Petri Nets

Sources

Gang Quan

“Petri Nets: Properties, analysis and applications”,
by T. Murata, 1989)

Review

- Finite State Machine
  - What
  - Representation
  - Mealy/Moore
  - NFSM
  - Equivalence
  - Minimization

Petri Net

- Introduction
- Modeling Examples
- Properties
- Petri Net Extensions

Introduction

- Originated from Carl Adam Petri's dissertation in 1962
- A graphical and mathematical modeling tool
- An effective and promising tool for capturing system concurrent, asynchronous, distributed, parallel, nondeterministic, stochastic characteristics.
- A bridge between the practitioners and theoreticians
- Various applications:
  - Performance evaluation
  - System verification
  - Communication protocols
  - Distributed database, etc

What is a Petri Net

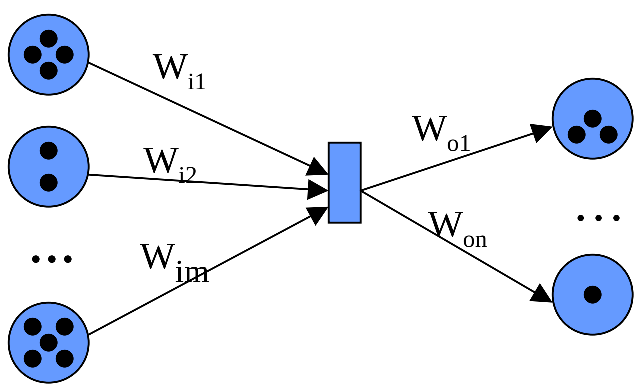
- **A directed, weighted, bipartite graph**
- **G = (V,E)**
  - **Nodes (V)**
    - places (shown as circles)
    - transitions (shown as bars)
  - **Arcs (E)**
    - from a place to a transition or from a transition to a place
    - labeled with a weight (a positive integer, omitted if it is 1)

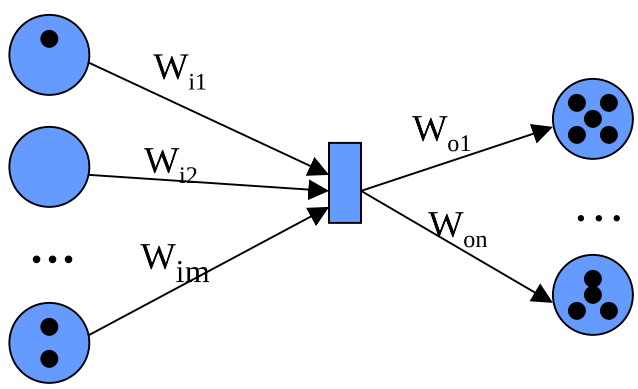
What is a Petri Net (Cont'd)

- **Marking (M)**
  - An m-vector $(k_0,k_1,\ldots,k_m)$
    - m: the number of places
    - $k_i >= 0$: the number of "tokens" in place p_i
- **Modeling**
  - Places ←→ Input/output data
  - Transitions ←→ Computation
  - Arcs (E)
    - Place →Transition: consume input data
    - Transition →place: produce output data

What is a Petri Net (Cont'd)

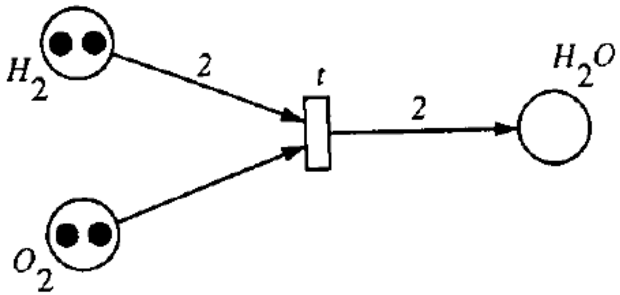
- **Firing rules**
  - An enabled transition
  - An enabled transition may or may not fire
  - A firing of an enabled transition

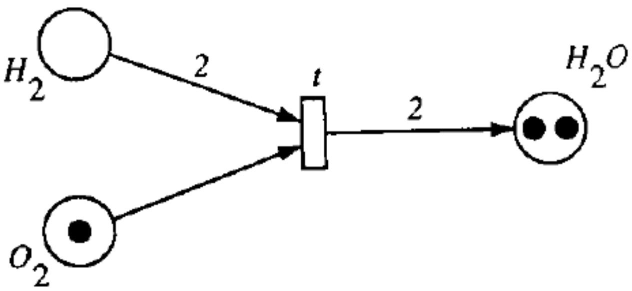




An Example

- $2H_2 + O_2 \rightarrow 2H_2O$

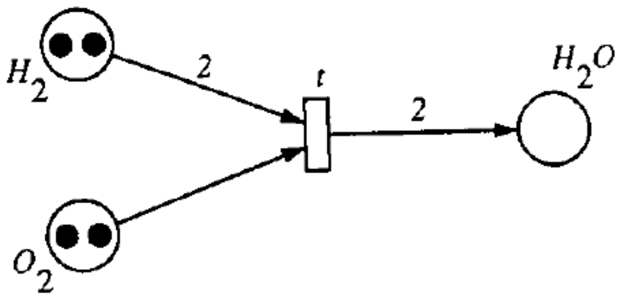


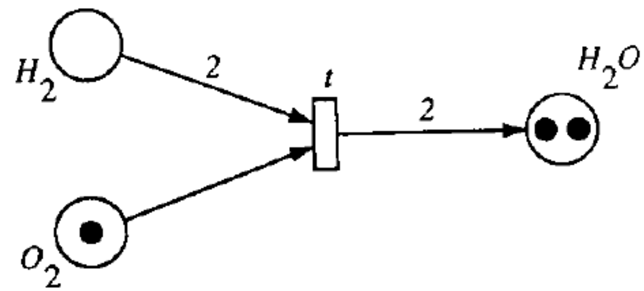


Formal Definition of Petri Net

- **PN = (P, T, F, W, M_0)**
  - P = {p_0,p_1,…,p_m}: a finite set of places
  - T = {t_1,t_2,…,t_n}: a finite set of transitions
  - $F \subseteq (P \times T) \bigcup (T \times P)$: a set of arcs (flow relation)
  - W: F→{1,2,3,…} weight function
  - M_0: P→{0,1,2,…} initial marking
  - $P \bigcap T = \varnothing \qquad P \bigcup T \neq \varnothing$

Formal Definition of Petri Net (cont'd)

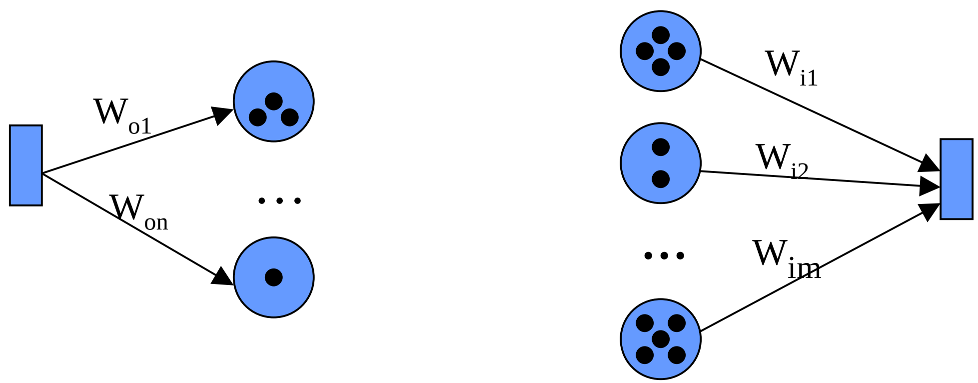




Places:
Transitions:
Arcs:
Weight:
Initial marking:

Formal Definition of Petri Net (Cont'd)

- **Source transition and sink transition**



- **Pure petri net and ordinary petri net**
  - Pure petri net: no self loop
  - Ordinary petri net: all the weights are 1's.
- **Infinite/finite capacity petri net**
  - K(p)

Formal Definition of Petri Net (Cont'd)

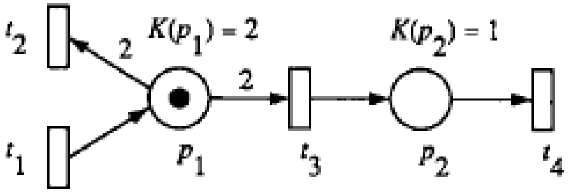
- **Strict/weak transition rule**
  - Strict: $K(p) < \infty$
  - Weak: $K(p) = \infty$
- **Theorem:**
  - For any pure finite-capacity net (N,M_0) with a strict transition rule, there must be another equivalent infinite-capacity net (N',M'_0) with a weak transition rule.

Equivalence

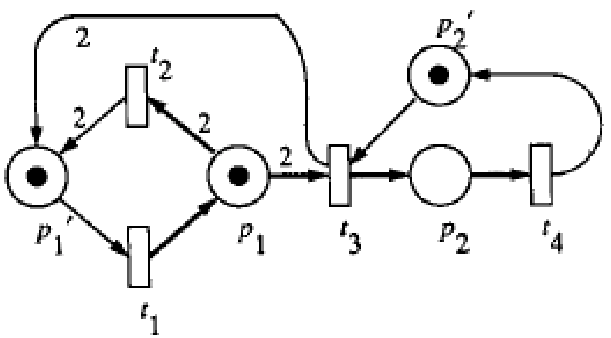
- **Reachability Graph G=(V,E)**
  - V: markings
  - E: firings

- **Equivalence**
  - Two petri nets (N,M_0) and (N',M'_0) are equivalent iff for any possible firing sequence in (N,M_0) same firing sequence can be found in (N',M'_0) and vice versa.

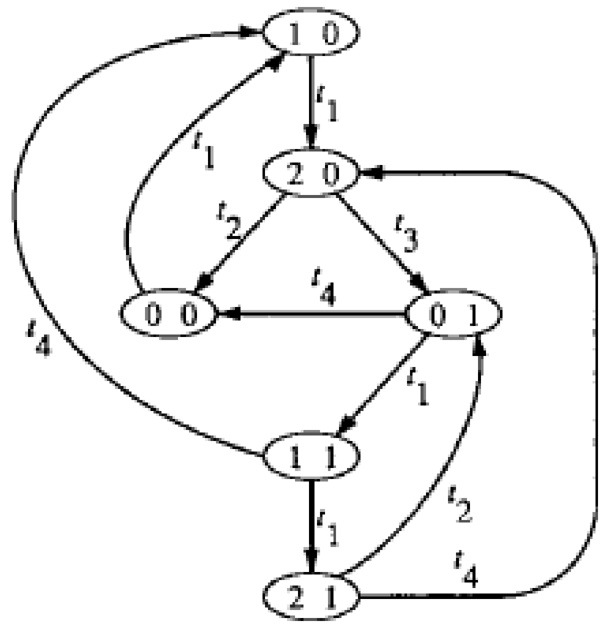
An Example



(a)

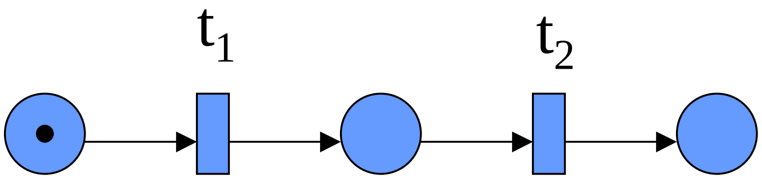


(b)

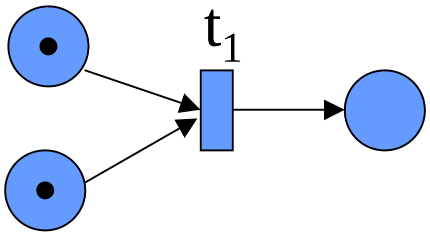


(c)

Parallel Activity

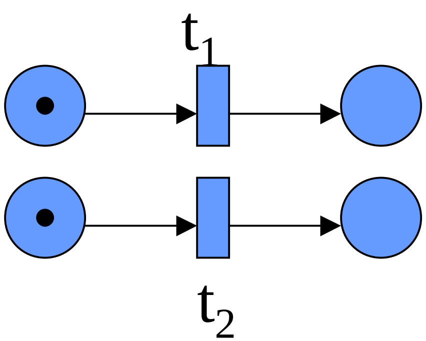


Sequencing

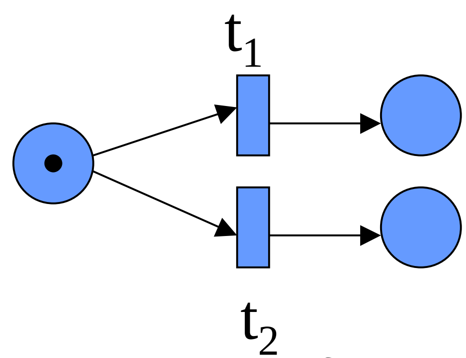


Synchronization

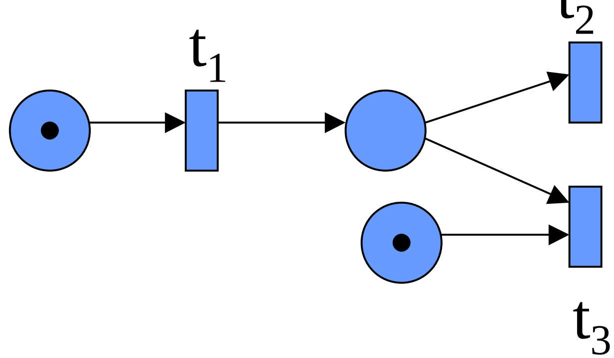
Parallel Activity (Cont'd)



Concurrence



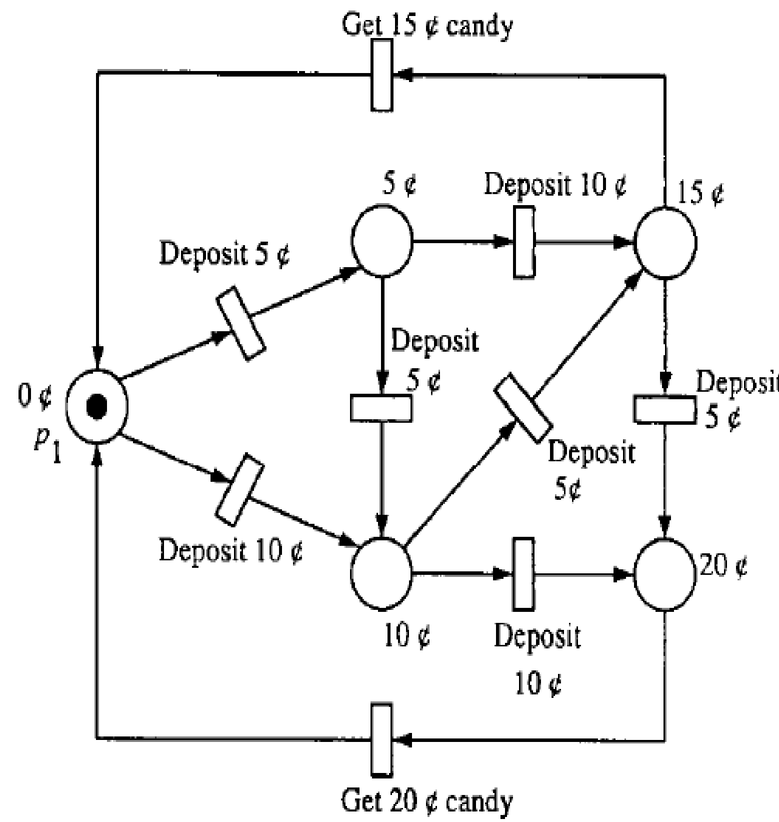
Choice (conflict)



Confusion (conflict + currency)

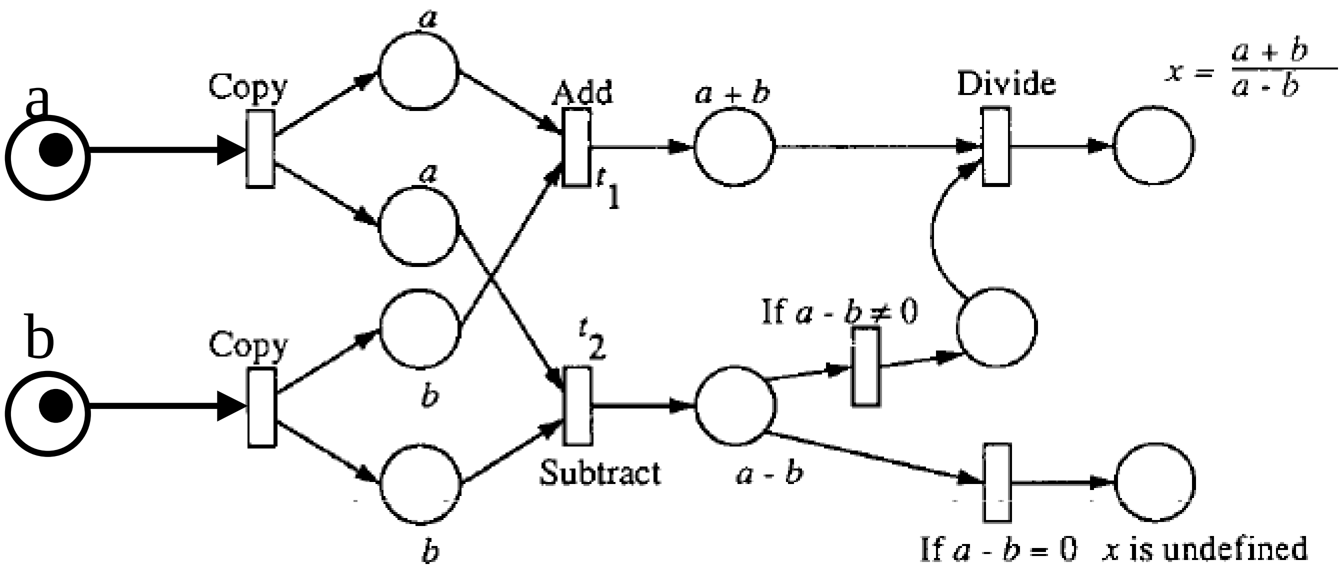
Finite State Machine

- A vender machine
  - accepts 5, 10 cents
  - sell candy bars worth 15 or 20 cents
  - maximum hold up coins = 20 cents

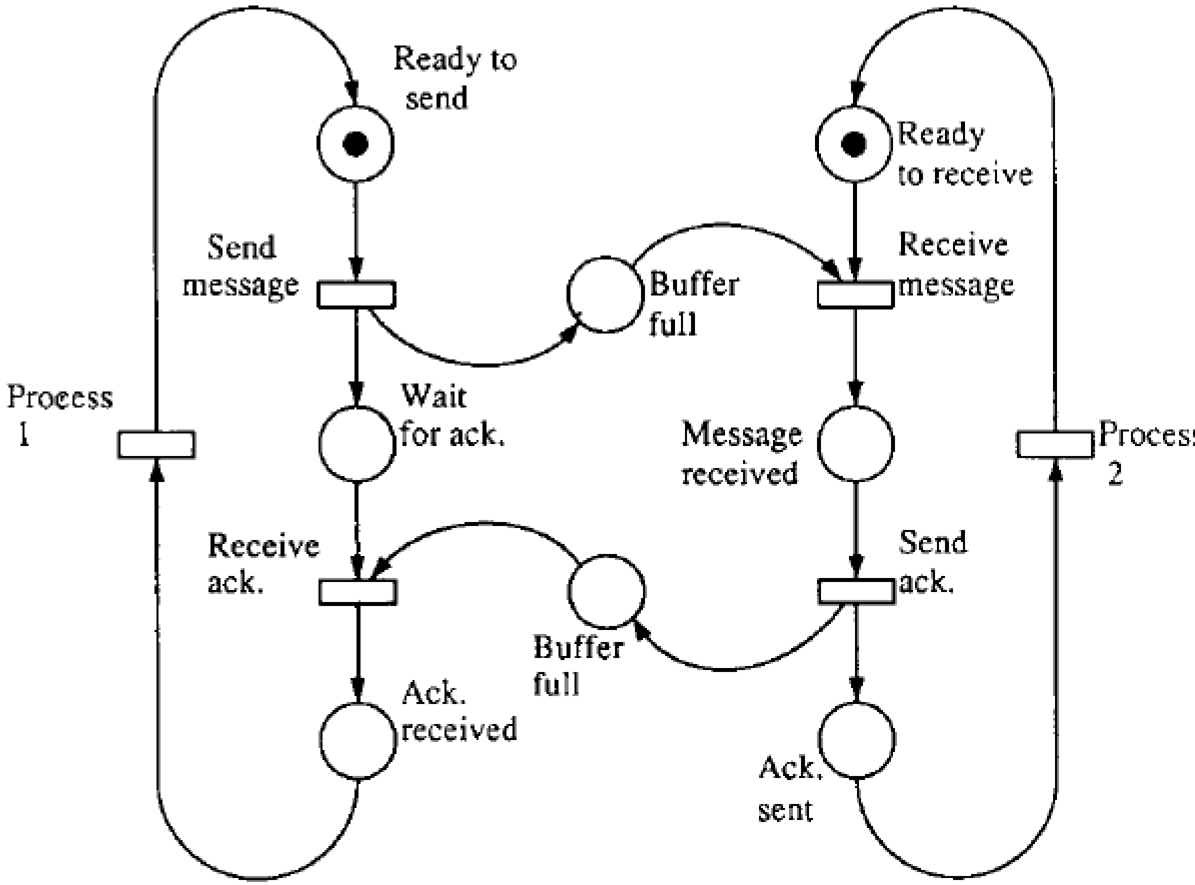


Data Flow Graph

- $X=(a+b)/(a-b)$

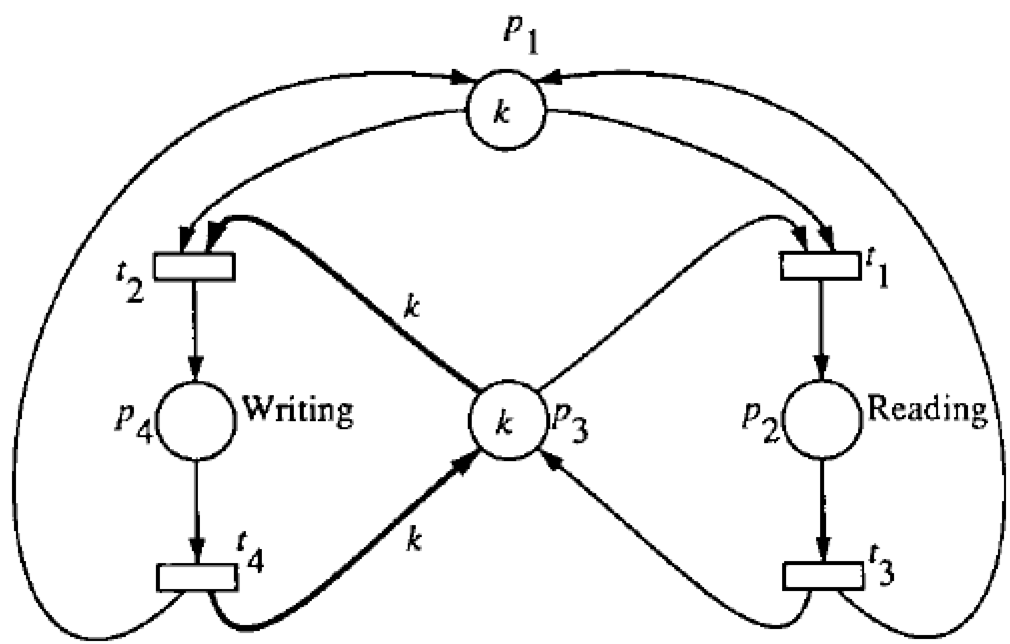


Communication Protocol



Synchronization Control

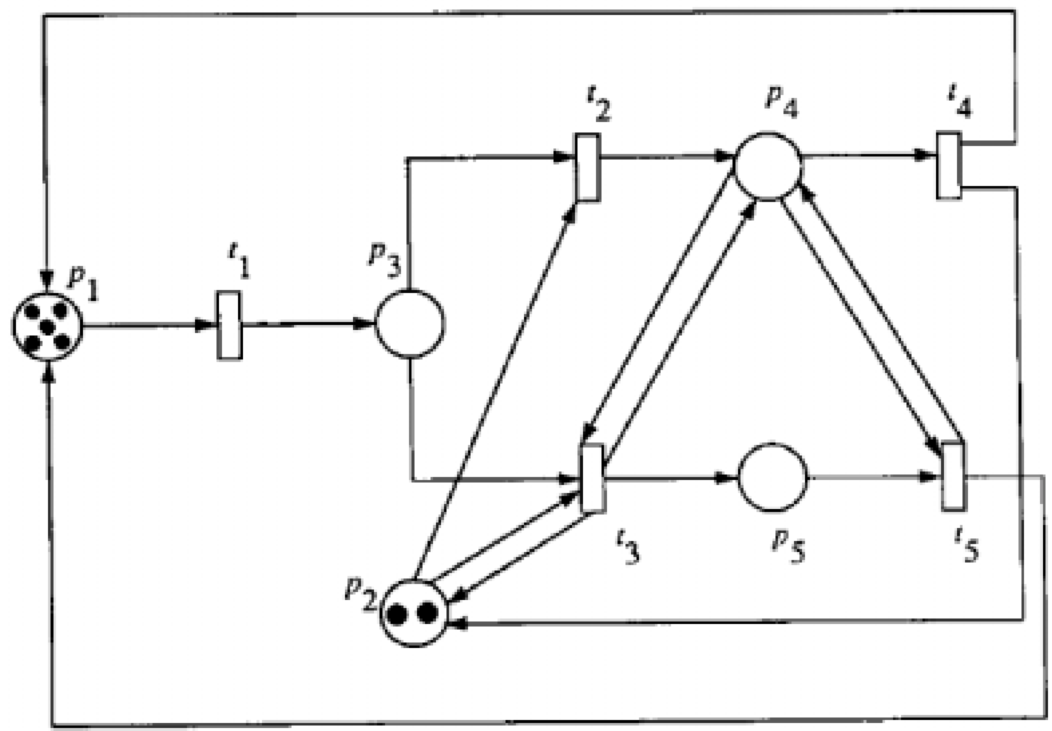
- k processes
- More than two can read simultaneously
- One is writing, no one can read
- One is reading, no one can write



Multiprocessor Systems

- 5 processors
- 2 buses
- 3 memories

P1: running processes
P2: available buses
P3: access requests
P4: access to the shared memory
P5: processors requesting the same shared memory with that in P4



Properties

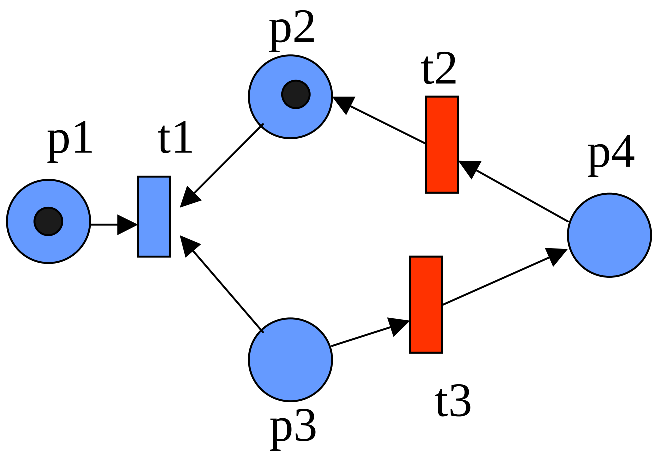
- Behavioral Properties
  - Properties hold only for the given initial marking
- Structural Properties
  - Independent of the initial marking
  - Depend on the topological structure of the nets

Behavioral Properties

- Reachability
- Boundedness
- Liveness
- Reversibility
- Coverability
- Persistence
- Synchronic distance
- Fairness

Reachability

- A Marking M_n is **reachable** from marking M_0 if there exists a sequence of firings $\sigma = t_1\ t_2 \dots t_n$ that transforms M_0 to M_n.



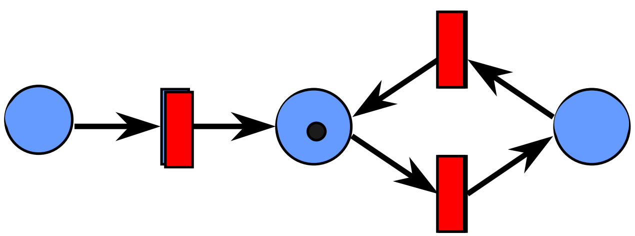
$M_0 = (1,0,1,0) \xrightarrow{?} M_n = (1,1,0,0)$

$t_3 \quad t_2$

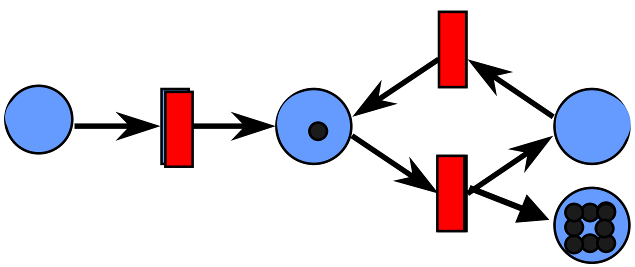
- $R(M_0)$: all the reachable markings

Boundedness

- $M(p_i) <= K$



Bounded

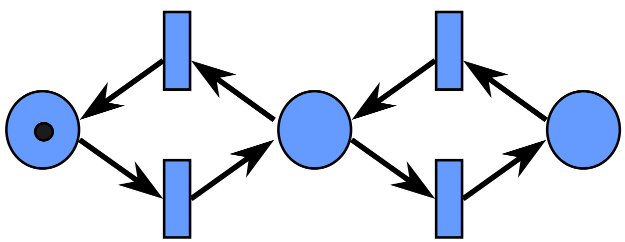


Unbounded

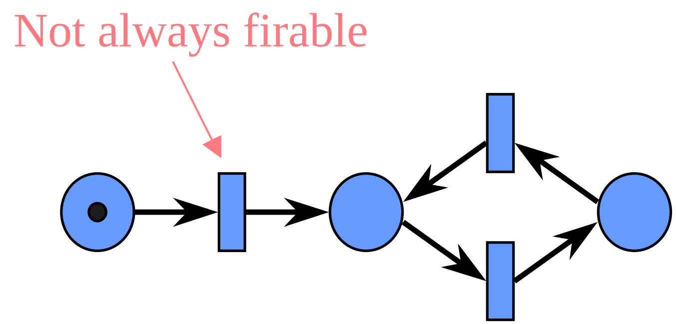
- **The net is *safe* if 1-bounded**

Liveness

- **From any reachable marking any transition can become fireable**



Live Petri Net

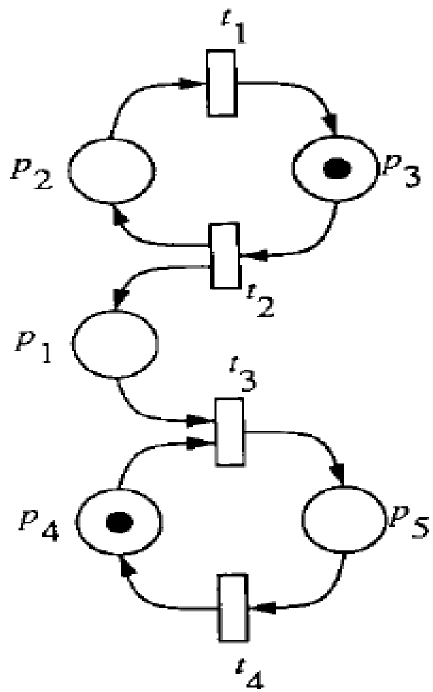


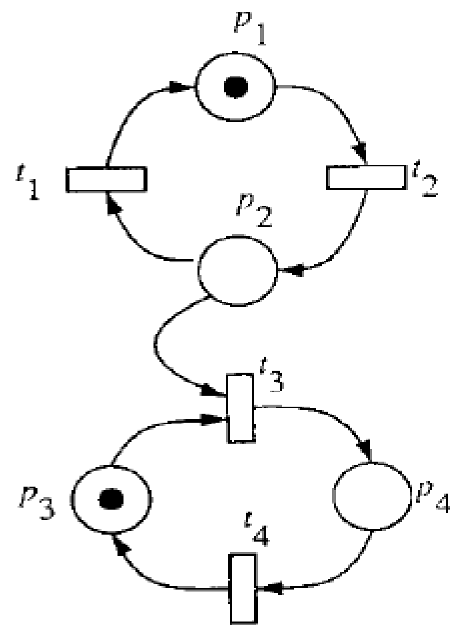
Nonlive Petri Net

- **Different levels of live: dead, L1-live, L2-live, L3-live, L4-live**

Reversibility

- For any M_k in $R(M_n)$, M_n is also in $R(M_k)$
  - M_n: *home state*



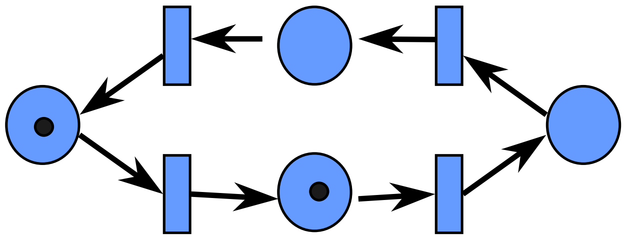


Coverability

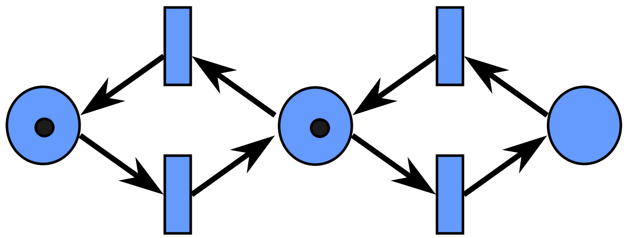
- Exist marking M’ in $R(M_0)$ such that $M'(p_i) >= M(p_i)$
  - Closely related to liveness
    - E.g. Let M be the minimum marking needed to enable a transition t, then
    - t is dead ←→ M is not coverable
      t can never be fired since no enough tokens are available
    - t is L1-live ←→ M is coverable
      t can be at least fired once since M’ is reachable and can offer enough tokens

Persistence

- **For any pair of enabled transitions, firing one will not disable another**



Persistent

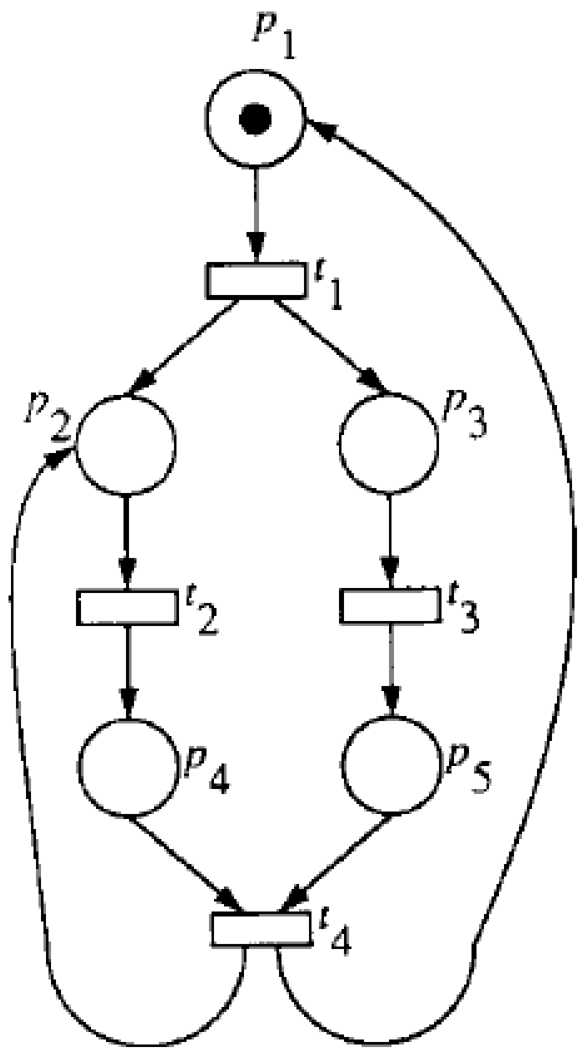


Not persistent

- **All the *marked graph* are persistent but *not* vice versa**
    - **Marked graph: each place has single input and single output**

Persistence (Cont'd)

Persistent, but not marked graph !

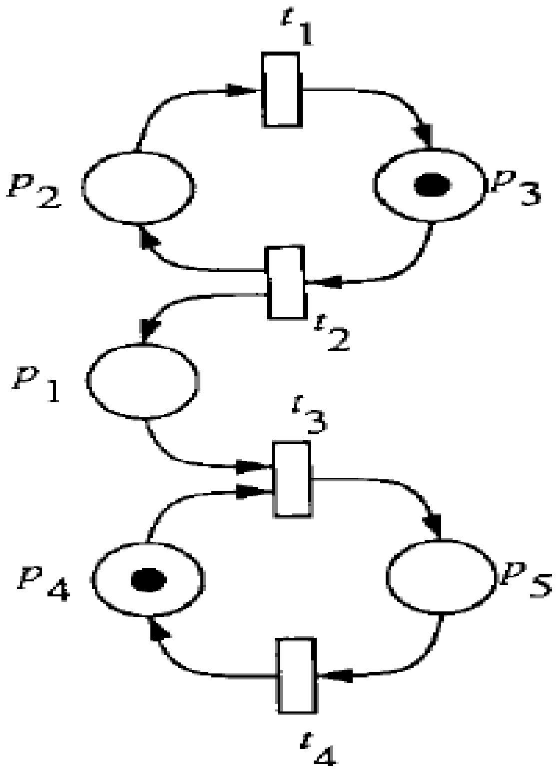


Synchronic Distance

- **The maximum possible difference between the numbers of times that two transitions fired**
  - **$d_{12} = \max_{\sigma} \mid T(t1) - T(t2) \mid$**

    σ: a firing sequence starting from any marking M_n in $R(M_0)$

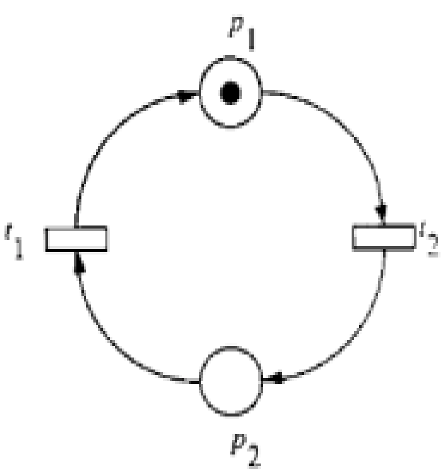
    T(t): the number of times that t is fired in σ

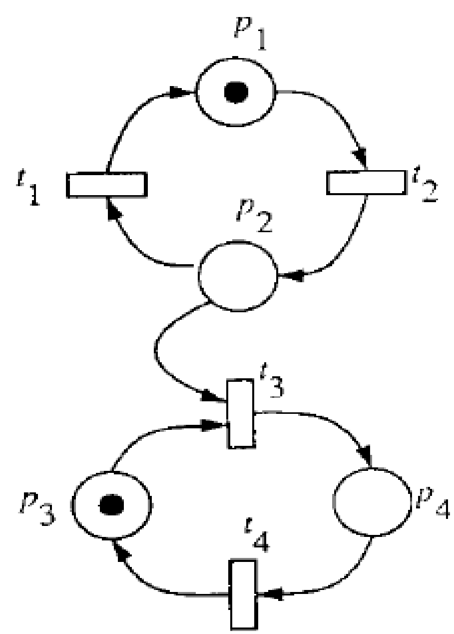


d12 = 1
d34 = 1
d13 = ∞

Fairness

- **Unconditional fairness**
  - **A firing sequence is unconditional fair if every transition in the net can appear infinitely often.**
  - **(N,M_0) is an unconditionally fair net if every firing sequence starting from M in $R(M_0)$ is unconditionally fair.**





Properties

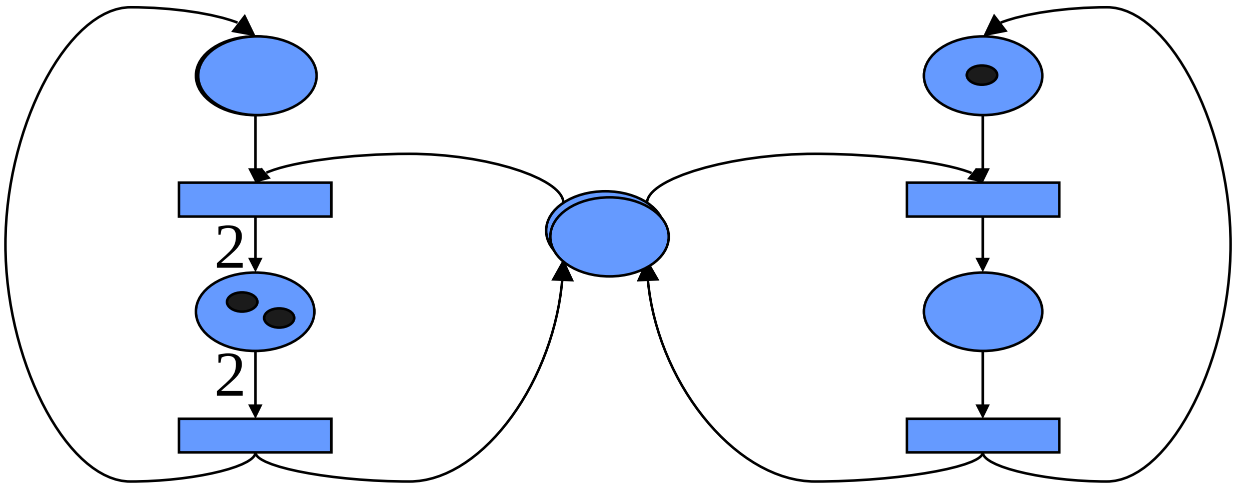
- Behavioral Properties
  - Properties hold only for the given initial marking
- Structural Properties
  - Independent of the initial marking
  - Depend on the topological structure of the nets

Structural Properties

- Structurally live
  - There exists a live initial marking for N
- Controllability
  - Any marking is reachable for any other marking
- Structural Boundedness
  - Bounded for any finite initial marking

Structural Properties (Cont'd)

- Conservativeness
  - The total number of the tokens in the net is a constant.



Structural Properties

- Repetitiveness
  - There exists a initial marking M_0 and a firing sequence from M_0 such that each transition can occur infinitely often.
- Consistency
  - There exists a initial marking M_0 and a firing sequence from M_0 back to M_0 such that each transition occurs at least once.

Petri Extensions

- Timed petri net
  - Introduce time delays associated with transitions and/or places
    - Deterministic net
      - Delays are determined
    - Stochastic net
      - Delays are probabilistic
- Colored petri net
  - Tokens have different values (colors)

Summary

- Graphical interpretation
- Convenient for represent distributed, concurrency, synchronization, etc
- Properties
  - Behavioral
  - Structural
- Extensions