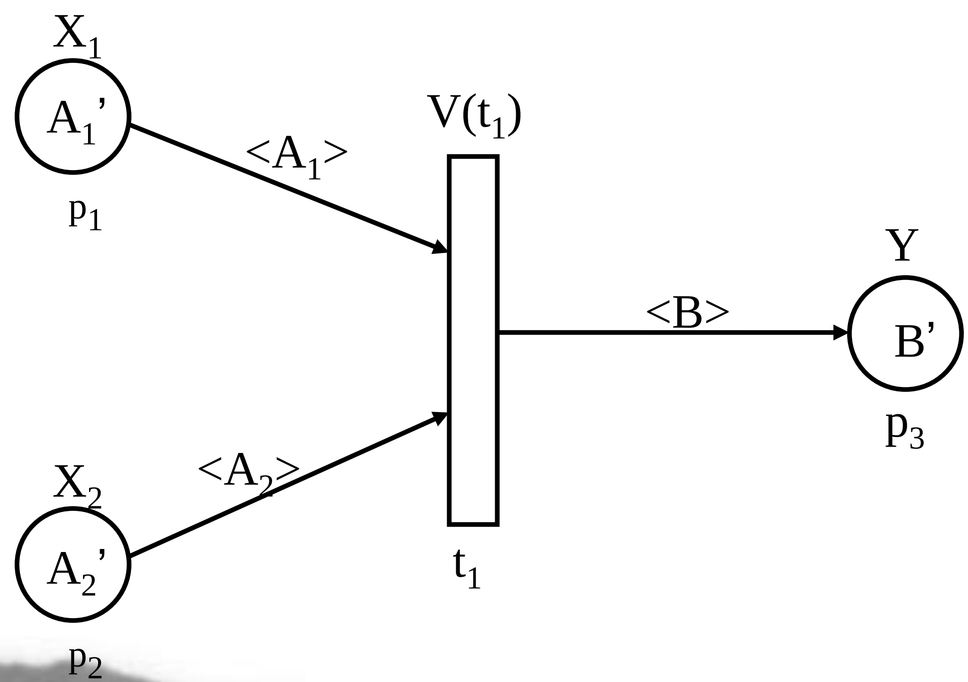



**Fig. 1.HLFPN for Example 2.2.**

# Example 5.1 :

- Consider that a set of three rules is shown as follows.
- $R_1$: IF $X_1$ is $A$ AND $X_2$ is $B$ THEN $Y_1$ is $C$.
- $R_2$: IF $X_2$ is $B$ AND $X_3$ is $D$ THEN $O$ is $E$.
- $R_3$: IF $Y_1$ is $C$ THEN $O$ is $F$.

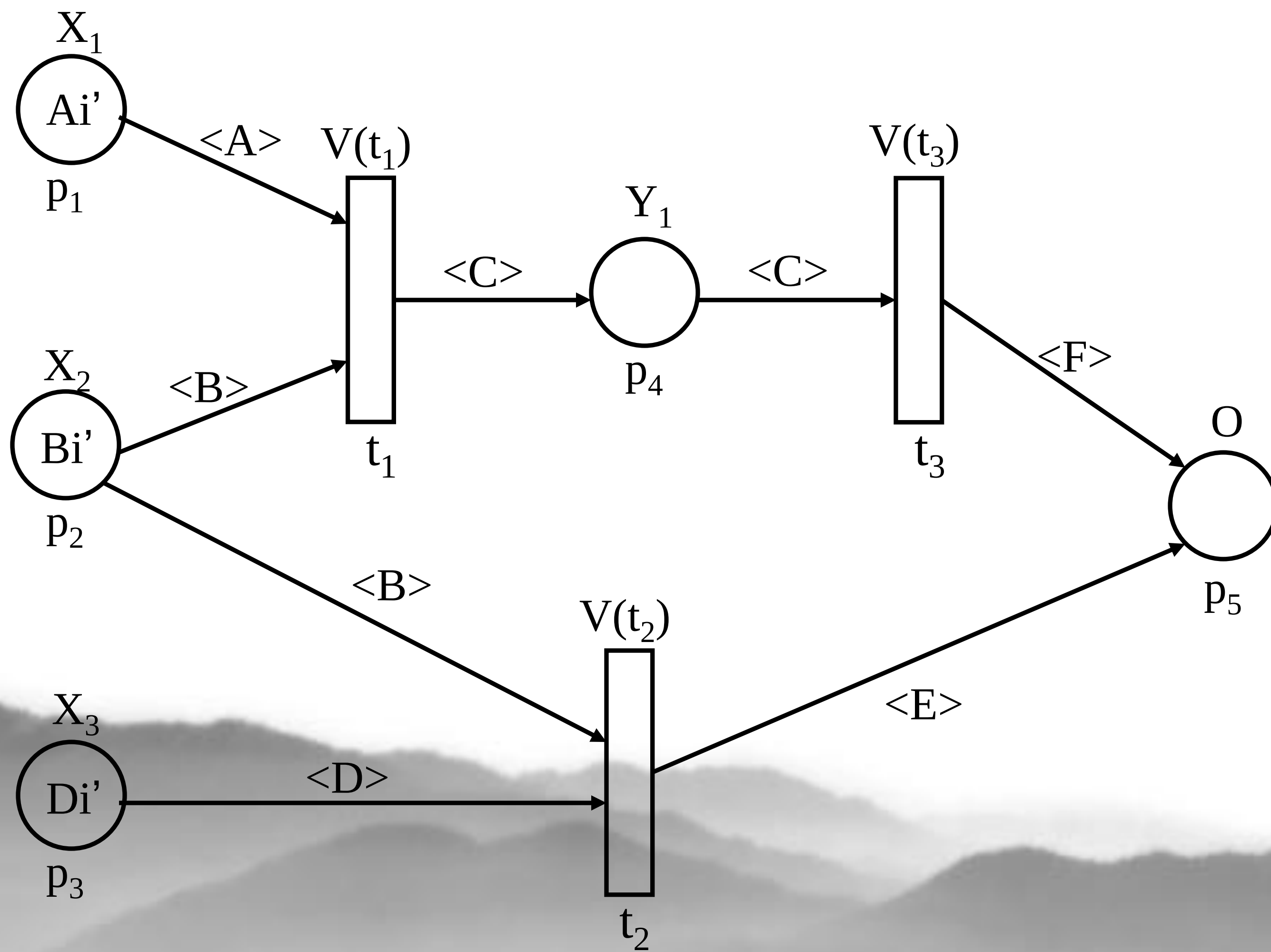



**Fig. 8. HLFPN for Example 5.1.**

# Example 5.2 : Temperature Control System

- In the temperature control system, a sensor is used to measure the current temperature in a room and obtain the temperature error by deducting the desired temperature from the real current temperature. Moreover, an electric heater is used to adjust the room temperature by tuning its electric power.
- Assume that the temperature error(E) is an input linguistic variable with fuzzy terms: high(H), zero(Z), and low(L);and the electric power(P) is an output linguistic variable with fuzzy terms: large(LA), medium(M), and small(S). Consider the following fuzzy production rules:
- $R_1$ : IF E is H THEN P is S.
- $R_2$ : IF E is L THEN P is LA.
- $R_3$ : IF E is Z THEN P is M.
- Fig. 13 illustrates an example of the HLFPN model representing the above three rules.

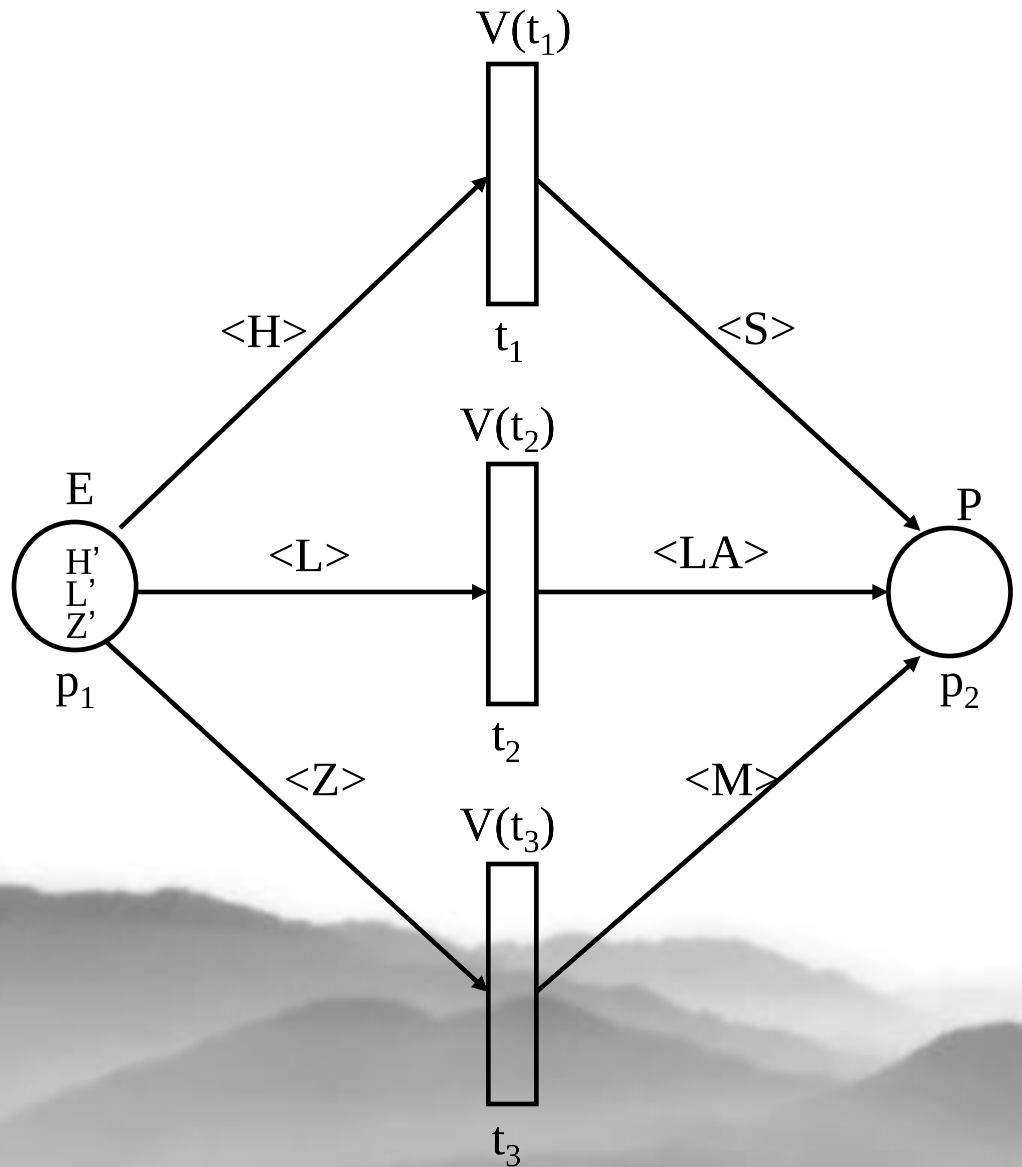



**Fig. 13.HLFPN for Example 5.2.**

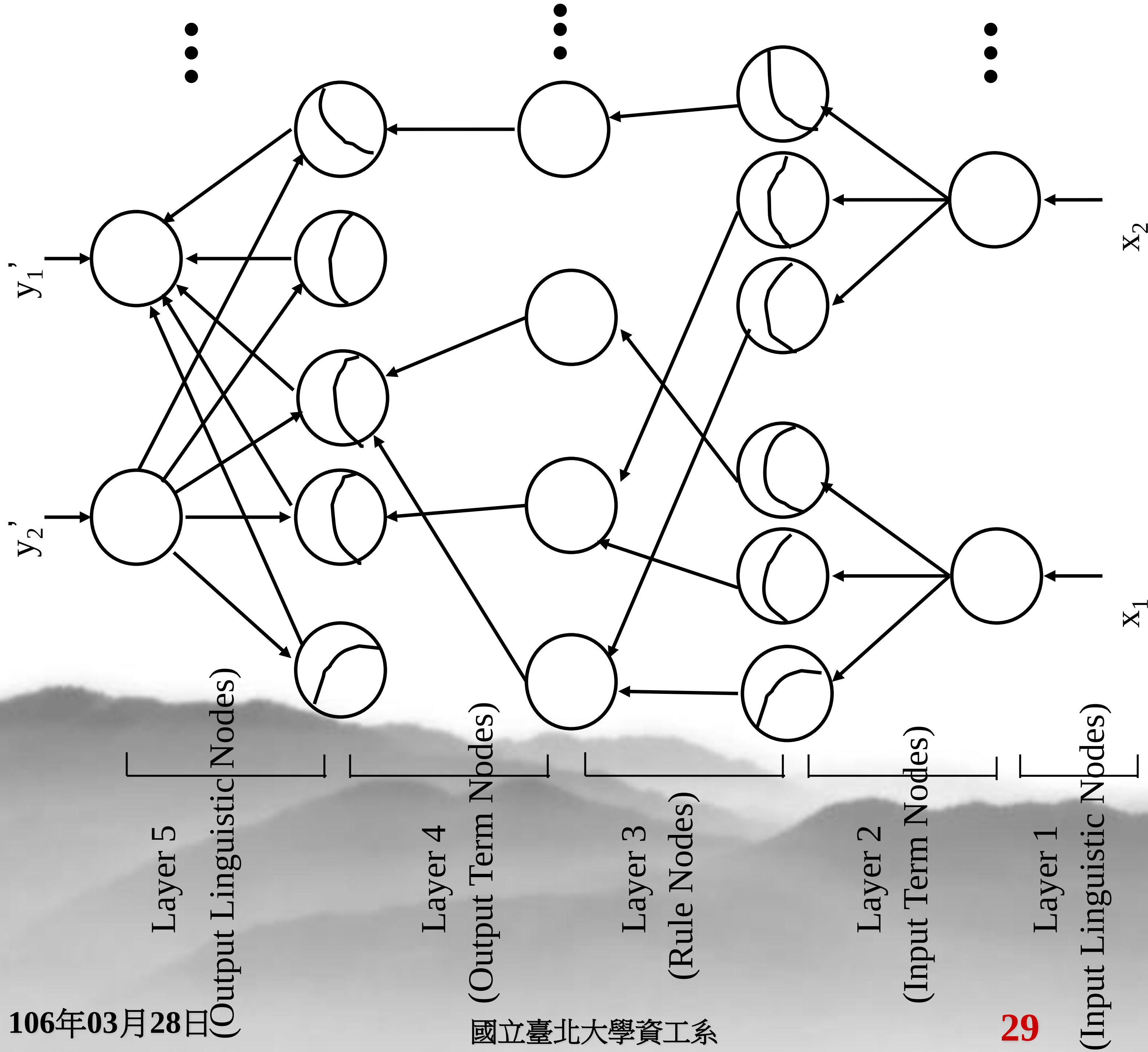



**Fig. 5. FALCON Structure.**

# Table 1: Functional Comparisons

| Items / Approaches | Data Structure | Information Storage | Type of Fuzzy Sets | Outputs Generation | Applicability | Learning Flexibility |
|---|---|---|---|---|---|---|
| HLFPN | Compact | Small | Discrete | Parallel | Broad | Yes |
| FALCON | Complex | Large | Continuous | Serial | Limited | No |

# Table 2: HLFPN *vs.* FALCON

| Average CPU Time (seconds) | HLFPN | | FALCON | |
|---|---|---|---|---|
| | Supervised | Unsupervised | Supervised | Unsupervised |
| *Fuzzy Car* | 3.6 | 4.7 | 5.9 | 6.3 |
| *Manufacturing Scheduling* | 2.3 | 3.8 | 4.8 | 6.1 |
| *Inverted Pendulum* | 3.7 | 4.9 | 6.2 | 7.9 |
| *Turning Force System* | 8.7 | 9.8 | 10.2 | 12.7 |

# 6. Applications

- Specifications and Verifications
- Computer Operating Systems
- Computer Software Systems
- Computer Hardware Systems
- Legal Systems
- Office Information System
- Formal Language Theory
- Communication Protocols
- Operational Research
- Chemical Systems
- Distributed Database Systems
- Factory Automation
- Performance Evaluation

# 7. Conclusions

* Verification, Simulation, and Performance Evaluation

* Natural Science and Social Science

* Broad Applications

* Thank you all for your attention!

# References


- [1] S.I. Ahson, "Petri net models of fuzzy neural networks," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 25, no. 6, pp. 926- 932, June 1995.
- [2] S.M. Chen, J.S. Ke, and J.F. Chang, "Knowledge representation using fuzzy Petri nets," *IEEE Trans. on Knowledge and Data Engineering*, vol. 2, no. 3, pp. 311-319, Sep. 1990.
- [3] S.M. Chen and Y.S. Shiau, "Knowledge representation using extended fuzzy Petri nets," *Proc. of International Conference on Artificial Intelligence*, pp. 224- 231, Dec. 19-21, 1996.
- [4] S.M. Chen, "Fuzzy backward reasoning for fuzzy Petri nets," *IEEE Trans. on Systems, Man, and Cybernetics- Part B: Cybernetics*, vol. 30, no. 6, pp. 846-856, Dec. 2000.
- [5] S.M. Chen, "Weighted fuzzy reasoning using weighted fuzzy Petri nets," *IEEE Trans. on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 386-397, Mar./Apr. 2002.
- [6] J. Lee and L.F. Lai, "A high-level Petri nets-based approach to verifying task structures," *IEEE Trans. on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 316-335, Mar./Apr. 2002.
- [7] A. Konar and A.K. Mandal, "Uncertainty management in expert systems using fuzzy Petri nets," *IEEE Trans. on Knowledge and Data Engineering*, vol. 8, no. 1, pp. 96- 105, Feb. 1996.
- [8] T. Murata and D. Zhang, "A predicate-transition net model for parallel interpretation of logic programs," *IEEE Trans. Software Engineering*, vol. 14, no. 4, pp. 481- 497, Apr. 1988.
- [9] T. Murata, "Petri nets: properties, analysis, and applications," *Proc. of the IEEE*, vol. 77, no. 4, pp. 541-580, Apr. 1989.
- [10] J. L. Peterson, *Petri Net Theory and the Modeling of Systems*, Englewood Cliffs, NJ: Prentice-Hall, Inc., 1981.
- [11] C.V. Ramamoorthy and G.S. Ho, "Performance evaluation of asynchronous concurrent systems using Petri nets," *IEEE Trans. on Software Engineering*, vol. SE-6, no. 5, pp.440-449, Sep. 1980.
- [12] H. Scarpelli, F. Gomide, and R.R. Yager, "A reasoning algorithm for high-level fuzzy Petri nets," *IEEE Trans. on Fuzzy Systems*, vol. 4, no. 3, pp. 282- 294, Aug. 1996.
- [13] J. Lee, K.-F.R Liu, and W. Chiang, "Modeling uncertainty reasoning with possibilistic Petri nets," *IEEE Trans. on Systems, Man and Cybernetics, Part B: Cybernetics*, vol. 33, no. 2, pp. 214- 224, Apr. 2003.
- [14] Xiaoou Li, Wen Yu, and F. Lara-Rosano, "Dynamic knowledge inference and learning under adaptive fuzzy Petri net framework," *IEEE Trans. on Systems, Man and Cybernetics, Part C: Applications and Reviews,* vol. 30, no. 4, pp. 442- 450, Nov. 2000.
- [15] M. Gao, M. Zhou, X. Huang, and Z. Wu, "Fuzzy reasoning Petri nets," *IEEE Trans. on Systems, Man, and Cybernetics- Part A: Systems and Humans*, vol. 33, no. 3, pp. 314-324, May 2003.
- [16] J. Lee, K.-F.R Liu, and W. Chiang, "Modeling uncertainty reasoning with possibilistic Petri nets," *IEEE Trans. on Systems, Man and Cybernetics, Part B: Cybernetics*, vol. 33, no. 2, pp. 214- 224, Apr. 2003.

# References


- [17] Y. Tang, M. C. Zhou, and M. Gao, " Fuzzy-Petri-net-based disassembly planning considering human factors," *IEEE Transactions on Systems, Man, and Cybernetics- Part A: Systems and Humans*, vol. 36, no. 4, pp. 718- 726, Jul. 2006.
- [18] W. Tan and M. C. Zhou, *Business and Scientific Workflows: A Service-Oriented Approach*, IEEE Press/Wiley, Hoboken, NJ, 2013.
- [19] N. Q. Wu and M. C. Zhou, *System Modeling and Control with Resource-Oriented Petri Nets*, CRC Press, New York, 2010.
- [20] B. Hruz and M. C. Zhou, *Modeling and Control of Discrete Event Dynamic Systems*, Springer, London, UK, 2007.
- [21] M. C. Zhou and K. Venkatesh, *Modeling, Simulation and Control of Flexible Manufacturing Systems: A Petri Net Approach.* World Scientific, Singapore, 1998.

# Selected Journal (SCI ) Papers

- 1. Victor R.L. Shen, Tzer-Shyong Chen, and Feipei Lai, (Oct. 1997) "Modified Cryptographic Key Assignment Scheme for a Group-Oriented Hierarchy," *IEICE Trans. on Fundamentals of Electronics, Communications, and Computer Science*, Vol. E80-A, No.10, pp. 2032-2034.
- 2. Victor R.L. Shen, Tzer-Shyong Chen, and Feipei Lai, (Oct. 1997) "Novel Cryptographic Key Assignment Scheme for Dynamic Access Control in a Hierarchy," *IEICE Trans. on Fundamentals of Electronics, Communications, and Computer Science*, Vol. E80-A, No.10, pp.2035-2037.
- 3. Victor R.L. Shen and Feipei Lai, (Oct.1998) "Requirements Specification and Analysis of Digital Systems Using Fuzzy and Marked Petri Nets," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 28, No. 5, pp. 748- 754.
- 4. Victor R.L. Shen, F. H. Kuo, and Feipei Lai, (Mar.1998) "Requirements Specification and Analysis of Digital Systems Using FARHDL," *IEICE Trans. on Information and Systems*, Vol. E81-D, No. 3, pp. 317-328.
- 5. Victor R.L. Shen, (Jul. 1999) "Incremental CTL Model Checker for Fair States," *IEICE Transactions on Information and Systems*, Vol. E82-D, No. 7, pp. 1126-1130.
- 6. F.H. Kuo, Victor R.L. Shen, Tzer-Shyong Chen, and Feipei Lai, (Sep. 1999) "A Cryptographic Key Assignment Scheme for Dynamic Access Control in a User Hierarchy," *IEE Procs. Computers and Digital Techniques*, Vol. 146, No. 5, pp. 235- 240.
- 7. Victor R.L. Shen, (July 2000), "Improvement of a Multi-Role-Based Access Control Policy," *IEICE Trans. on Fundamentals of Electronics, Communications, and Computer Science*, Vol. E83-A, No. 7, pp. 1483-1485.
- 8. Victor R.L. Shen, (Aug. 2000) "Correctness in Hierarchical Knowledge-Based Requirements," *IEEE Transactions on Systems, Man, and Cybernetics- Part B: Cybernetics*, Vol. 30, No. 4, pp. 625- 631.
- 9. Victor R.L. Shen, Frank H.C. Shen, (Jan. 2002) "Requirements Specification and Analysis of Fault-Tolerant Digital Systems," *IEEE Transactions on Systems, Man, and Cybernetics- Part A: Systems and Humans*, Vol. 32, No. 1, pp. 149- 159.
- 10. Victor R.L. Shen and Tzer-Shyong Chen, (Feb. 2002) "A Novel Key Management Scheme Based on Discrete Logarithms and Polynomial Interpolations," *Computers and Security*, United Kingdom, Vol. 21, No. 2, pp. 164- 171.
- 11. Victor R.L. Shen and T. S. Chen, (Oct. 2002) "A Novel Cryptosystem with Lock Generation and Sum-Difference Replacement Ladder," *IEICE Trans. on Information and Systems*, Vol. E85-D, No. 10, pp. 1719-1722. ( *SCI , EI* )
- 12. Victor R.L. Shen, (Apr. 2003) "Reinforcement Learning for High-Level Fuzzy Petri Nets," *IEEE Transactions on Systems, Man, and Cybernetics -Part B: Cybernetics*, Vol. 33, No. 2, pp. 351- 362.

# Selected Journal (SCI ) Papers

- 13. Victor R.L. Shen, T. S. Chen, and K. Q. Shi, (Jan. 2005) “A Novel Cryptosystem Based on Grey System Theory and Genetic Algorithm,” *Applied Mathematics and Computation* Vol.170, No. 2, pp. 1290-1302.
- 14. Victor R.L. Shen, (May 2006) "A PN-based Approach to the High-Level Synthesis of Digital Systems," *Integration, the VLSI Journal*, Vol. 39, No. 3, pp. 182-204.
- 15. Victor R.L. Shen, (Nov. 2006) “Knowledge Representation Using High-Level Fuzzy Petri Nets,” *IEEE Transactions on Systems, Man, and Cybernetics- Part A: Systems and Humans*, Vol. 36, No. 6, pp. 2120- 2127.
- 16. Victor R.L. Shen and Tony Tong-Ying Juang (Jan. 2008) “Verification of Knowledge-Based Systems Using Predicate/Transition Nets,” *IEEE Transactions on Systems, Man, and Cybernetics- Part A: Systems and Humans*, Vol. 38, No. 1, pp. 78- 87.
- 17. Victor R.L. Shen, Yu-Fang Chung, and T. S. Chen, (Feb. 2009) " A Novel Application of Grey System Theory to Information Security (Part I)," *Computer Standards and Interfaces*, Vol. 31, No. 1, pp. 277-281.
- 18. Victor R.L. Shen, Yu-Fang Chung, and T. S. Chen, (Feb. 2010) "A Novel Method of Grey Data Generating Techniques in Cryptosystem," *International Journal of Innovative Computing, Information and Control (IJICIC)*, Vol. 6, No. 2, pp. 501-508.
- 19. Victor R.L. Shen, Yue-Shan Chang, and Tony Tong-Ying Juang (Mar. 2010) "Supervised and Unsupervised Learning by Using Petri Nets," *IEEE Transactions on Systems, Man, and Cybernetics- Part A: Systems and Humans*, Vol. 40, No. 2, pp. 363-375.
- 20. Victor R.L. Shen and Cheng-Ying Yang, (Dec. 2010), “A Special Approach to Teach Artificial Intelligence,” *International Journal of Engineering Education*, Vol. 26, No. 6, pp. 1414-1421.
- 21. Victor R.L. Shen and Cheng-Ying Yang, (Apr. 2011), “An Intelligent Multiagent Tutoring System in Artificial Intelligence”, *International Journal of Engineering Education*, Vol. 27, No. 2, pp. 248-256.

# Selected Journal (SCI ) Papers

22. Victor R.L. Shen, Yu-Fang Chung, Tzer-Shyong Chen, and Yu-An Lin, (Sep. 2011), "A Blind Signature Based on Discrete Logarithm Problem," *International Journal of Innovative Computing, Information and Control (IJICIC)*, Vol. 7, No. 9, pp. 5403-5416.
23. Victor R.L. Shen, Cheng-Ying Yang, and Jia-Hong Lee, (Nov. 2012), "Analysis of the Problem Description in the Algorithm for Inventive Problem Solving Using Petri Nets," *Knowledge-Based Systems*, Vol. 35, pp.290-303.
24. Victor R.L. Shen, Yu-Ying Wang, Cheng-Ying Yang, and Szu-Tso Yeh, (Nov. 2012) "Verification of Problem-Based Learning Systems Using Modified Petri Nets," *Expert Systems with Applications*, Vol. 39, No. 16, pp. 12636-12649.
25. Victor R.L. Shen, Cheng-Ying Yang, Yu-Ying Wang, and Yu-Hsiang Lin, (Dec. 2012) "Application of High-Level Fuzzy Petri Nets to Educational Grading System," *Expert Systems with Applications*, Vol. 39, No. 17, pp.12935-12946.
26. Victor R.L. Shen, Jhong-Yi Guo, and Shyi-Ming Chen, (Sep. 2013) "A Novel Reduction Method for Petri Net Systems Based on Matching Theory," *Expert Systems with Applications*, Vol. 40, No. 11, pp.4562-4576.
27. Victor R.L. Shen, Horng-Yih Lai, and Ah-Fur Lai, (Jan. 2015) "The Implementation of a Smartphone-Based Fall Detection System Using a High-Level Fuzzy Petri Net," *Applied Soft Computing*, Vol. 26, pp. 390-400.
28. Victor R.L. Shen, Cheng-Ying Yang, and Chien Hung Chen, (Sep. 2015) "A Smart Home Management System with Hierarchical Behavior Suggestion and Recovery Mechanism," *Computer Standards and Interfaces*, Vol. 41, No. 1, pp. 98-111.

# Selected Journal (SCI ) Papers

29. Victor R.L. Shen and Wei-Chieh Huang, (Dec. 2015) "A Time-Bound and Hierarchical Key Management Scheme for Secure Multicast Systems," *Wireless Personal Communications*, Vol. 85, No. 4, pp. 1741–1764.
30. Victor R.L. Shen, Yu-Ying Wang, and Li-Ya Yu, (Feb. 2016) "A Novel Blood Pressure Verification System for Home Care," *Computer Standards and Interfaces*, Vol. 44, No. 2, pp. 42-53.
31. Victor R.L. Shen, Rong-Kuan Shen, Cheng-Ying Yang, and Gibran, (Oct. 2016) "Cost optimization of a path protection system with partial bandwidth using Petri nets," *Wireless Personal Communications*, Vol. 90, No. 3, pp. 1239-1259.
32. Victor R.L. Shen, Cheng-Ying Yang, Rong-Kuan Shen, and Yu-Chia Chen, (Feb. 2016) "Application of Petri Nets to Deadlock Avoidance in iPad-like Manufacturing Systems," accepted for publication online in *Journal of Intelligent Manufacturing*.
33. Victor R.L. Shen, Rong-Kuan Shen, Cheng-Ying Yang, and Wei-Cheng Chen, (Aug. 2016) "A Novel Fall Prediction System on Smartphones," accepted for publication in *IEEE Sensors Journal.*

# 有關 Petri Nets 的影片

- 
- https://www.youtube.com/watch?v=1IPOIE0PvQY
- 
- https://www.youtube.com/watch?v=EmYVZuczJ6k
- 
- https://www.youtube.com/watch?v=pgFfMc6J3YU
- 
- https://www.youtube.com/watch?v=uMBGm01arFs&t=10s

# Petri Net Tools

- **HPSim (**Henryk Petri Simulator**)**
- **PIPE 2 (**Platform Independent Petri net Editor 2**)**
- **WoPeD (**Workflow Petri net Designer**)**


敬　請　指　教