

DIM0436

7. Autômatos & redes de Petri

Richard Bonichon

20140812

1 Autômatos

2 Redes de Petri

1 Autômatos

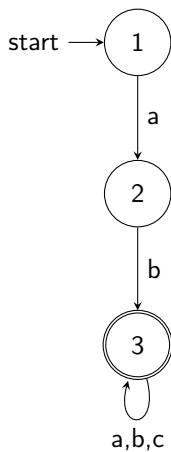
2 Redes de Petri

Autômatos determinísticos (DFA)

Um autômato é uma 5-tupla $(Q, \Sigma, \delta, q_0, F)$, onde:

- Q é um conjunto finito de estados.
- Σ é um conjunto finito de símbolos, chamado de alfabeto do autômato.
- δ é a função de transição, isto é, $\delta : Q \times \Sigma \rightarrow Q$.
- q_0 é o estado inicial, isto é, o estado do autômato antes de qualquer entrada ser processada, onde $q_0 \in Q$.
- F é um conjunto de estados de Q (isto é, $F \subseteq Q$) chamado de estados de aceitação.

Exemplo



- $Q = \{1, 2, 3\}$

- $\Sigma = \{a, b, c\}$

- $\delta =$

	a	b	c
1	2		
2		3	
3	3	3	3

- $q_0 = 1$

- $F = \{3\}$

Autômatos indeterminísticos (NFA)

Definição (Autômatos indeterminísticos (NFA))

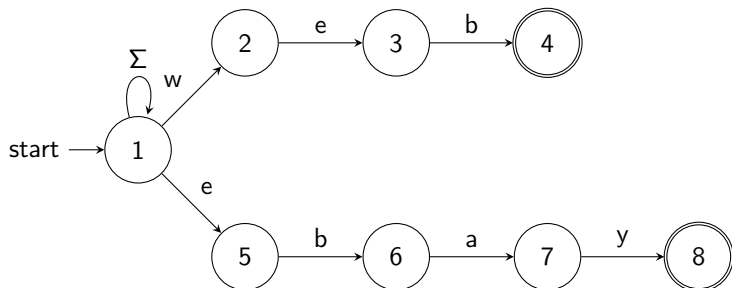
Um autômato é uma 5-tupla $(Q, \Sigma, \delta, q_0, F)$, onde:

- Q é um conjunto finito de estados.
- Σ é um conjunto finito de símbolos, chamado de alfabeto do autômato.
- δ é a função de transição, isto é, $\delta : Q \times \Sigma \rightarrow (P)Q$.
- q_0 é o estado inicial, isto é, o estado do autômato antes de qualquer entrada ser processada, onde $q_0 \in Q$.
- F é um conjunto de estados de Q (isto é, $F \subseteq Q$) chamado de estados de aceitação.

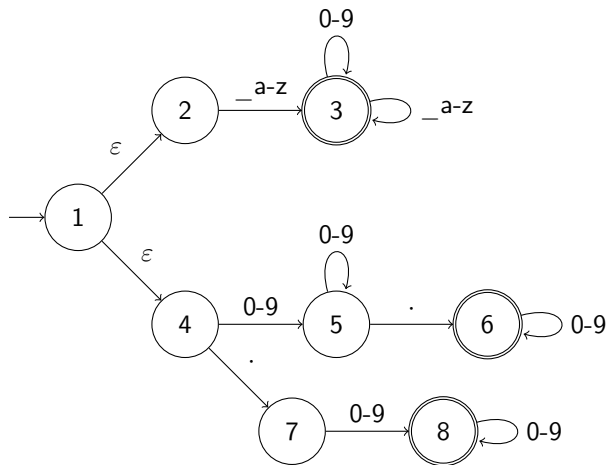
Observações

- A palavra vazia ε faz parte dos símbolos autorizados para ε -NFA.
- Sintaticamente : $DFA \subseteq NFA \subseteq \varepsilon$ -NFA

Exemplo (NFA)



Exemplo (ϵ -NFA)



Expressões regulares (ER)

Definição minimal

- ε é a cadeia de caracteres vazia
 - $a \in \Sigma$ é um literal
- Se R e S são expressões regulares:
- $R \cdot S$: concatenação de
 - $R|S$: alternância
 - R^* : fecho de Kleene

Exemplo (E)

- $[_a - b][a - b0 - 9_]^*$
- $[0 - 9] + |[0 - 9] + \cdot|[0 - 9] + \cdot[0 - 9] +$

Autômatos com pilha

Definição

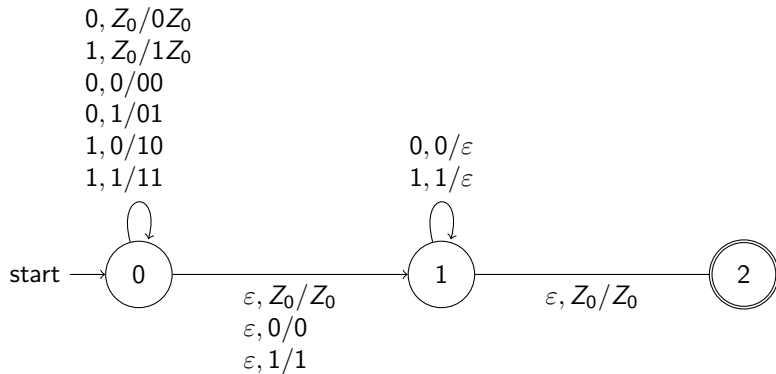
Um autômato de pilha é formalmente definido por uma 7-tupla:

$$PDA = (Q, \Sigma, \Gamma, \Delta, q_0, Z_0, F)$$

Onde:

- Q é um conjunto finito de estados.
- Σ é um conjunto finito de símbolos, denominado alfabeto de entrada.
- Γ é um conjunto finito de símbolos, denominado alfabeto da pilha.
- $\Delta \subseteq (Q \times \Sigma^* \times \Gamma^*) \times (Q \times \Gamma^*)$ é a relação de transição.
- $q_0 \in Q$ é o estado inicial.
- $Z_0 \in \Gamma$: o símbolo inicial, conteúdo inicial da pilha.
- $F \subseteq Q$ é o conjunto de estados finais (ou de aceitação).

Exemplo



Regra

$$(q, aw, X\beta) \rightarrow (p, w, \alpha\beta)$$

No exemplo

$$\begin{array}{lcl} q_0, 1111, Z_0 & \xrightarrow{1, Z_0 / 1Z_0} & q_0, 111, 1Z_0 \\ & \xrightarrow{1, 1 / 11} & q_0, 11, 11Z_0 \\ & \xrightarrow{1, 1 / 11} & q_0, 1, 111Z_0 \\ & \xrightarrow{1, 1 / 11} & q_0, \varepsilon, 1111Z_0 \\ & \xrightarrow{\varepsilon, 1 / 1} & q_1, \varepsilon, 1111Z_0 \\ & \xrightarrow{??} & \odot \end{array}$$

Reconhecimento de linguagens

Linguagens regulares

- A classe de linguagens reconhecida por autômatos determinísticos é as *linguagens regulares*.
- Reconhecimento de palavras numa frase
- $ER = \varepsilon - NFA \leftrightarrow NFA \leftrightarrow DFA$

Gramáticas livres de contexto

- A classe de linguagens reconhecida por autômatos com pilha é as *linguagens livres de contexto*.
- Reconhecimento de frases

Aplicações e outros tipos

Uso

- Compilação
 - ▶ Lexing (análise léxica)
 - ▶ Parsing (análise sintática)
- Model-checking
- Biologia
- Simulação do universo (?)

Outros autômatos

- **Automâtos de Büchi**: é um tipo de autômato ω , que estende um autômato finito para entradas infinitas. Ele aceita como entrada uma sequência infinita sse existe uma execução do autômato que visita (pelo menos) um dos estados finais infinitas vezes.
- **Autômatos celulares**: jogo da vida

1 Autômatos

2 Redes de Petri

- Inventado em 1962 por Carl Adam Petri

Objetivo

- Modelizar sistemas **concorrentes** e **raciocinar** sobre eles.
- Modelizar causalidade e conflitos
- É um grafo bipartido

Uma rede de Petri é composta de

Lugares ou Places representam uma condição, uma atividade ou um recurso.

Fichas, Marcas ou Tokens representam o estado de um sistema.

Transições representam um evento.

Arcos indicam os lugares de entrada ou saída para as transições.

Definição formal

Redes de Petri

Uma rede de Petri (RP) é um quintuplo:

- $P = p_1, p_2, \dots, p_n$, conjunto (finito) de *places*
- $T = t_1, t_2, \dots, t_m$, conjunto (finito) de transições
- $F \subseteq (P \times T) \cup (T \times P)$, conjunto de arcos (de saída e de entrada)
- $W : F \rightarrow \mathbb{N}^*$, função de peso
- $M_0 : P \rightarrow \mathbb{N}$ é a marcação inicial

Requisitos

$$P \cap T = \emptyset \wedge P \cup T = \emptyset$$

Execução: o disparo

- O comportamento dinâmico duma RP é simulado através trocas de marcações, a partir de M_0 .

Execução: o disparo

- O comportamento dinâmico numa RP é simulado através trocas de marcações, a partir de M_0 .
- A regra de **disparo** numa transição t é a seguinte:

Execução: o disparo

- O comportamento dinâmico numa RP é simulado através trocas de marcações, a partir de M_0 .
- A regra de **disparo** numa transição t é a seguinte:
 - 1 Uma transição é **habilitada** se todo *places* de entrada p de t tem pelo menos $w(p, t)$ *tokens*, onde $w(p, t)$ é o peso do arco de p a t .

Execução: o disparo

- O comportamento dinâmico duma RP é simulado através trocas de marcações, a partir de M_0 .
- A regra de **disparo** duma transição t é a seguinte:
 - 1 Uma transição é **habilitada** se todo *places* de entrada p de t tem pelo menos $w(p, t)$ *tokens*, onde $w(p, t)$ é o peso do arco de p a t .
 - 2 Uma transição habilitada pode ser **disparada** (ou não) de acordo com os eventos acontecendo.

Execução: o disparo

- O comportamento dinâmico duma RP é simulado através trocas de marcações, a partir de M_0 .
- A regra de **disparo** duma transição t é a seguinte:
 - 1 Uma transição é **habilitada** se todo *places* de entrada p de t tem pelo menos $w(p, t)$ *tokens*, onde $w(p, t)$ é o peso do arco de p a t .
 - 2 Uma transição habilitada pode ser **disparada** (ou não) de acordo com os eventos acontecendo.
 - 3 O **disparo** de uma transição habilitada

Execução: o disparo

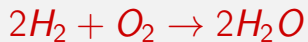
- O comportamento dinâmico duma RP é simulado através trocas de marcações, a partir de M_0 .
- A regra de **disparo** duma transição t é a seguinte:
 - 1 Uma transição é **habilitada** se todo *places* de entrada p de t tem pelo menos $w(p, t)$ *tokens*, onde $w(p, t)$ é o peso do arco de p a t .
 - 2 Uma transição habilitada pode ser **disparada** (ou não) de acordo com os eventos acontecendo.
 - 3 O **disparo** de uma transição habilitada
 - ★ tira $w(p, t)$ *tokens* dos *places* de entrada

Execução: o disparo

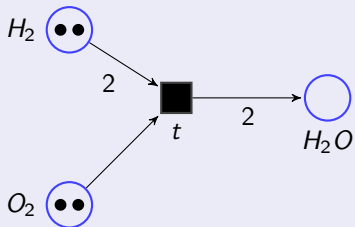
- O comportamento dinâmico duma RP é simulado através trocas de marcações, a partir de M_0 .
- A regra de **disparo** duma transição t é a seguinte:
 - 1 Uma transição é **habilitada** se todo *places* de entrada p de t tem pelo menos $w(p, t)$ *tokens*, onde $w(p, t)$ é o peso do arco de p a t .
 - 2 Uma transição habilitada pode ser **disparada** (ou não) de acordo com os eventos acontecendo.
 - 3 O **disparo** de uma transição habilitada
 - ★ tira $w(p, t)$ *tokens* dos *places* de entrada
 - ★ coloca $w(t, p)$ *tokens* nas *places* de saída

Execução: o disparo

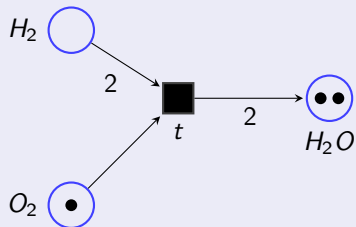
- O comportamento dinâmico duma RP é simulado através trocas de marcações, a partir de M_0 .
- A regra de **disparo** duma transição t é a seguinte:
 - 1 Uma transição é **habilitada** se todo *places* de entrada p de t tem pelo menos $w(p, t)$ *tokens*, onde $w(p, t)$ é o peso do arco de p a t .
 - 2 Uma transição habilitada pode ser **disparada** (ou não) de acordo com os eventos acontecendo.
 - 3 O **disparo** de uma transição habilitada
 - ★ tira $w(p, t)$ *tokens* dos *places* de entrada
 - ★ coloca $w(t, p)$ *tokens* nas *places* de saída
 - ★ de acordo com os pesos de cada arco.



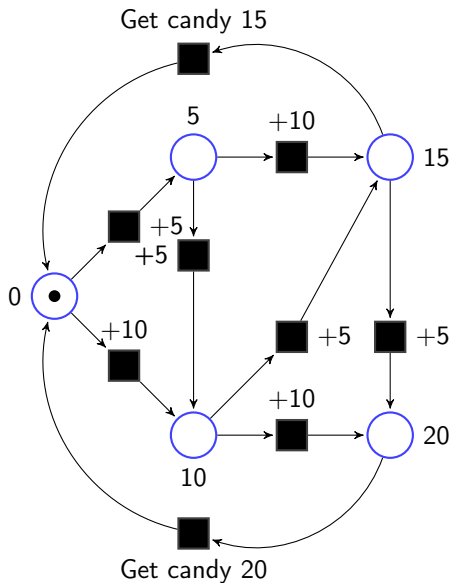
Situação inicial



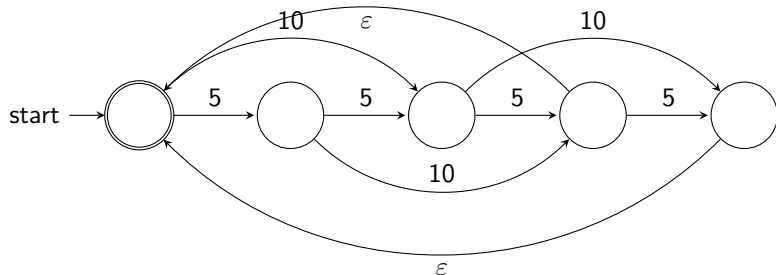
Depois do disparo

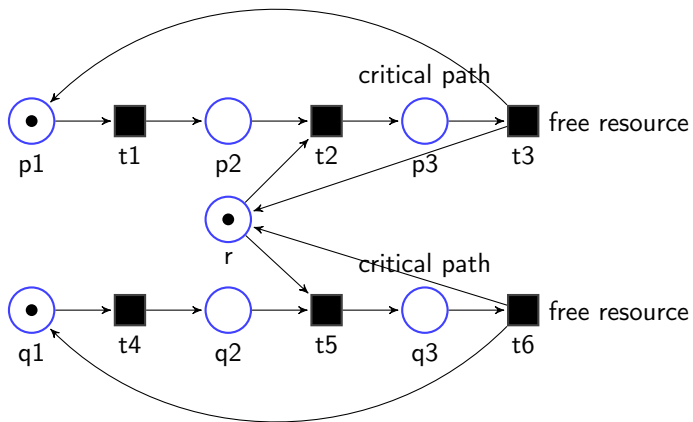


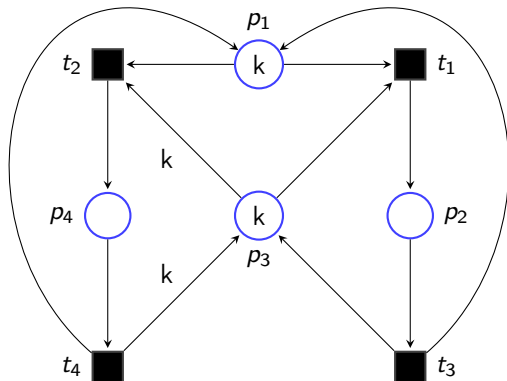
Máquina de vendas



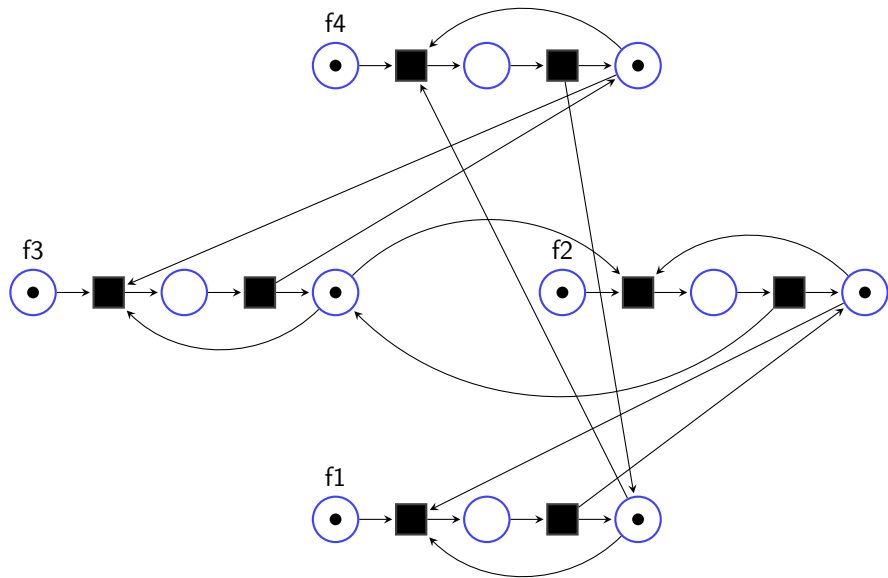
Máquinas de estado finito / DFA





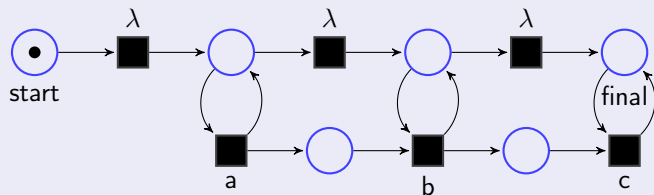


Jantar dos filósofos



Linguagens formais

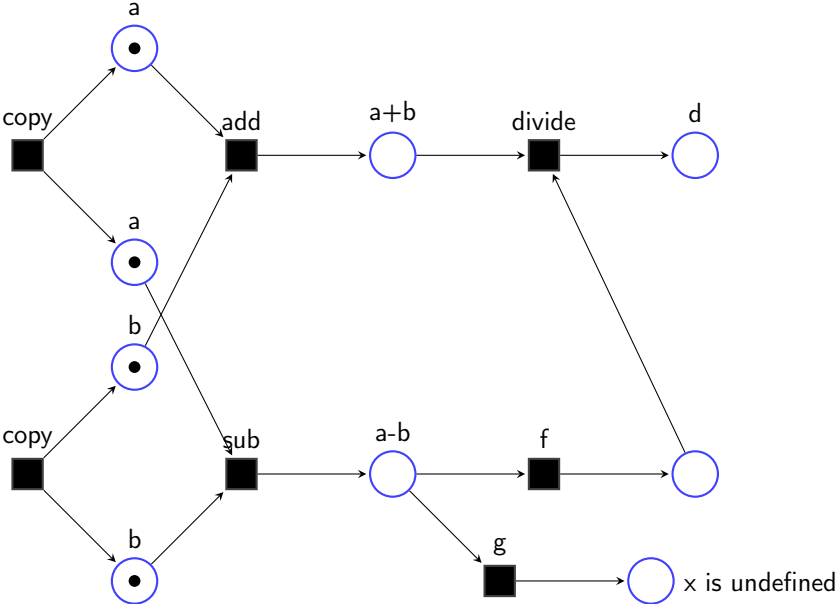
Reconhecimento de linguagem: $L(M_0) = a^n b^n c^n, n \geq 0$



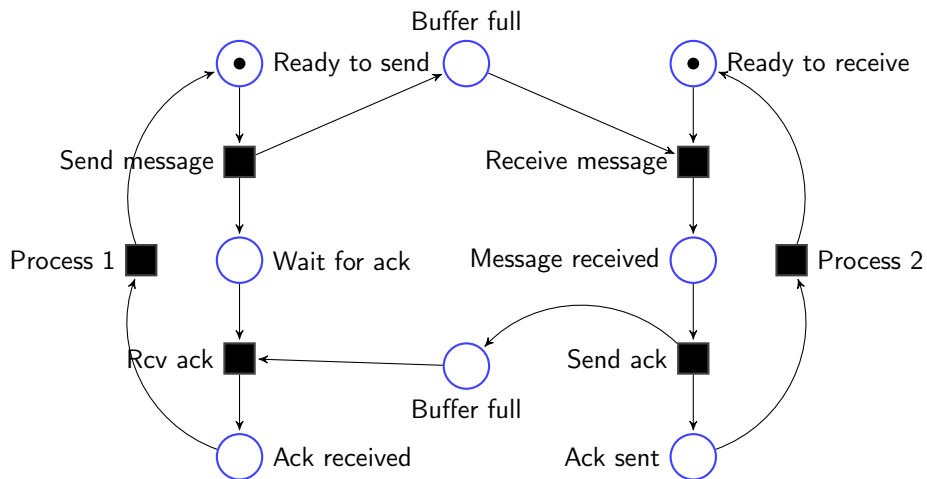
Linguagens da RP

- Linguagens reconhecidas pelas RP são linguagens sensível ao contexto (*context-sensitive*).
- Uma automôto finito reconhece só linguagens regulares.
- Hierarquia de Chomsky (de gramáticas)
regulares \subseteq livres de contexto \subseteq sensível ao contexto \subseteq estrutura de frase.

DFD



Protocolo de comunicação



Alcançabilidade (*Reachability*)

Definição

- Uma marcação M_n é **alcançável** a partir de M_0 se existe uma sequência de disparos que transforma M_0 em M_n .
- A sequência de disparos é denotada por

$$\sigma = M_0 t_1 M_1 t_2 \dots t_n M_n$$

- $R(M_0)$ é o conjunto de marcações alcançáveis numa rede RP.

Problema de alcançabilidade

- Achar se uma marcação dada $M_n \in R(M_0)$
- Esse problema é **decidível** em tempo e espaço exponenciais
- Pode-se usar árvores de alcançabilidade para enumerar as marcações alcançáveis.

Vivacidade (*Liveness*)

Definição (Vivacidade)

Uma RP é **viva** se, independente da marcação M corrente, tem uma transição que pode ser disparada.

Observação

- A vivacidade é ligada com a ausência de *deadlocks*

Níveis de vivacidade

Verificação de vivacidade

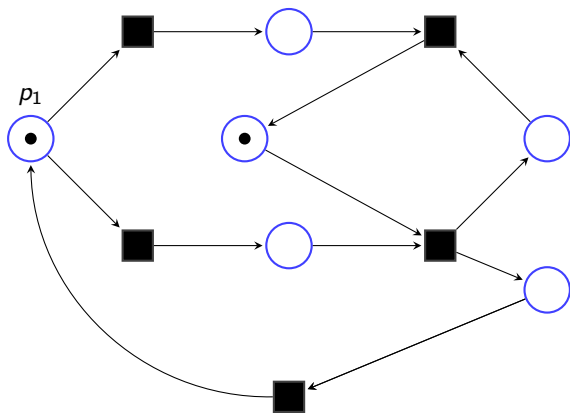
Não prático é muito difícil verificar a propriedade de vivacidade para um sistema.

Definição (Níveis de vivacidade)

Uma transição t é:

- 1 **morta** (L0-viva) se t nunca pode ser disparada numa sequência em $L(M_0)$
- 2 L1-viva (*potencialmente disparavel*) se t pode ser disparada pelo menos uma vez numa sequência em $L(M_0)$
- 3 L2-viva, se para todo inteiro $k \geq 0$, t pode ser disparada pelo menos k vezes numa sequência em $L(M_0)$
- 4 L3-viva se t acontece infinitamente numa sequência em $L(M_0)$
- 5 L4-viva (ou viva) se t é L1-viva para toda marcação M de $R(M_0)$

Exemplo



Finitude (*Boundedness*)

Definição

Uma RP é *k*-*finito* se o número de *tokens* nos *places* não ultrapassa *k*.

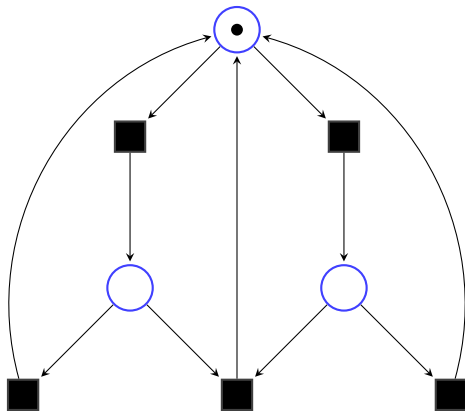
Segurança

Uma RP é *seguro* sse ele é 1-finito

Propriedades

- Qual é o maior/menor número de *tokens* nos *places*?

Exemplo

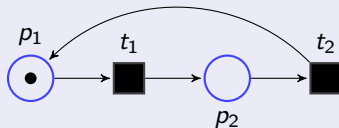


Estados caseiros (*Home states*)

Definição

- Uma marcação M é um estado caseiro se para toda marcação $M' \in R(M_0)$, M é alcançável de M' .
- Uma RP é reversível se M_0 é um estado caseiro.

Exemplo



Cobertura (*Coverability*)

Cobertura

Uma marcação M dum RP é **cobrável** se existe uma marcação M' de $R(M_0)$ tal que $M'(p) \geq M(p)$

Relação à vivacidade

Seja M a marcação minimal para habilitar uma transição t

- t é morta se e somente se M não é cobrável
- t é L1-viva se e somente se M é cobrável

Distância sincrônica

Significado

A distância sincrônica é uma métrica que permite avaliar a dependência mutual de dois eventos do sistema.

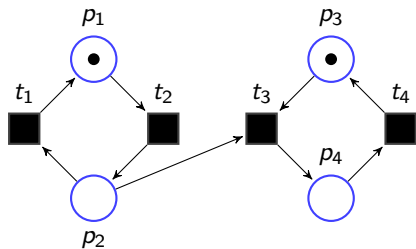
Definição (Distância sincrônica)

Seja t_1 e t_2 duas transições duma RP (N, M_0) . A distância sincrônica entre t_1 e t_2

$$d_{12} = \max_{\sigma} |\bar{\sigma}(t_1) - \bar{\sigma}(t_2)|$$

- σ uma sequência de disparos
- $\bar{\sigma}(t)$ é o número de vezes que t é disparada em σ

Exemplo



- $d_{12} = 1$
- $d_{34} = 1$
- $d_{13} = \infty$

Justiça (*Fairness*)

Definição (Justiça limitada (Bounded fairness))

- Duas transições t_1 e t_2 são numa relação justa limitativa se o número máximo de vezes que t_1 ou t_2 pode ser disparada **sem** que outra seja disparada é limitado.
- Uma RP é **limitativamente justa** se todos os pares de transições dessa RP são numa relação justa limitativa.

Definição (Justiça incondicional)

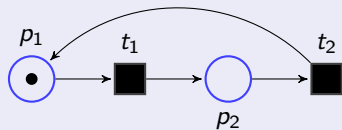
- Uma sequência de disparos σ é incondicionalmente justa se ela é:
 - ▶ finita, ou
 - ▶ toda transição da RP acontece infinitamente frequentemente.
- Uma RP é incondicionalmente justa se toda sequência de disparos dela é incondicionalmente justa.

Relação entre as duas formas de justiça

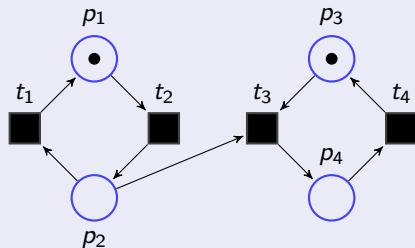
- Justiça limitada \Rightarrow Justiça incondicional
- Justiça incondicional numa rede limitado \Rightarrow justiça limitada nessa rede.

Exemplo

RP limitativamente justa



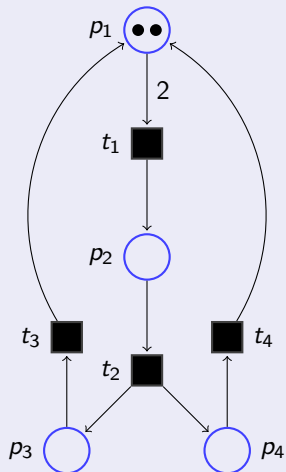
RP injusta



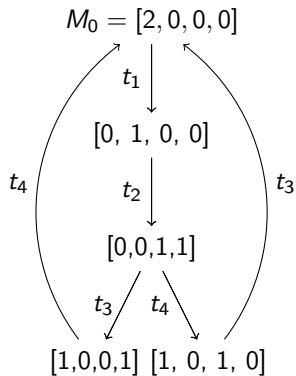
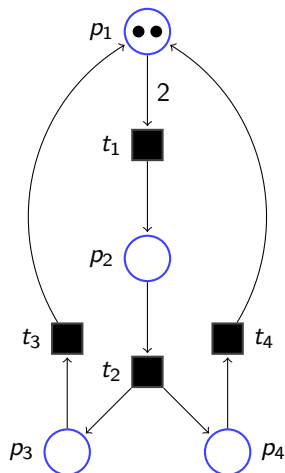
Notação de marcação

Uma marcação é um vetor de *tokens*

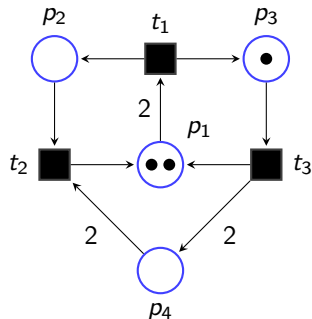
RP



Árvore de alcançabilidade



Notação matricial



Matriz

$$\begin{pmatrix} -2 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & -2 & 2 \end{pmatrix}$$

Marcação

$$M_0 = (2 \ 0 \ 1 \ 0)$$

Transição

$$M_k = M_{k-1} + A^T u_k$$

- A análise de RP pode ser dividida em 3 partes
 - ▶ Análise por enumeração
 - ▶ Análise por matriz de incidência e equação de estado
 - ▶ Análise por redução/decomposição
 - ★ Fusão de *place* em série
 - ★ Fusão de transições em série
 - ★ Fusão de transições em paralelo
 - ★ Fusão de *places* em paralelo
 - ★ Eliminação de *places* em *loop*
 - ★ Eliminação de transições em *loop*
- Lista de ferramentas e recursos
<http://www.informatik.uni-hamburg.de/TGI/PetriNets/>

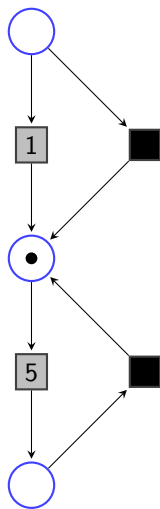
Porque?

- Disparos das RPs tradicionais são baseadas na presença dos *tokens*
- No entanto, pode ser necessário associar ao disparo um retardo. Sendo assim, para cada **transição** atribuímos um valor de tempo.
- Elas permitem uma análise quantitativa dos sistemas além da qualitativa

Significados possíveis

- Tempo de aquisição de um recurso.
- Tempo de utilização de um recurso.
- Tempo em que não necessita de um recurso.

Exemplo



Descrição

- t_1 é temporizada em 1 segundo.
- t_2 é temporizada em 5 segundos.
- t_3 e t_4 não são temporizadas

Disparos

Durante os 4 primeiros segundos a transição t_1 estará habilitada. No quinto segundo as transições t_1 e t_2 estarão habilitadas e haverá um conflito de disparo.

Redes de Petri estocásticas

Porque ?

- Redes temporizadas determinísticas não são suficientes para modelar sistemas estocásticos, que possuem taxas aleatórias.
- Uma temporização de distribuição exponencial é associada a cada transição
- Sistemas de produção onde o tempo de funcionamento real entre 2 paradas de uma máquina é aleatório são um exemplo destes sistemas

Redes de Petri estocásticas (RPE)

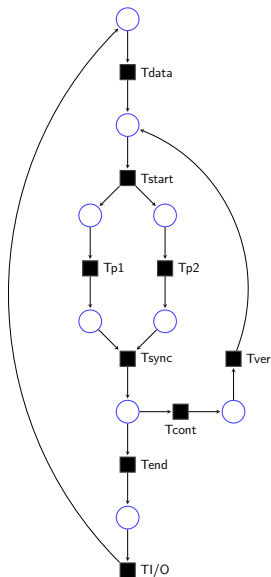
Uma RPE é uma 6-tupla $(P, T, F, W, M_0, \Delta)$:

- Δ : conjunto das taxas de disparo associadas às transições que obedecem a uma **distribuição exponencial**

Cadeias de Markov

Uma RPE é isomórfica a uma Cadeia de Markov finita e de tempo contínuo.

Exemplo



Parâmetros

Transição	Taxa	Valor
Tdata	λ	1
Tstart	τ	1000
Tp1	μ_1	10
Tp2	μ_2	5
Tsync	σ	2500
Tend	α	9900
Tcont	β	100
TI/O	ν	25
Tver	ω	0.5

Definição (RPC)

Uma RP colorida é uma RP com:

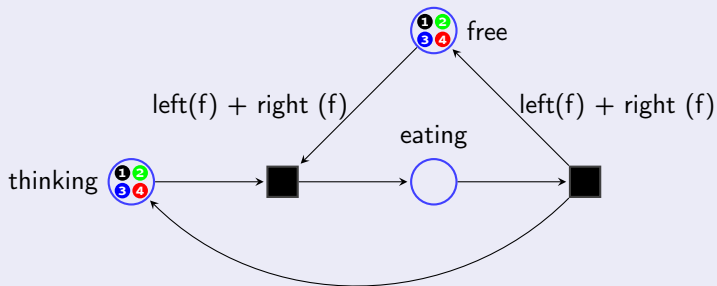
- Σ um conjunto de cores
- $C : P \rightarrow \Sigma$ uma função de cor

Significado

- Cores representam tipos de dados

Exemplo

RPC



Observações




- $\text{left}(i) = i$
- $\text{right}(i) = (i + 1) \bmod 4$

Modelização de

- sistemas distribuídos
- protocolos de comunicação
- sistemas de manufatura
- VLSI
- base de dados
- logística
- sistemas de tempo real

1 Autômatos

2 Redes de Petri

-  John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman, *Introduction to automata theory, languages, and computation*, Addison-Wesley, 2003.
-  Tadao Murata, *Petri Nets: Properties, Analysis and Applications.*, Proceedings of the IEEE, April 1989, NewsletterInfo: 33Published as Proceedings of the IEEE, volume 77, number 4, pp. 541–580.
-  Wolfgang Reisig, *Understanding Petri Nets - Modeling Techniques, Analysis Methods, Case Studies*, Springer, 2013.

Perguntas ?



<http://dimap.ufrn.br/~richard/dim0436>