Redes de Petri

Algumas semelhanças com a abordagem baseada em automata;

Pode representar uma classe mais ampla de SED's;

Poucos resultados relativos à síntese sistemática de controladores;

Há modelos temporizados e não-temporizados evento = *transição*

informações sobre as condições para ocorrência de um evento = *lugares*

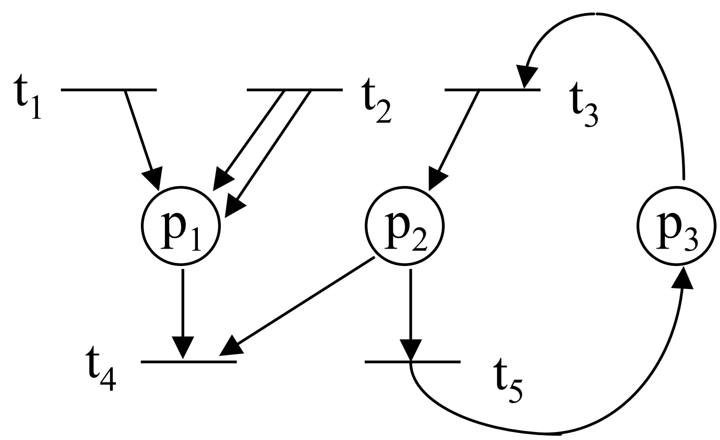
Definição: Uma rede de Petri é uma quádrupla (P, T, A, w), onde:
P é um conjunto finito de lugares;
T é um conjunto finito de transições;
A é um conjunto de arcos, sub-conjunto do conjunto $(P \times T) \cup (T \times P)$;
w é uma função-peso $w: A \to \mathbb{N}$

Em geral: $P = \{p_1, ..., p_n\}$
$T = \{t_1, ..., t_m\}$
arcos: (p_i, t_j) ou (t_j, p_i)

A condição de conjuntos finitos pode ser relaxada, admitindo-se conjuntos contáveis.

Representação gráfica através de grafos com dois tipos de nós: lugares e transições

Exemplo:



Lugares de saída da transição t_j:
$O(t_j) = \{p_i: (t_j, p_i) \in A\}$

Lugares de entrada da transição t_j:
$I(t_j) = \{p_i: (p_i, t_j) \in A\}$

Analogamente para os lugares.

Definição: Uma marcação de uma rede de Petri é uma função $x: P \rightarrow N$

Uma marcação em geral é representada por um vetor $x = [x(p_1) \ ... \ x(p_n)]$;

No grafo, a marcação é representada por "fichas" dentro dos lugares.

Definição: Uma rede de Petri marcada é uma quíntupla (P, T, A, w, x_o) onde: (P, T, A, w) é uma rede de Petri e x_o é uma marcação inical.

Definição: O estado de uma rede de Petri marcada é sua marcação $[x(p_1) \ ... \ x(p_n)]$.

Obs.: O espaço de estados, X é em geral infinito: $X = N^n$.

Dinâmica das Redes de Petri

Representação da dinâmica = movimento das fichas, quando os eventos (transições) ocorrem.

Definição: Uma transição $t_j \in T$ numa rede de Petri marcada é dita habilitada se:

$$x(p_i) \geq w(p_i, t_j) \quad \text{para todo } p_i \in I(t_j).$$

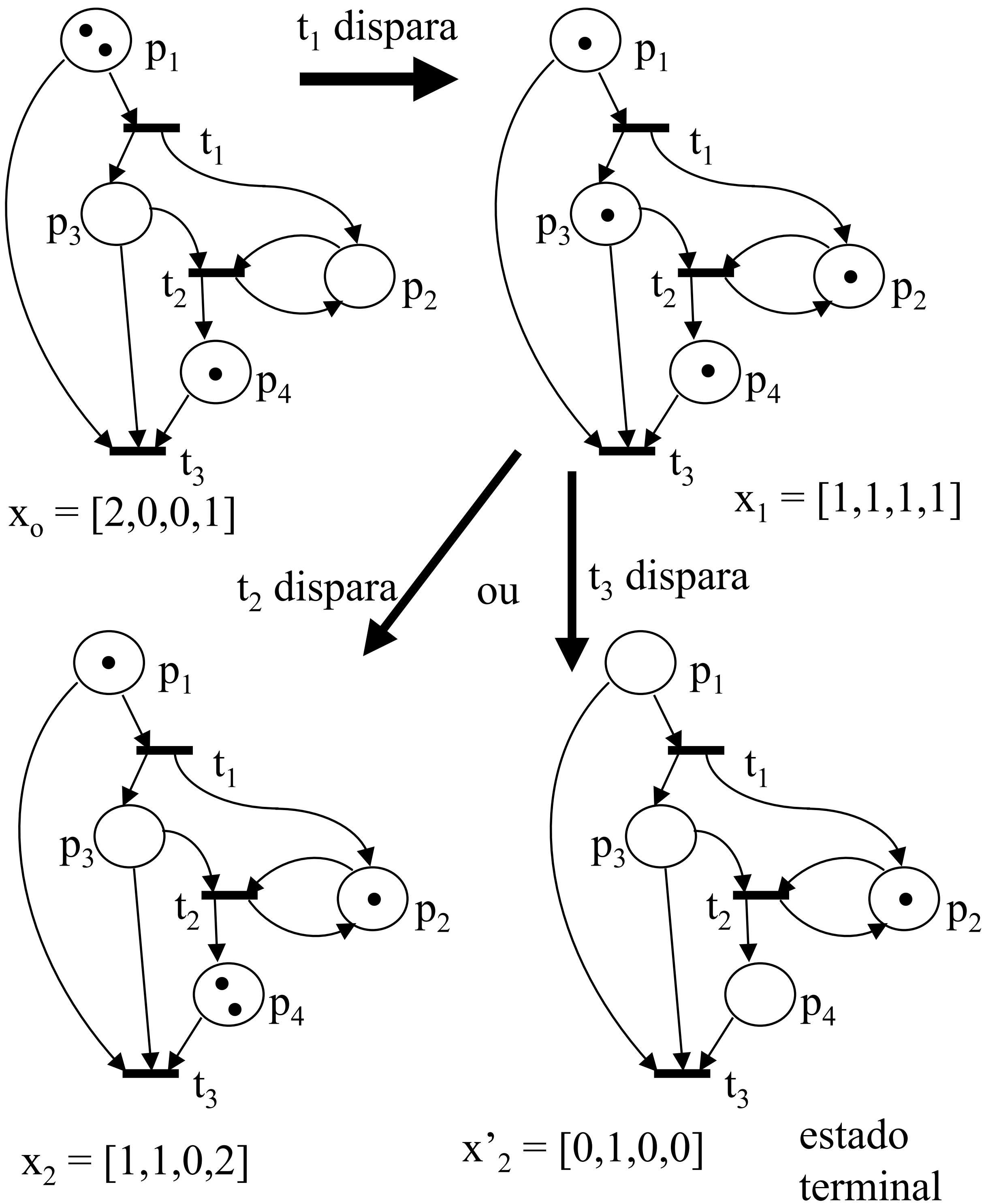
Definição: Uma função de transição de estados, $f: N^n \times T \to N^n$, de uma rede de Petri marcada é definida para uma transição $t_j \in T$ se e somente se esta transição está habilitada.

Se $f(x, t_j)$ é definida, diz-se que $x' = f(x, t_j)$, onde:

$$x'(p_i) = x(p_i) - w(p_i, t_j) + w(t_j, p_i)$$
$$i = 1, ..., n$$

Obs.: O número de fichas não se conserva necessariamente.

Exemplo:



p1
t1 dispara
p1
t1
t1
p3
p3
t2
p2
t2
p2
p4
p4
t3
t3
xo = [2,0,0,1]
x1 = [1,1,1,1]
t2 dispara
ou
t3 dispara
p1
p1
t1
t1
p3
p3
t2
p2
t2
p2
p4
p4
t3
t3
x2 = [1,1,0,2]
x'2 = [0,1,0,0]
estado
terminal

Em geral, a dinâmica de uma RP pode ser representada da seguinte forma:

Sejam:
$x = [x(p_1), x(p_2), ..., x(p_n)]$: o atual estado;
$x' = [x'(p_1), x'(p_2), ..., x'(p_n)]$: o próximo estado, após o disparo da j-ésima transição;

Definindo: $u = [0, 0, ...,0 , \underset{\substack{\uparrow \\ \text{j-ésima posição}}}{1}, 0, ...,0]$

Então:

$$\boxed{x' = x + uA}$$

onde: $A = [a_{ji}]$; $a_{ji} = w(t_j, p_i) - w(p_i, t_j)$

A é chamada matriz de incidência

No exemplo anterior:

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

A equação correspondente à primeira transição é:

$$[2\ 0\ 0\ 1] + [1\ 0\ 0].A = [1\ 1\ 1\ 1]$$

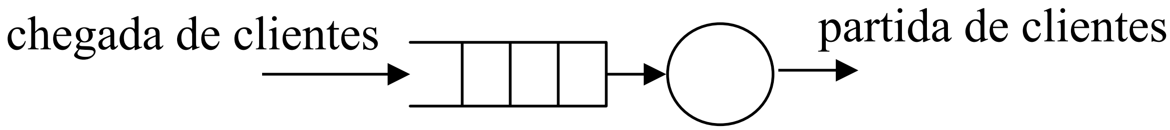
Para a segunda transição:

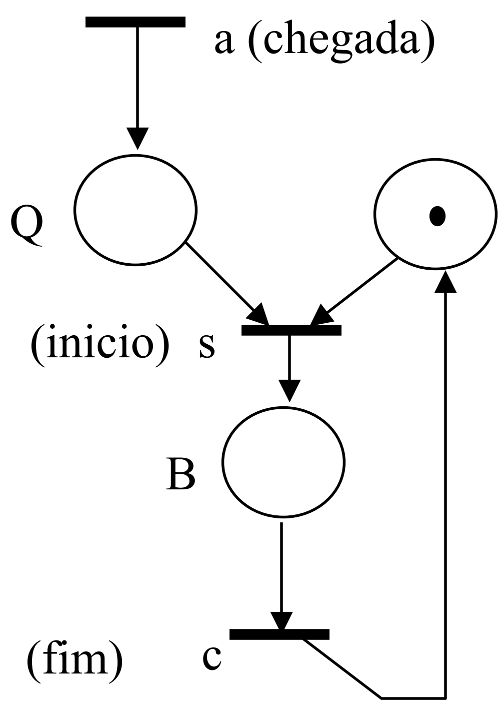
$$[1\ 1\ 1\ 1] = [0\ 1\ 0].A = [1\ 1\ 0\ 2]$$

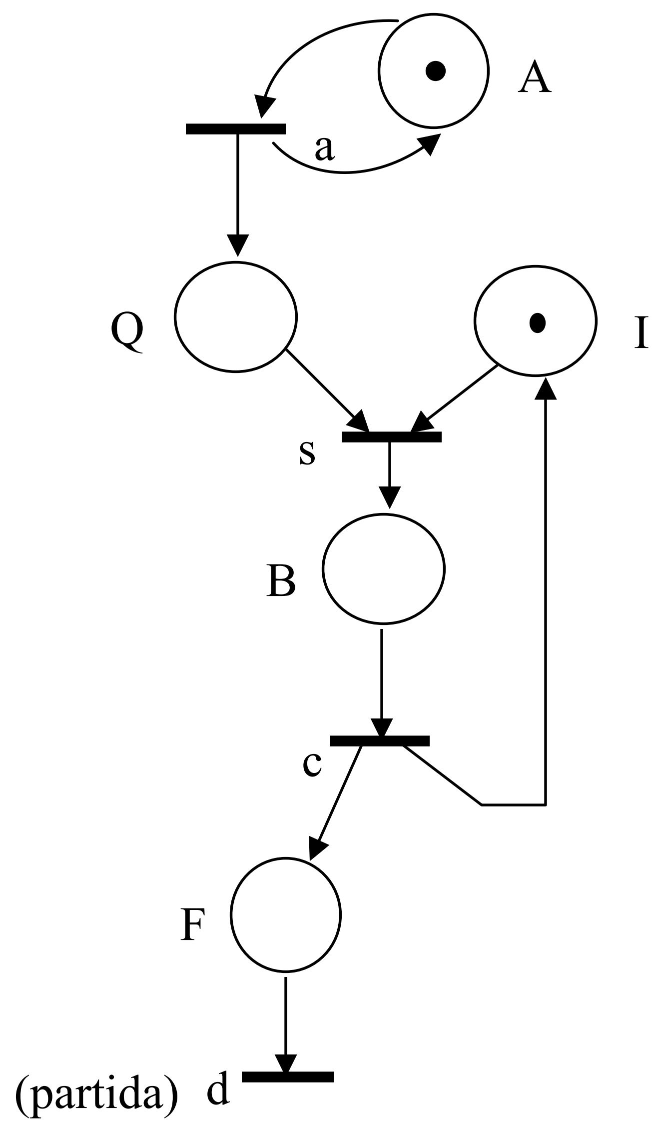
De uma maneira geral pode-se descrever a trajetória de um RP através da equação recursiva:

$$x_{k+1} = f(x_k, t^k) = x_k + u_k.A$$

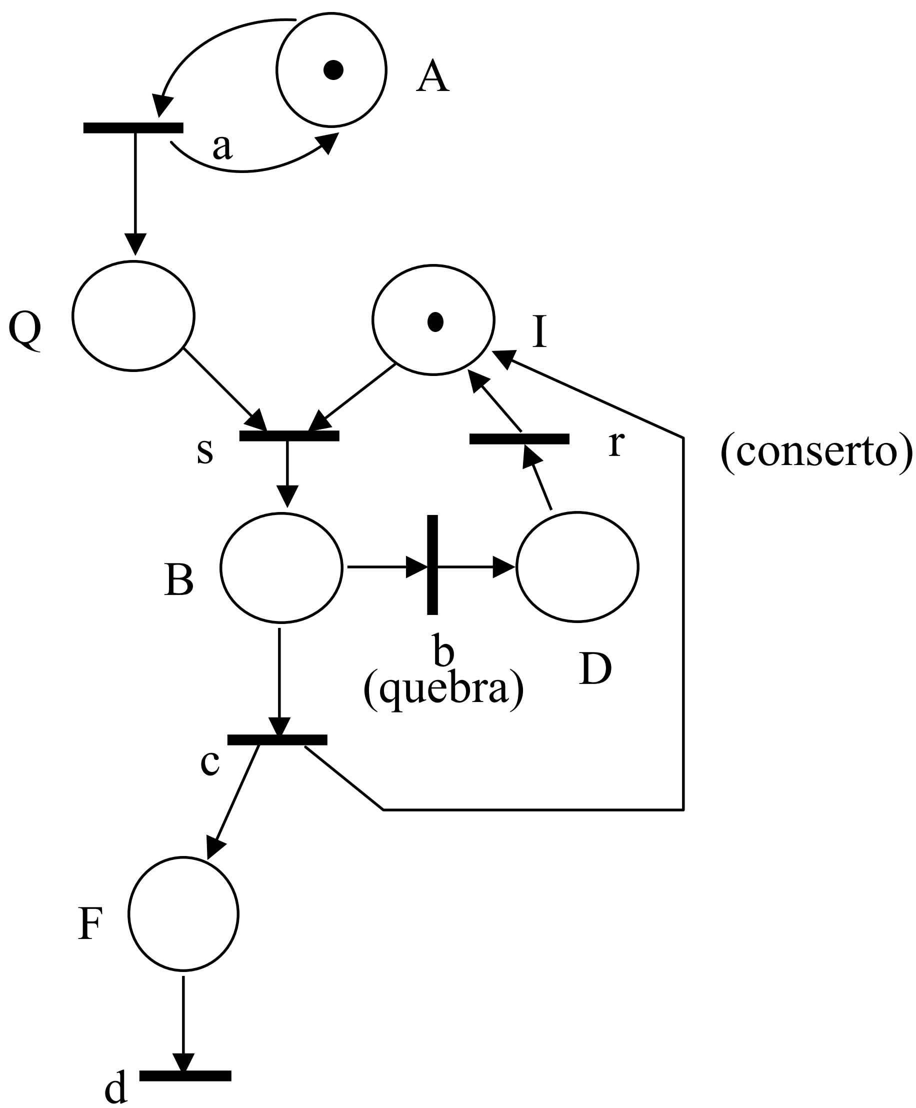
Modelos para uma fila:







Modelo que considera quebras no servidor



Problemas de Análise em RP

Os problemas a seguir, concernem qualquer modelo para SED's

Em geral, as definições buscam caracterizar propriedades "desejáveis" ou "indesejáveis" nos sistemas em estudo.

Limitação: Um lugar $p_i \in P$ numa Rede de Petri com uma cond. inicial x_o é dito *k-limitado* ou *k-seguro*, se $x(p_I) \leq k$ para qualquer estado em qualquer trajetória possível.

- Um lugar 1-seguro é dito seguro
- Um lugar k-limitado é dito limitado
- Se numa RP todos os lugares são limitado então a rede é dita limitada.

Conservação: Uma RP com um dado estado inicial x_o é dita ser conservativa em relação a um vetor $\gamma = [\gamma_1, \gamma_2, ..., \gamma_n]$ se:

$$\sum_{i=1}^{n} \gamma_i x(p_i) = const.$$

para qualquer estado em qualquer trajetória possível.

Em geral, numa RP o número de fichas não se conserva, entretanto a definição acima permite a análise de situações em que algo deve ser conservado.

Uma rede conservativa pode representar um sistema no qual recursos não são criados nem destruídos.

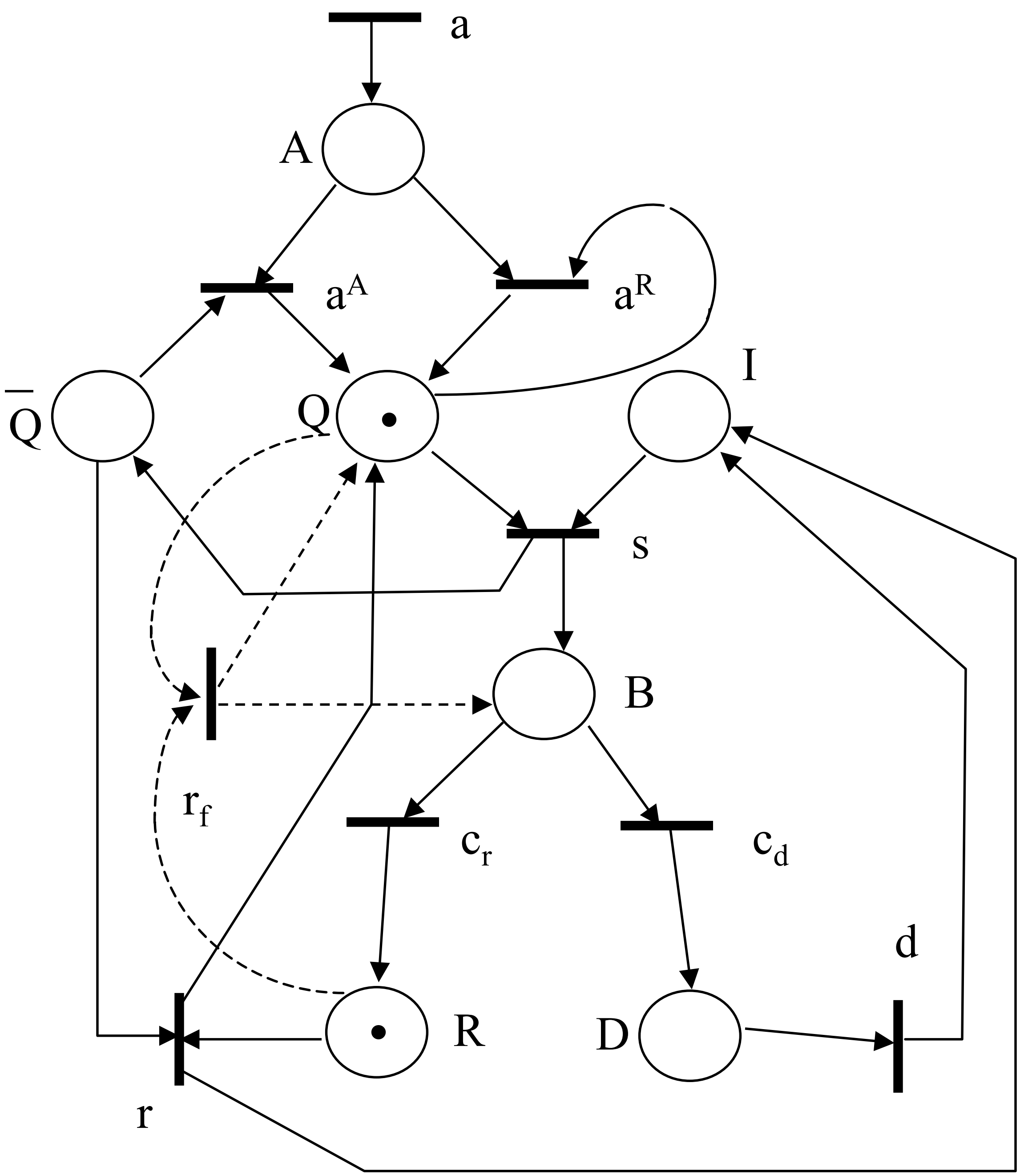
Vivacidade e bloqueio (deadlock):
São características opostas, de grande importância na análise de um sistema.

Uma RP com um dado estado inicial x_o é dita *viva* se, a partir de qualquer estado alcançado a partir de x_o, existir alguma trajetória na qual uma transição qualquer possa disparar.

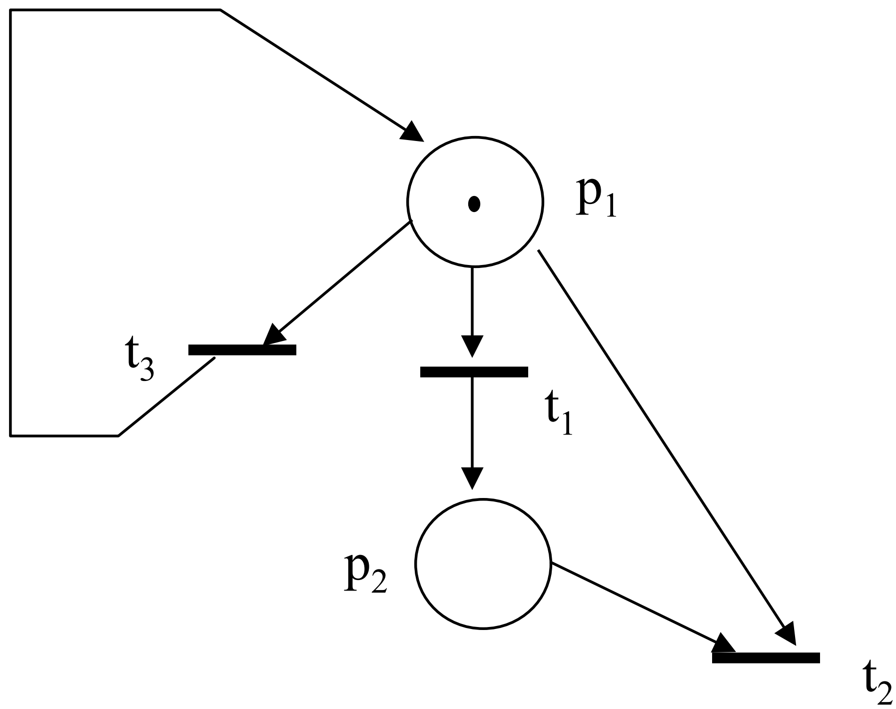
Esta definição é de difícil verificação, levando a distinções relativas a uma transição dada:

- **L0-viva ou morta**, se a transição nunca dispará a partir do estado inicial;
- **L1-viva**, se existir alguma sequência de disparos tal que a transição possa disparar pelo menos uma vez;
- **L2-viva** se a transição pode disparar pelo menos k- vezes para algum número positivo k;
- **L3-viva** se existir alguma sequência infinita de disparos na qual a transição aparece infinitas vezes
- **L4-viva ou viva** se a transição for L1-viva para qualquer estado alcançado a partir do estado inicial.

Exemplo: Sistema de fila com retraballho



Exemplo: Níveis de vivacidade



t_2 - morta
t_1 - L1-viva
t_3 - L3-viva mas não L4-viva, pois pode tornar-se morta no estado resultante de um disparo de t_1.

Alcançabilidade: Um estado x numa RP é dito alcançável a partir de um estado x_o se existir uma sequência de transições iniciando-se em x_o e tal que o estado pode se tornar x.

"Coverabilidade": Dada uma RP com estado inicial x_o, um estado y pode ser coberto se existir uma sequência de transições inicando-se em x_o e tal que o estado pode se tornar x e $x(p_i) \geq y(p_i)$, para todo i = 1, ..., n.

Diz-se que o estado x cobre o estado y.
Se y é o estado que garante minimamente o disparo de uma transição t_j, então, se y não puder ser coberto a partir do estado corrente, pode-se afirmar que t_j está morta.

(conceito muito próximo ao de L1-vivacidade)

Persistência: Uma RP é dita persistente se, para cada duas transições habilitadas, o disparo de uma delas não desabilita a outra.

No exemplo anterior, se t_1 ocorre, t_3 é desabilitada, sendo que ambas estavam habilitadas

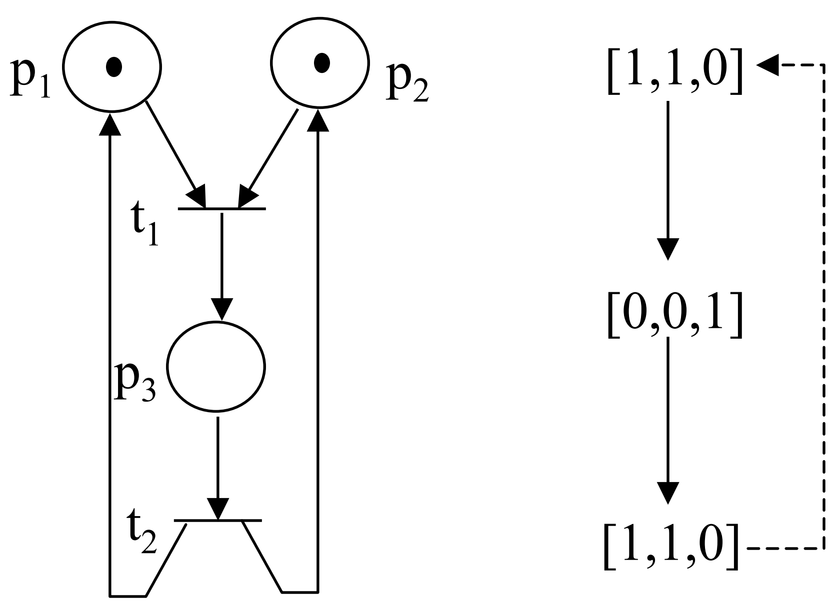
Em RP's temporizadas, em que uma transição habilitada só é disparada após um determinado atraso, este conceito é equivalente ao de não-interruptibilidade. A interruptibilidade permite uma analogia com sistemas não-lineares.

Em RP's também é possível estabelecer o problema de reconhecimento de linguagens, onde uma sequência de transições pode ou não acontecer numa determinada rede.

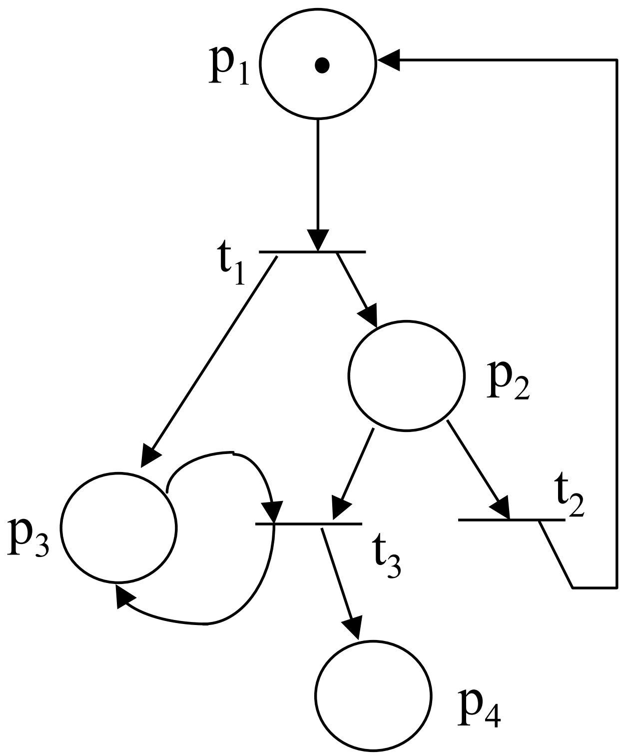
Árvore de Coverabilidade

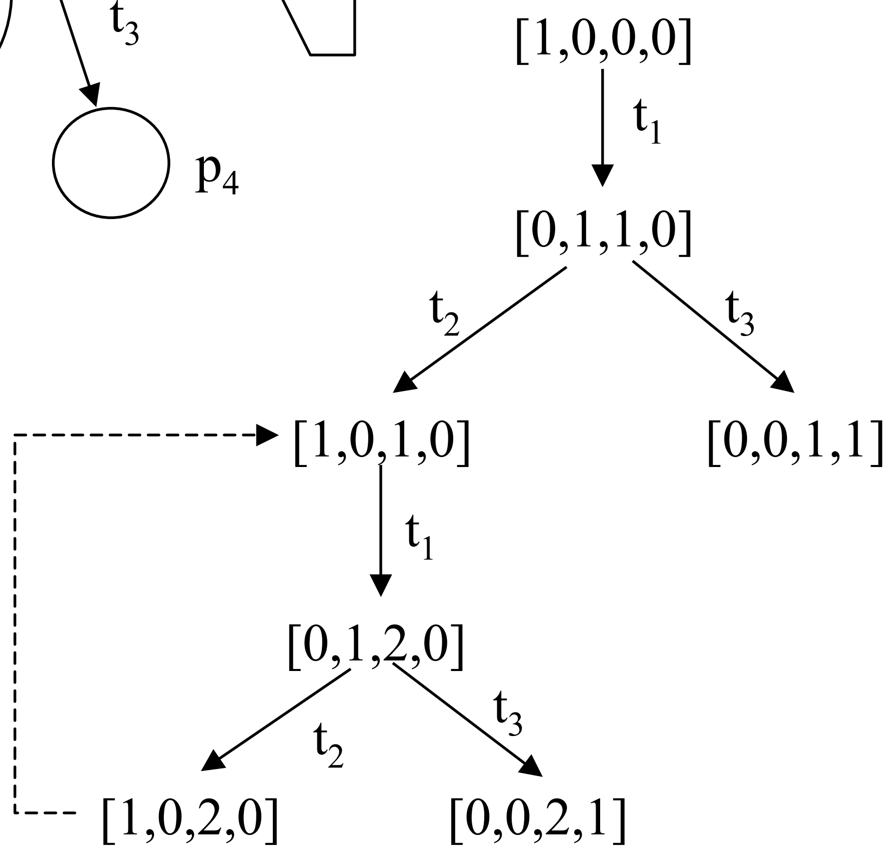
- Técnica de Análise
- Representação Finita
- Alguma perda de informação

Motivação: Exemplo



Exemplo: Motivação





Terminologia:

Nó Raiz: Corresponde à marcação inicial da rede;

Nó Terminal: Nó a partir do qual nenhuma transição pode ocorrer;

Nó Duplicado: Nó idêntico a um nó já presente na rede;

Dominância de Nós: Se **x** e **y** são dois estados de uma RP (nós da árvore), diz-se que **x** domina **y** (notação: $\mathbf{x} >_d \mathbf{y}$) se:
a) $x(p_i) \geq y(p_i)$ para todo i = 1, ..., n
b) $x(p_i) > y(p_i)$ para algum i = 1, ..., n;

Símbolo ω: Usado para indicar que uma marcação é ilimitada. Na árvore de coverabilidade é utilizado para identificar dominância de nós.

Algoritimo para Construção da Árvore de Coverabilidade

1) Inicializar com a marcação inicial (x_o);

2) Para cada novo nó construído, x, avaliar a função $f(x,t_j)$ para todo $t_j \in T$:

2.1) Se $f(x,t_j)$ é indefinida, então x é um nó terminal;

2.2) Se $f(x,t_j)$ é definida, para algum $t_j \in T$, criar um novo nó $x' = f(x,t_j)$:

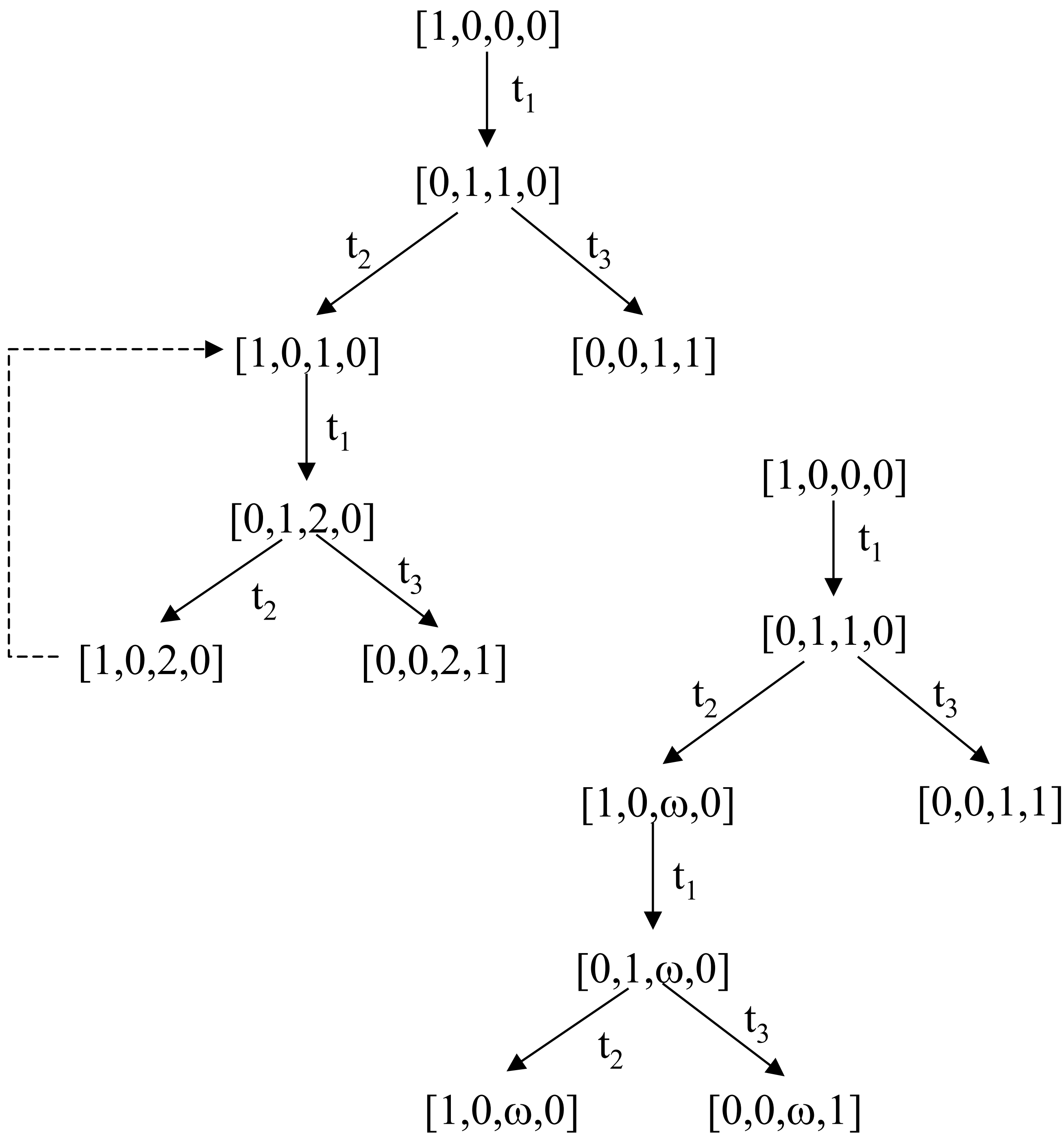
2.2.1) Se $x(p_i) = \omega$ para algum i, fazer $x'(p_i) = \omega$;

2.2.2) Se existir um nó y, no caminho de x_o até x (inclusive), tal que $x' >_d y$, fazer $x'(p_i) = \omega$ para todo p_i tal que $x'(p_i) > y(p_i)$;

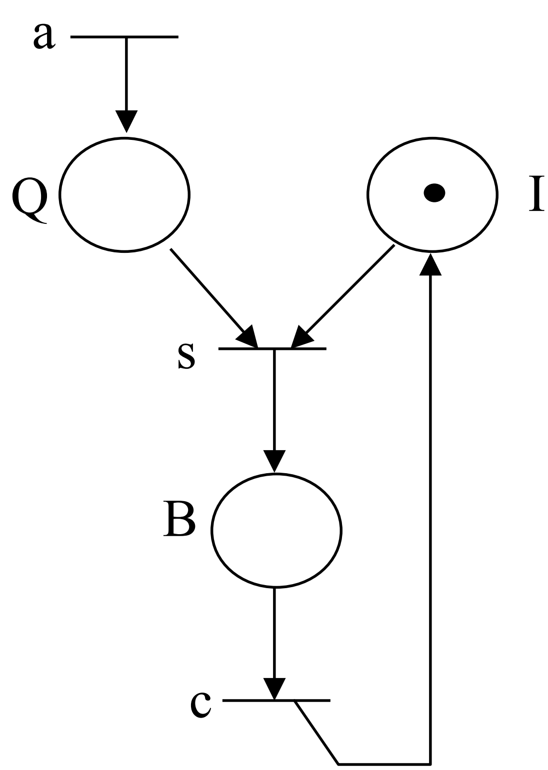
2.2.3) Senão, fazer $x' = f(x,t_j)$;

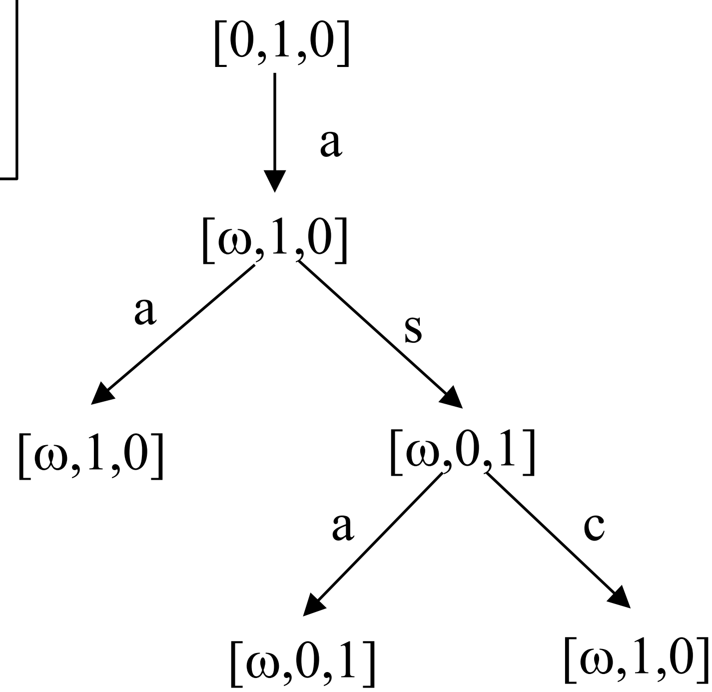
3) Se todos os nós forem duplicados ou terminais, parar.

Exemplo: Árvore de coverabilidade da Rede de Petri anterior:



Exemplo: Sistema de Fila:





Aplicações da Árvore de Coverabilidade

Problemas de Limitação:

Uma condição necessária e suficiente para que uma RP seja limitada é que o símbolo ω nunca apareça em sua árvore de coverabilidade.

Neste caso, a árvore de coverabilidade coincide com a árvore de alcançabilidade.

Além disso, o maior valor observado para $x(p_i)$ na árvore é um limitante para p_i.

Se a árvore de coverabilidade de uma RP só contem 0's e 1's então todos os seus lugares são seguros e a rede é segura.

Problemas de Conservação:

Uma RP é conservativa se existir um vetor $\gamma = [\gamma_1, ..., \gamma_n]$ tal que:

$$\sum_{i=1}^{n} \gamma_i x(p_i) = const.$$

Se $x(p_i) = \omega$ para algum nó da árvore, então, para que a RP seja conservativa, deve-se ter:

$$\gamma_i = 0$$

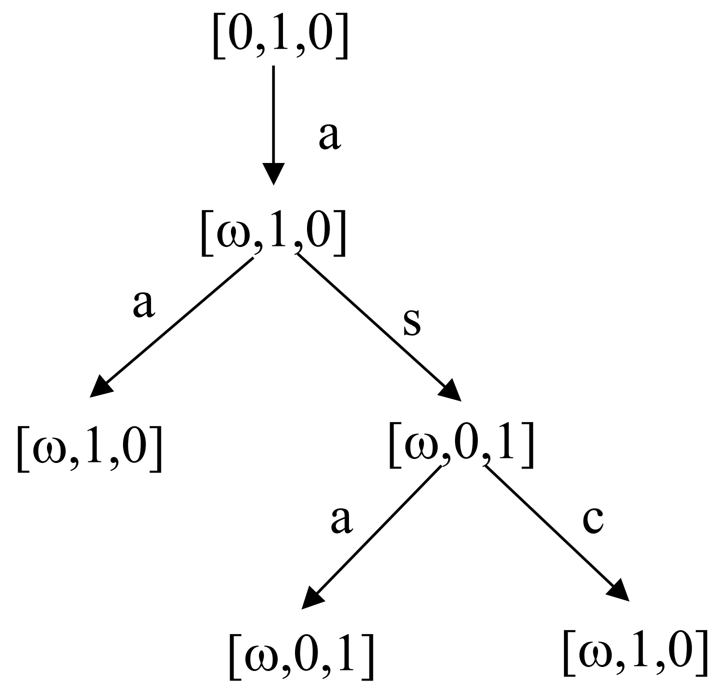
Seja $b \leq n$ o número de lugares limitados e r o número de nós da árvore de coverabilidade

Tem-se então um sistema de equações lineares com r equações e b+1 incógnitas da forma:

$$\sum_{i=1}^{n} \gamma_i x(p_i) = C$$

obs.: C também é uma incógnita.

Exemplo: Sistema de Filas



6 equações (na realidade 2)
3 incógnitas γ_2, γ_3 e C

$$\begin{cases} \gamma_2 \,.\, 1 + \gamma_3 \,.0 = C \\ \gamma_2 \,.\, 0 + \gamma_3 \,.1 = C \end{cases}$$

Solução: $\gamma_2, \gamma_3 = 1$ e $C = 1$

Em geral, a solução pode não ser única ou não existir.

Problemas de Coverabilidade:

Seja $y(p_i)$, $i = 1, ..., n$ um estado que se deseja cobrir.

Se existir um nó na árvore de coverabilidade tal que $x(p_i) \geq y(p_i)$, $i = 1, ..., n$, então o estado y é coberto por x

O caminho na árvore de x_o até x indica a sequência de transições necessárias para alcançar o estado x

Se o estado x contem o símbolo ω o caminho deve incluir um laço. Pode-se então determinar o número de vezes que se percorre o laço para cobrir y.

No exemplo anterior, o estado [3,1,0] pode ser coberto pois [ω,1,0] pertence à árvore.

Sequências de disparos para cobrir y:
{a,a,a} ou {a,s,a,c,s,a,c,a,a}

Limitações da Árvore de Coverabilidade

O símbolo ω representa um conjunto de valores alcançáveis por um lugar. Portanto, a menos que este símbolo não apareça (espaço de estados finito), alguns problemas não podem ser resolvidos utilizando-se a árvore de coverabilidade.

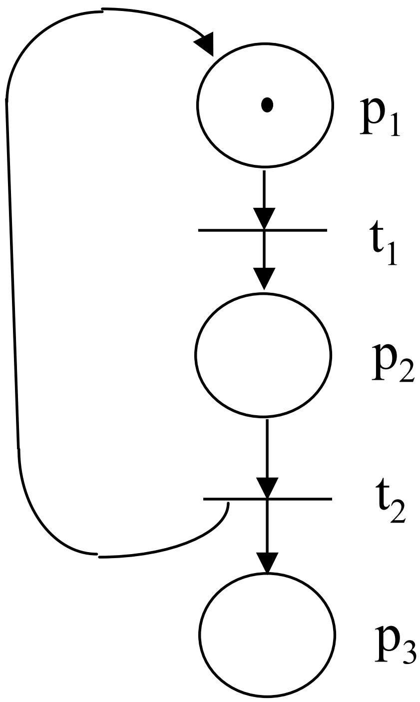
Alguns destes problemas:

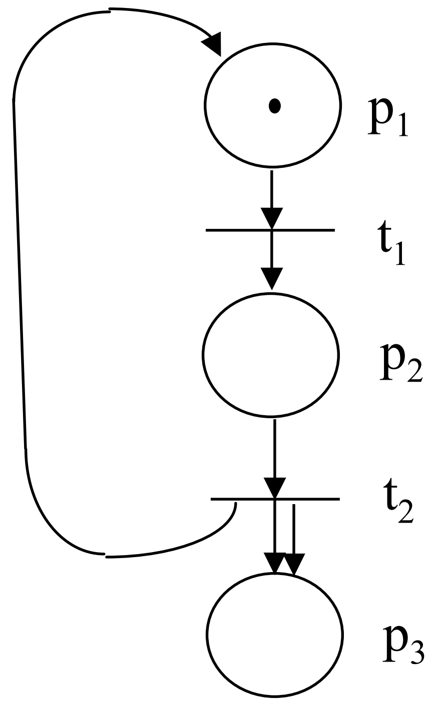
- impedimento de bloqueio;
- alcançabilidade de estados;
- reconhecimento de linguagens.

Alterntivas de Análise:

Por exemplo, uma ferramenta algébrica, como a equação de estado: $x' = x + uA$

Exemplo: Impossibilidade de análise da alcançabilidade de estados:





$$[1,0,0] \xrightarrow{t_1} [0,1,0] \xrightarrow{t_2} [1,0,\omega] \xrightarrow{t_1} [0,1,\omega] \xrightarrow{t_2} [1,0,\omega]$$

ω = **{1,2,3,...}**

$$[1,0,0] \xrightarrow{t_1} [0,1,0] \xrightarrow{t_2} [1,0,\omega] \xrightarrow{t_1} [0,1,\omega] \xrightarrow{t_2} [1,0,\omega]$$

ω = **{2,4,6,...}**

Comparação entre Geradores e Redes de Petri

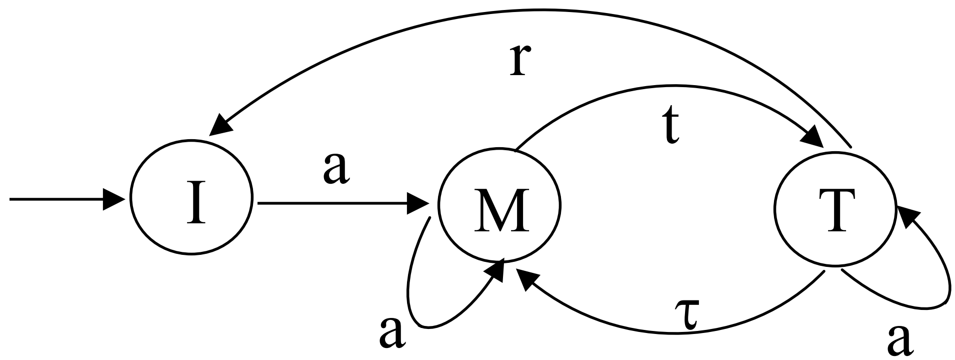
Não há "melhor" modelo; a escolha depende do problema tratado e de preferências pessoais. Contudo, alguma comparação é possível.

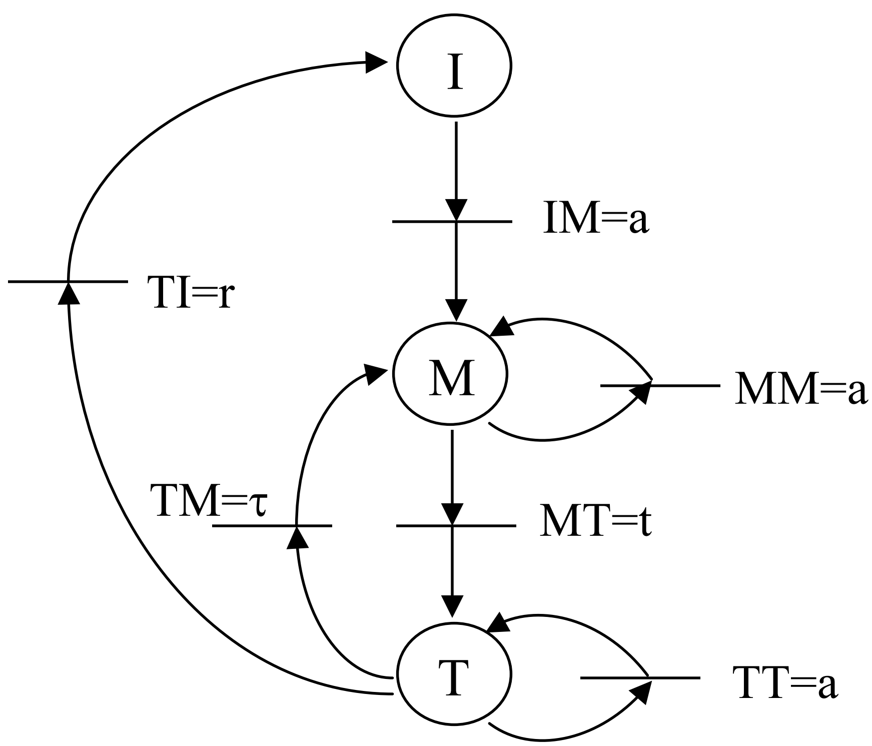
Dado um gerador (E, X, f, x_o) é possível construir uma rede de Petri (P, T, A, w, x_o) da seguinte maneira:

$$P = X$$
$$T = \{(x,x'): x \in X,\ x' = f(x,e) \text{ onde } f(x,e)!\}$$
$$A = \{(x,t): x \in X,\ t \in T \text{ e } t = (x, x')\}$$
$$\cup \{(t,x): x \in X,\ t \in T \text{ e } t = (x', x)\}$$
$$w(a) = 1 \text{ para todo } a \in A$$

O estado inicial x_o é representado pela marcação do lugar correspondente no conjunto P.

Exemplo: Protocolo de Comunicação





Observações:

•Este método não é único, podendo haver outras RP’s que melhor representem o sistema.

•Uma vantagem das redes de Petri ocorre quando se faz a associação de sistemas, correspondendo à associação assíncrona:

Rede de Petri: pequeno esforço adicional para modelagem.

Gerador: novo sistema muito mais complexo (explosão combinacional no número de estados)

•A abordagem baseada em geradores contudo, apresenta melhores resultados no que diz respeito a *decidabilidade*.