



Design Lab

PMR 5237

Modelagem e Design de Sistemas Discretos

Aula 6

Prof. Dr. José Reinaldo Silva
reinaldo@usp.br

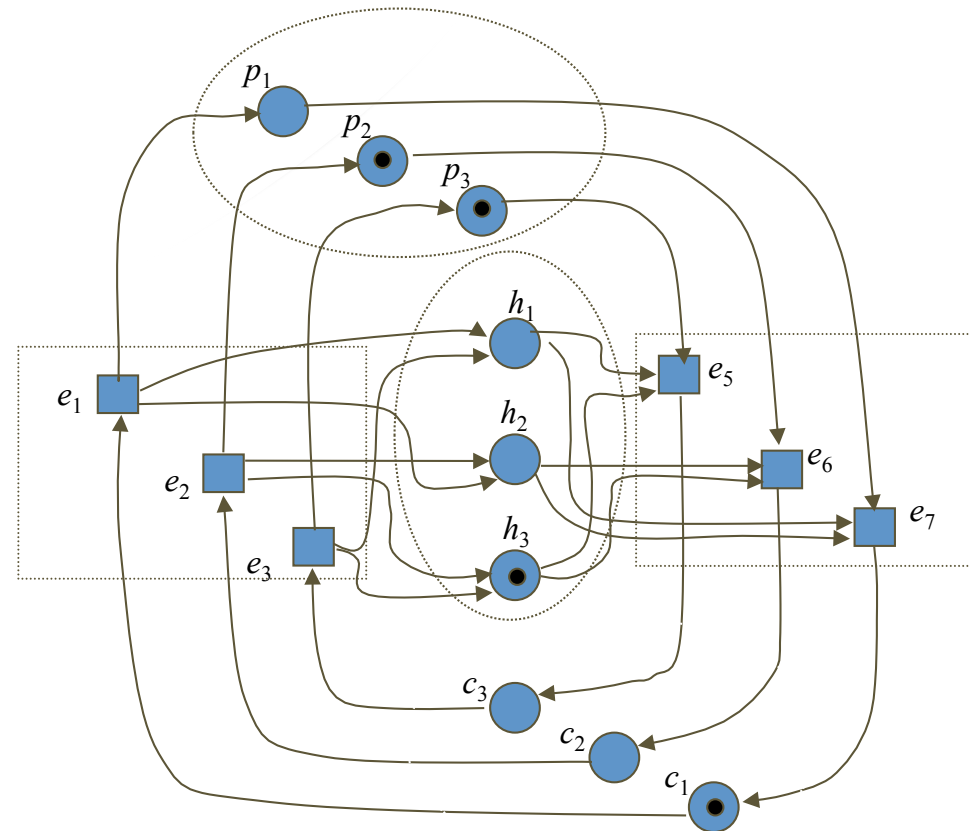
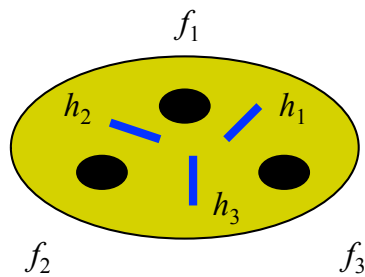


Escola Politécnica da USP



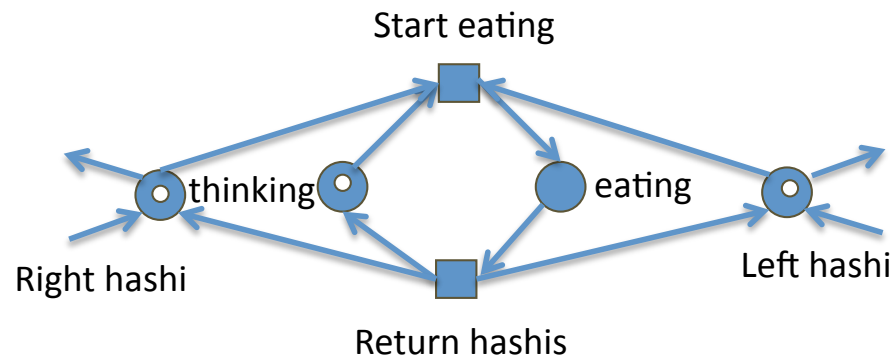
Dobramento e individualidade das marcas

O exemplo dos filósofos



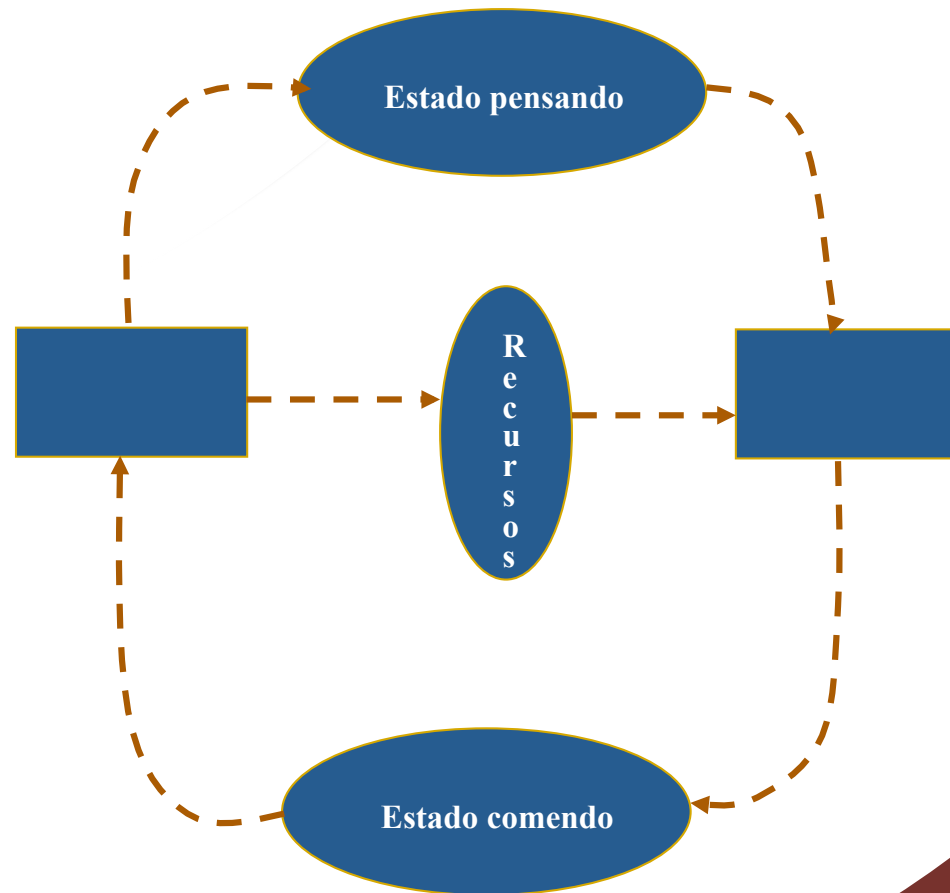
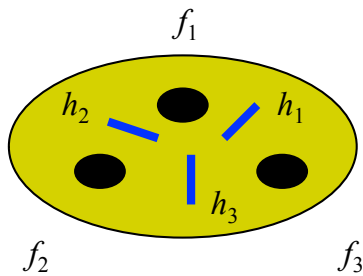
Uma visão composicional do problema

Certamente, uma forma de olhar o problema é procurar, logo de início, o relacionamento entre TODOS os seus elementos constituintes, como feito no slide anterior. Mas é possível também olhar cada um dos elementos composicionais, especialmente aqueles que apresentam propriedades repetitivas. No exemplo dos filósofos, se olharmos cada um dos filósofos, temos:



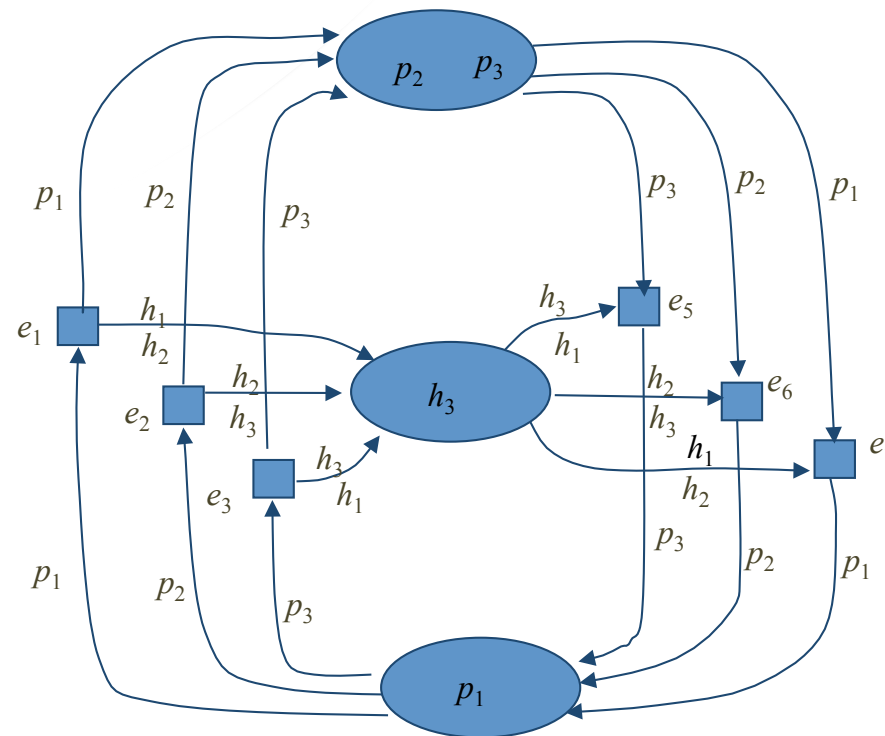
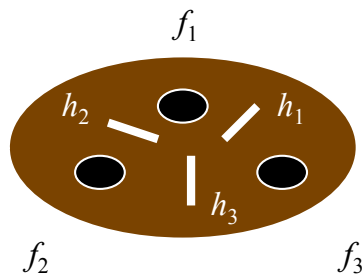
Explorando Simetrias

O exemplo dos filósofos



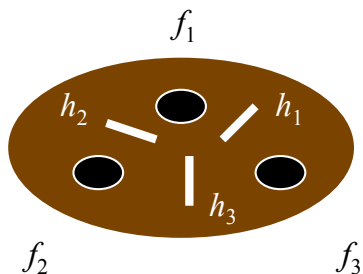
Distinguibilidade das marcas

O exemplo dos filósofos



Introduzindo predicados

O exemplo dos filósofos



$$P = \{p_1, p_2, p_3\}$$

$$H = \{h_1, h_2, h_3\}$$

$$U = P \cup H$$

$$l : P \rightarrow H$$

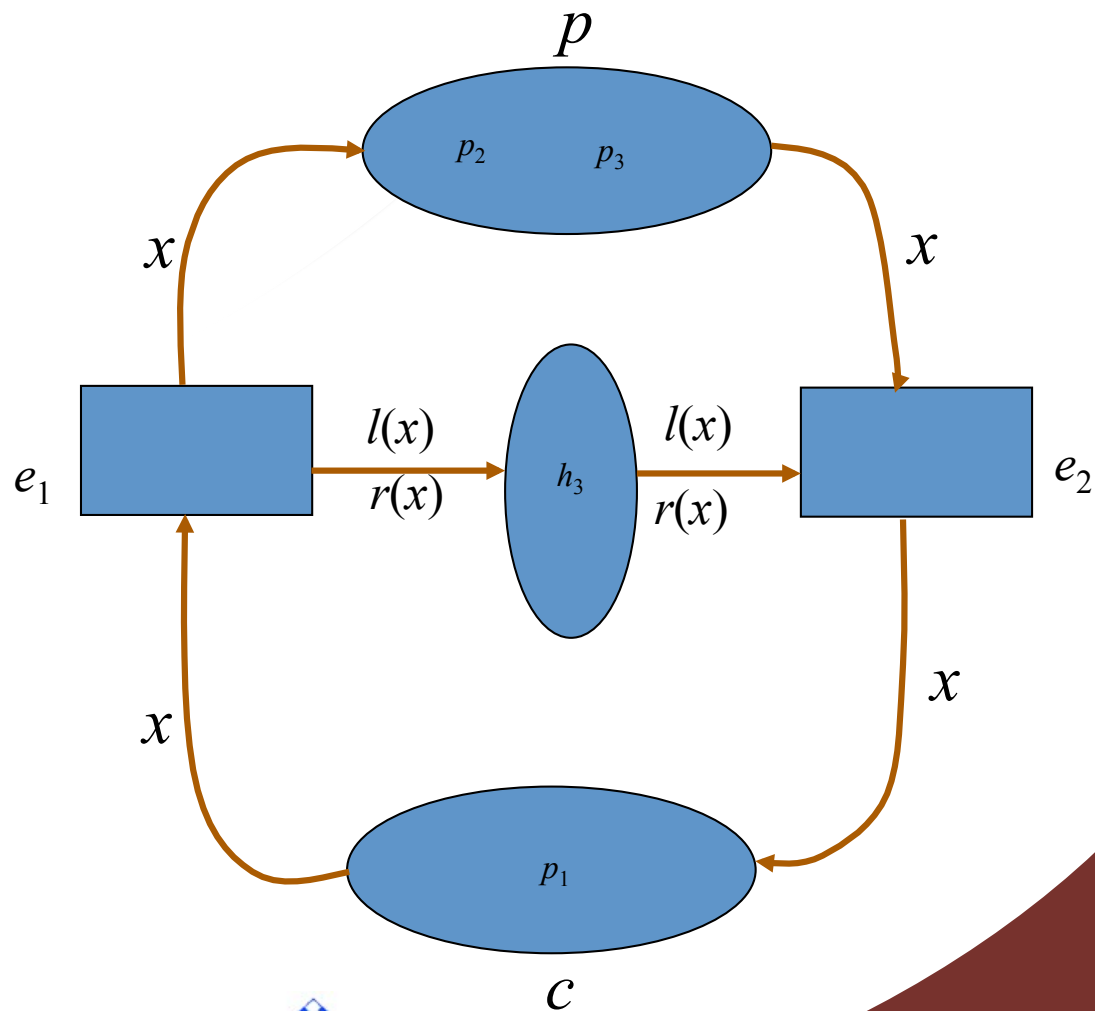
$$p_i \rightarrow h_i$$

$$r : P \rightarrow H$$

$$p_1 \rightarrow h_2$$

$$p_2 \rightarrow h_3$$

$$p_3 \rightarrow h_1$$



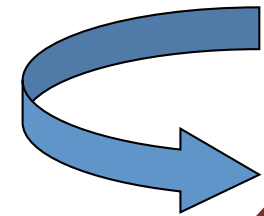
Fatoração

A este processo de exploração das simetrias de uma rede para produzir uma nova rede dinamicamente equivalente à anterior mas de tamanho menor dá-se o nome de fatoração.

A rede resultante da fatoração é **chamada** de rede-quociente.

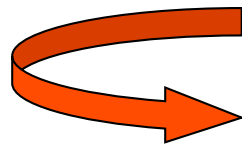
Voltando ao problema da distinguibilidade

Como vimos anteriormente com o problema dos filósofos, a introdução da distinguibilidade das marcas (dobramento) não afeta as propriedades principais das redes.



... isto é,

- preserva a dualidade
- preserva o conceito de localidade
- preserva o princípio da concorrência
- preserva o princípio da representação gráfica
- preserva o princípio da representação algébrica



Não saímos do domínio das redes de Petri

High Level Nets

Como vimos na aula passada podemos avançar bastante nas redes coloridas mas com a aparente desvantagem de perder um pouco a intuição no processo de modelagem. Ademais, o problema do cálculo das propriedades não está claro até aqui, dado que não temos mais uma matriz de incidência composta de inteiros.

Portanto o nosso objetivo passa a ser, por enquanto, tentar compor uma rede de alto nível, baseada no dobramento das redes (seja durante a modelagem ou depois de sintetizada uma rede P/T), e nas simetrias, porém simples, e que, acima de tudo preserve as propriedades descritas na transparências anterior, e além disso, que garanta um algoritmo de transformação direto para as redes clássicas e de volta, das redes clássicas para uma rede de alto nível.



Referências

Seguiremos daqui em diante duas referências básicas: o artigo do Einar Smith, pertencente ao mesmo LNCS que contém o último curso de Redes de Petri dado pelos maiores pesquisadores da área, e a própria definição da rede de alto nível, segundo o padrão ISO/IEC 15.909-1 (ambos colocados no Moodle).

Smith, E.; Principles of High Level Nets, LNCS 1491, pp. 174-210.

ISO/IEC 15.909, Final Draft, v.4.7.1; High Level Nets: Concepts, Definitions and Graphical Notations, October, 2000.



Aplicações (segundo o draft)

- requirements analysis;
- development of specifications, designs and test suites;
- descriptions of existing systems prior to re-engineering;
- modelling business and software processes;
- providing the semantics for concurrent languages;
- simulation of systems to increase confidence;
- formal analysis of the behaviour of critical systems; and
- development of Petri net support tools.



Fatoração

A este processo de exploração das simetrias de uma rede para produzir uma nova rede dinamicamente equivalente à anterior mas de tamanho menor dá-se o nome de fatoração.

A rede resultante da fatoração é **chamada** de rede-quociente.

Rede P/T

Fatoração



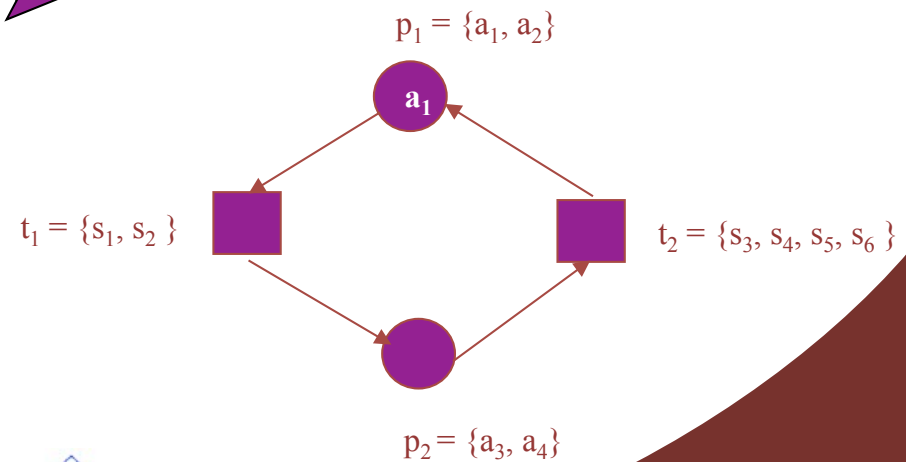
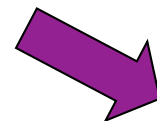
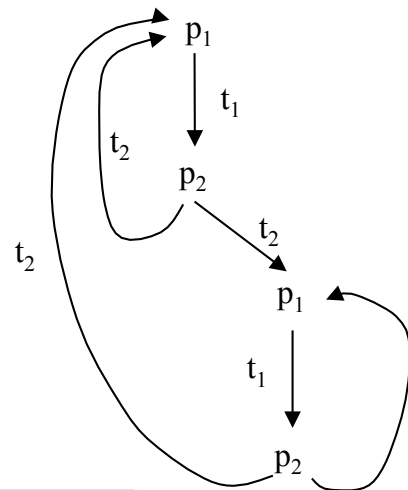
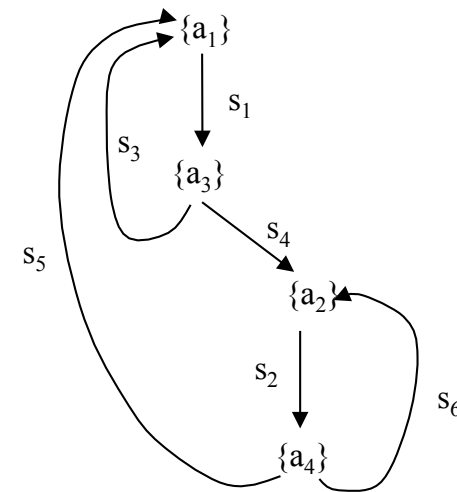
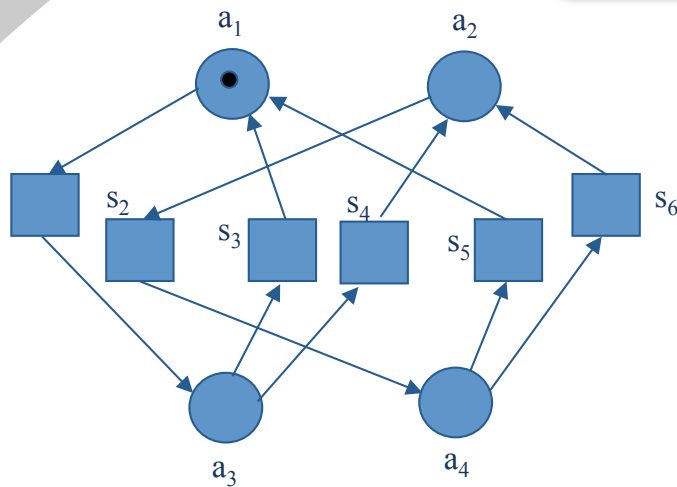
Rede Quociente



Unfolding



Um exemplo

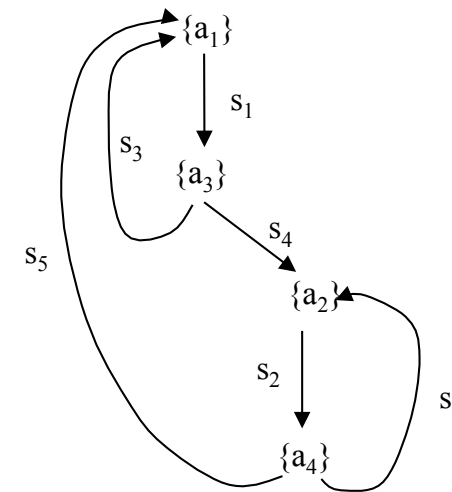
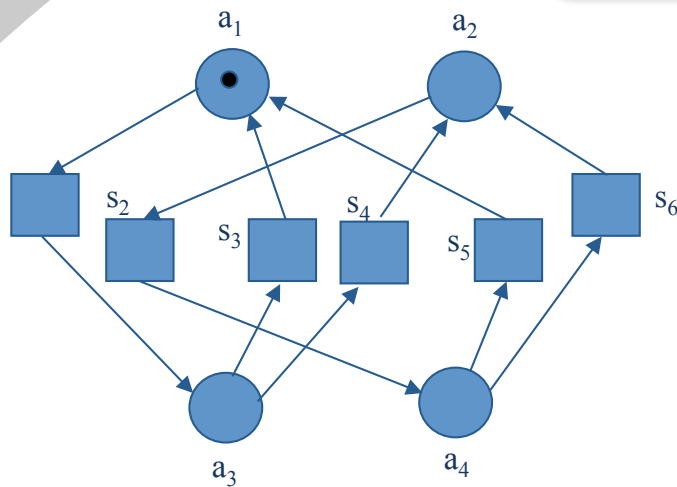


Equivalência entre redes

Def.32] Seja uma rede $N (P/T)$ e seja o seu domínio $X = P \cup T$. Existe uma equivalência ρ entre esta rede e sua rede quociente $\bar{N}$, tal que :

- i) $\forall x \in X, \bar{x} = \{y \in X \mid x \rho y\}$;
- ii) Para um dado $Y \subseteq X, \bar{Y} = \{\bar{x} \mid x \in Y\}$;
- iii) ρ preserva o sort, isto é, $\rho \cap (P \times T) = \emptyset$;
- iv) A relação de fluxo $\bar{F}$, definida sobre a rede $\bar{N}$, é dada por :
$$\bar{x} \bar{F} \bar{y} \Leftrightarrow \exists (x \in \bar{x} \text{ e } y \in \bar{y}). x F y;$$
- v) A estrutura da rede quociente é dada por $\bar{N} = (\bar{P}, \bar{T}, \bar{F})$.

Identificando a equivalência

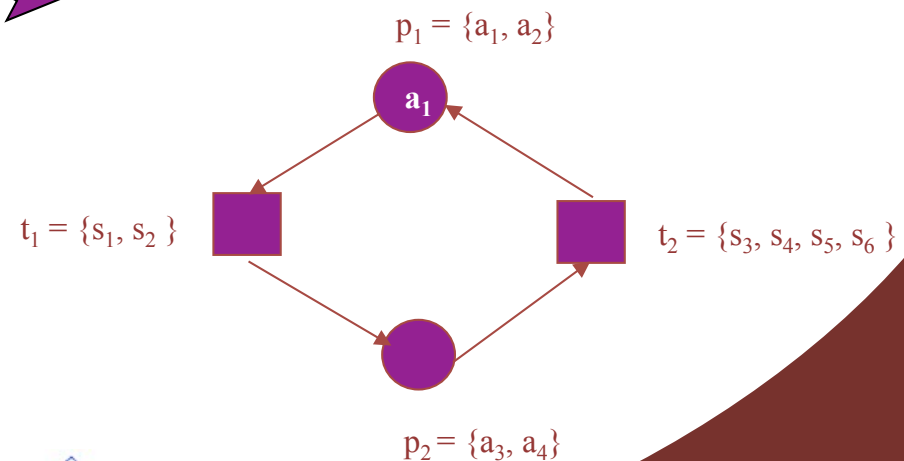
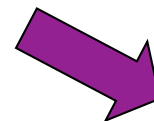


$$\rho: \bar{x}_1 = \{a_1, a_2\} \rightarrow p_1$$

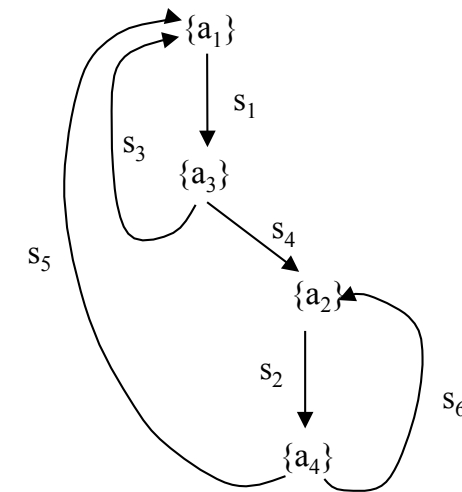
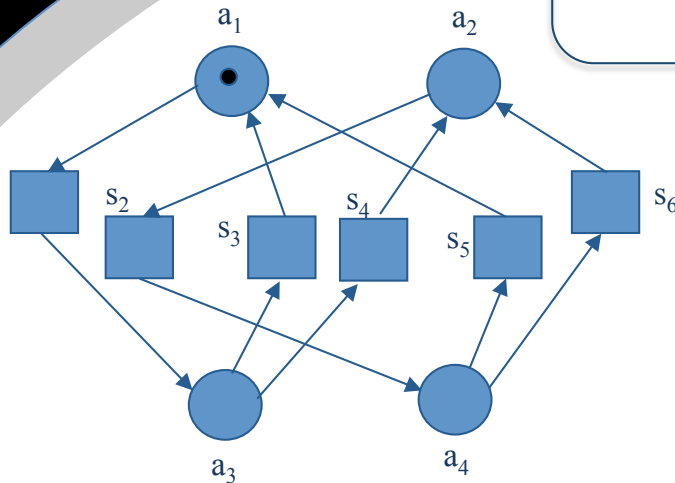
$$\bar{x}_2 = \{a_3, a_4\} \rightarrow p_2$$

$$\bar{x}_3 = \{s_1, s_2\} \rightarrow t_1$$

$$\bar{x}_4 = \{s_3, s_4, s_5, s_6\} \rightarrow t_2$$



Mapeando a marcação



$$\rho: \bar{x}_1 = \{a_1, a_2\} \rightarrow p_1$$

$$\bar{x}_2 = \{a_3, a_4\} \rightarrow p_2$$

$$\bar{x}_3 = \{s_1, s_2\} \rightarrow t_1$$

$$\bar{x}_4 = \{s_3, s_4, s_5, s_6\} \rightarrow t_2$$

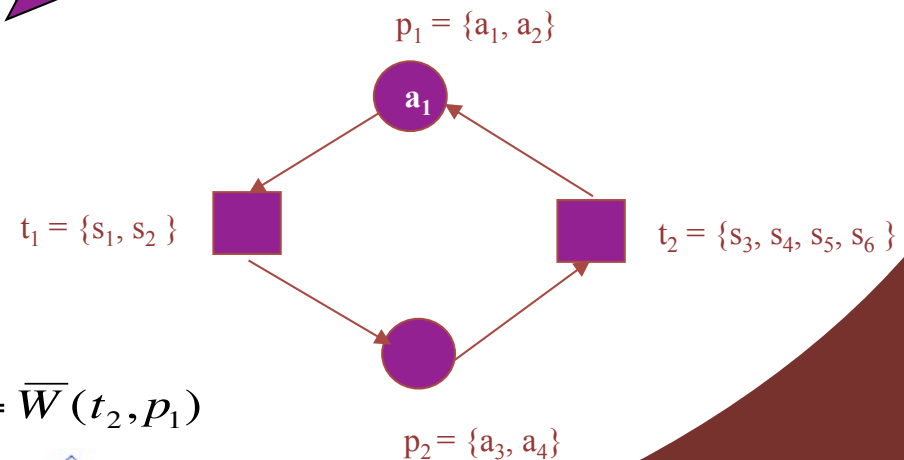
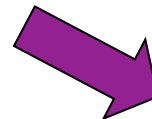
$$\bar{x} = \{x_i\}$$

$$\rho: \mathbf{M}(\bar{x}) = \mathbf{M}\{x_i\}$$

$$\mathbf{M}(p_1) = \mu \mathbf{M}\{a_1, a_2\} = 1a_1$$

$$\mathbf{M}(p_2) = \mu \mathbf{M}\{a_3, a_4\} = 0$$

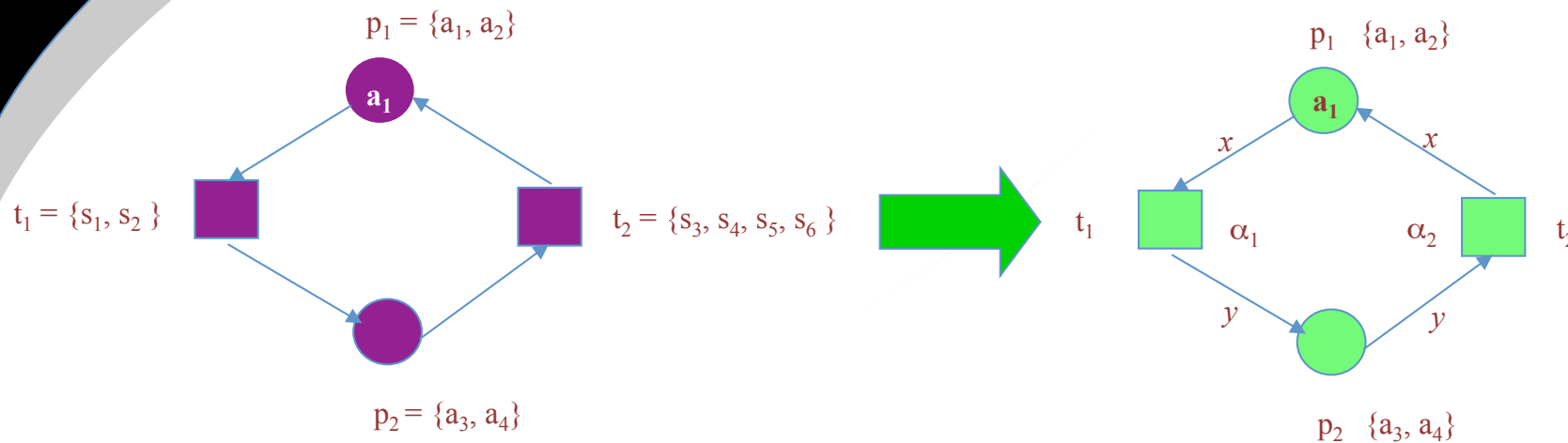
$$\bar{W}(p_1, t_1) = a_1 \vee a_2 = \bar{W}(t_1, p_2) = \bar{W}(p_2, t_2) = \bar{W}(t_2, p_1)$$



A rede quociente gerada

Def.33] Seja um sistema Place/Transition com estrutura N , $PT = (N, K, W, M_0)$, e uma bijeção (equivalência) ρ que preserva sort. Chama-se rede quociente com relação a ρ ao sistema $\overline{PT} = (\overline{N}, \overline{K}, \overline{W}, \overline{M}_0)$, com a mesma dinâmica de PT .

Rumo às BHLN



- lugares possuem marcas individualizadas
- transições ocorrem em diferentes modos restritos à regra de localidade (agir somente sobre o seu pre-set e pós-set)

Smith, E.; Principles of High Level Nets, LNCS, 1491, Springer Verlag, 1998.

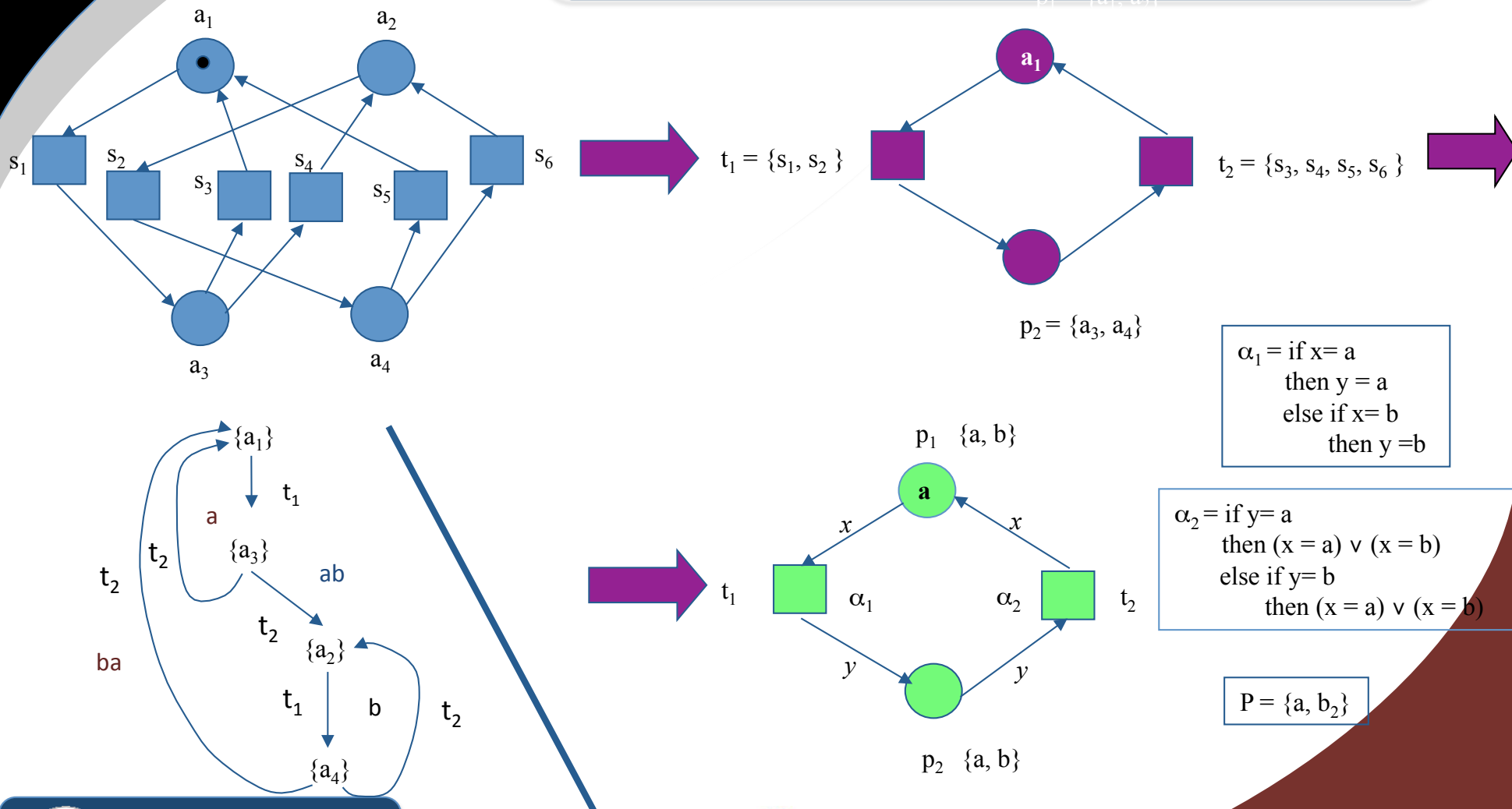
$\alpha_1 = \text{if } x = a_1$
 then $y = a_3$
 else if $x = a_2$
 then $y = a_4$

$\alpha_2 = \text{if } y = a_3$
 then $(x = a_1) \vee (x = a_2)$
 else if $y = a_4$
 then $(x = a_1) \vee (x = a_2)$

$P = \{a_1, a_2, a_3, a_4\}$



Modos de funcionamento



High Level Petri Net

ISO/IEC 15.909

Def.34] Uma rede de Petri de alto nível, HLPN é uma estrutura dada pela n -upla $HLPN = (P, T, D; Time, Pre, Post, M_0)$ onde :

- P é um conjunto finito de elementos chamados lugares;
- T é um conjunto finito de elementos chamados transições;
- D é um conjunto finito, não - vazio, de domínios ou tipos;
- $Pre, Post : TRANS \rightarrow \mu PLACE$, onde

$$TRANS = \{(t, m) | t \in T, m \in Type(t)\}$$

$$PLACE = \{(p, g) | p \in P, g \in Type(p)\}$$

- $M_0 \in \mu PLACE$ é o multiset que denota a marcação inicial da rede.



Habilitação de transições

Uma transição $t \in T$ é dita habilitada em uma marcação M , se e somente se

$$Pre(t) \leq M$$

(onde a desigualdade é entre multisets).

Multisets

Relação de Ordem parcial : Dados dois multisets, representados pelos respectivos vetores de coeficientes sobre um conjunto de base (base set) S ,

$$m = (m_1, m_2, \dots, m_i, \dots, m_s),$$
$$n = (n_1, n_2, \dots, n_j, \dots, n_s)$$

Está definida a comparação entre estes dois multisets

$m \geq n$ se e somente se $m_i \geq n_i$ para todo i

Habilitação de passos

Generalizando o conceito de transição temos que um passo T_μ é um multiset sobre os modos de transição. Um tal passo está habilitado em uma marcação M se e somente se

$$\text{Pre}(T_\mu) \leq M$$

onde,

$$\text{Pre}(T_\mu) = \sum_{t \in T_\mu} T_\mu(t) \text{Pre}(t)$$

Regra de transição

Um passo T_μ habilitado em uma marcação M ocorre, resultando em outra marcação M' , obtida a partir da marcação original pela equação,

$$M' = M - Pre(T_\mu) + Post(T_\mu)$$



Representação Gráfica

A High-level Petri Net Graph (HLPNG) comprises:

- *A Net Graph*, consisting of sets of nodes of two different kinds, called *places* and *transitions*, and *arcs* connecting places to transitions, and transitions to places.
- *Place Types*. These are non-empty sets. One type is associated with each place.
- *Place Marking*. A collection of elements (data items) chosen from the place's type and associated with the place. Repetition of items is allowed. The items associated with places are called *tokens*.
- *Arc Annotations*: Arcs are inscribed with expressions which may comprise constants, variables (e.g., x, y) and function images (e.g., $f(x)$). The variables are typed. The expressions are evaluated by assigning values to each of the variables. When an arc's expression is evaluated, it must result in a collection of items taken from the type of the arc's place. The collection may have repetitions.
- *Transition Condition*: A boolean expression (e.g., $x < y$) inscribing a transition.
- *Declarations*: comprising definitions of place types, typing of variables, and function definitions.



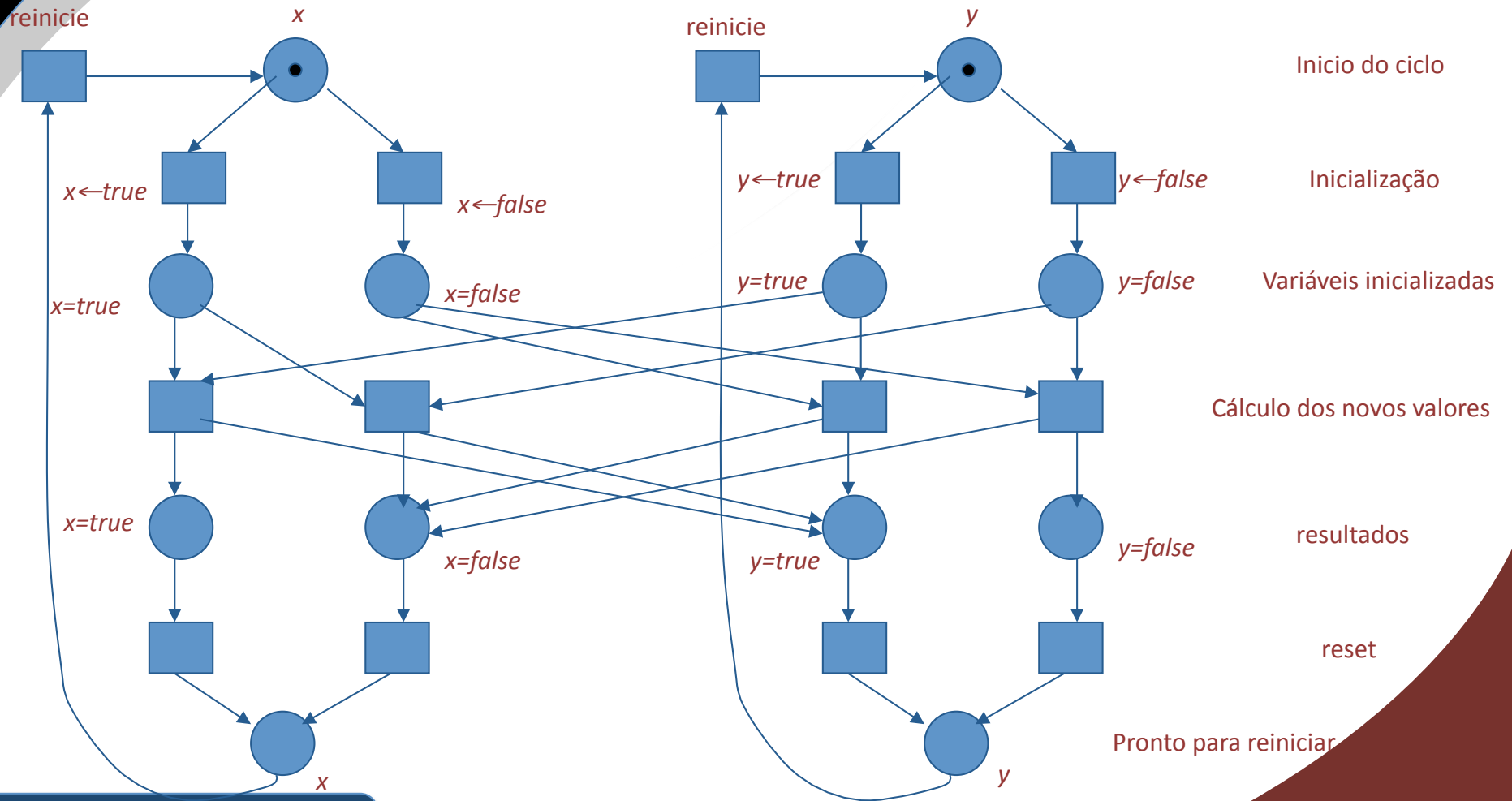
Exemplo

Tomemos como exemplo uma máquina (de fato um problema de circuitos lógicos) onde duas entradas, representadas pelas variáveis, x e y , podem assumir valores lógicos em $\{true, false\}$. A máquina opera de tal modo que sobre a variável x se subscreve $x \wedge y$ enquanto que sobre o valor de y se subscreve $x \vee y$. O conteúdo das variáveis é então eliminado e o sistema fica preparado para receber novos valores.

A rede de Petri representando a operação desta máquina é mostrada a seguir.



Uma máquina lógica



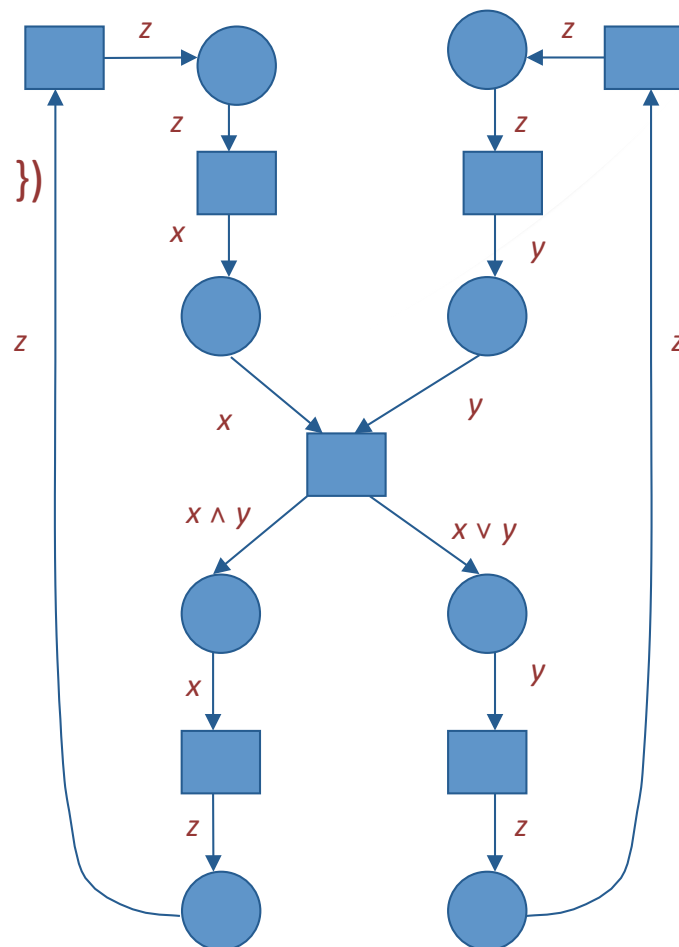
$D = \{\text{true}, \text{false}\}$

$\phi = \{ \wedge, \vee \}$

$\delta = (D; \phi) = (\{\text{true}, \text{false}\}, \{ \wedge, \vee \})$

*if $x = \text{true}$ and $y = \text{true}$
then $x \wedge y = \text{true}$
else $x \wedge y = \text{false}$*

*if $x = \text{false}$ and $y = \text{false}$
then $x \vee y = \text{false}$
else $x \vee y = \text{true}$*



Início do ciclo

Inicialização

Variáveis inicializadas

Cálculo dos novos valores

resultados

reset

Pronto para reiniciar



Exercício

Até aqui as redes de alto nível HLPN foram apresentadas como o resultado do processo de fatoração (folding) de uma rede clássica P/T. Assim as redes HLPN têm sempre uma rede base P/T e fica sub-entendido que o processo clássico de modelagem seria sintetizar uma rede P/T e depois fatorá-la para achar a rede quociente e daí a rede HLPN.

Vamos exercitar agora um processo diferente: colocar um problema onde parece bastante atraente tentar sintetizar uma rede HLPN ao invés de uma rede clássica.

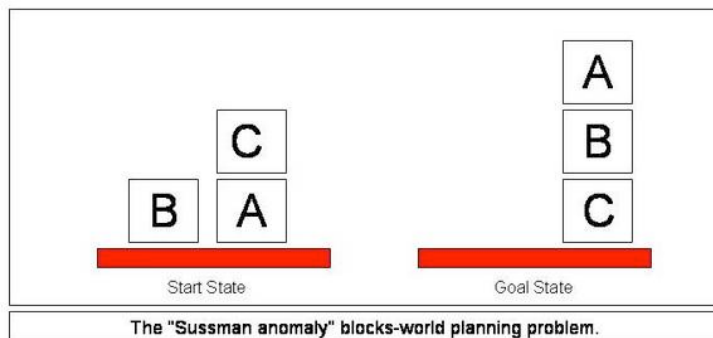


O mundo de blocos



Imagine um mundo hipotético composto de blocos tridimensionais identificados por letras maiúsculas e um robô manipulador que só consegue pegar um bloco de cada vez. Outra regra importante é que este robô só pode pegar um bloco se este for o primeiro da pilha, isto é, não existe nenhum outro bloco sobre ele.

Com estas regras pretende-se fazer um plano de ações para que o robô transforme o estado inicial mostrado na figura no estado final.

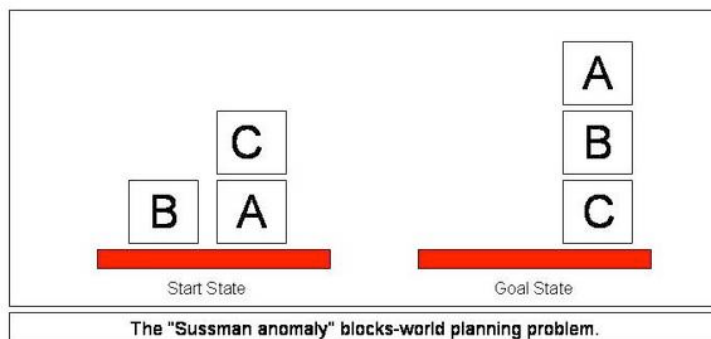


Descrição dos estados

Descritores dos estados: $ONTABLE(x)$, $ON(x,y)$, $CLEAR(x)$, $HANDEMPY$, $HOLDING(x)$

Estado inicial: $ONTABLE(B)$, $CLEAR(B)$, $ONTABLE(A)$, $ON(C,A)$, $CLEAR(C)$, $HANDEMPY$

Estado final: $ONTABLE(C)$, $ON(B,C)$, $ON(A,B)$, $CLEAR(A)$, $HANDEMPY$



As ações definidas



1. Pickup(x)

(robô pega um bloco x da mesa)

Pré-condições: ONTABLE(x), CLEAR(x), HANDEEMPTY

Pós-condição: HOLDING(x);

2. Putdown(x)

(robô deposita bloco x na mesa)

Pré-condição: HOLDING(x)

Pós-condição: ONTABLE(x), CLEAR(x), HANDEEMPTY

3. Stack(x, y)

(robô empilha bloco x sobre o bloco y)

Pré-condição: HOLDING(x), CLEAR(y)

Pós-condição: ON(x,y), HANDEEMPTY, CLEAR(x)

4. Unstack(x, y)

(robô tira bloco x de cima do bloco y)

Pré-condição: ON(x,y), CLEAR(x), HANDEEMPTY

Pós-condição: CLEAR(y), HOLDING(x)

Problema

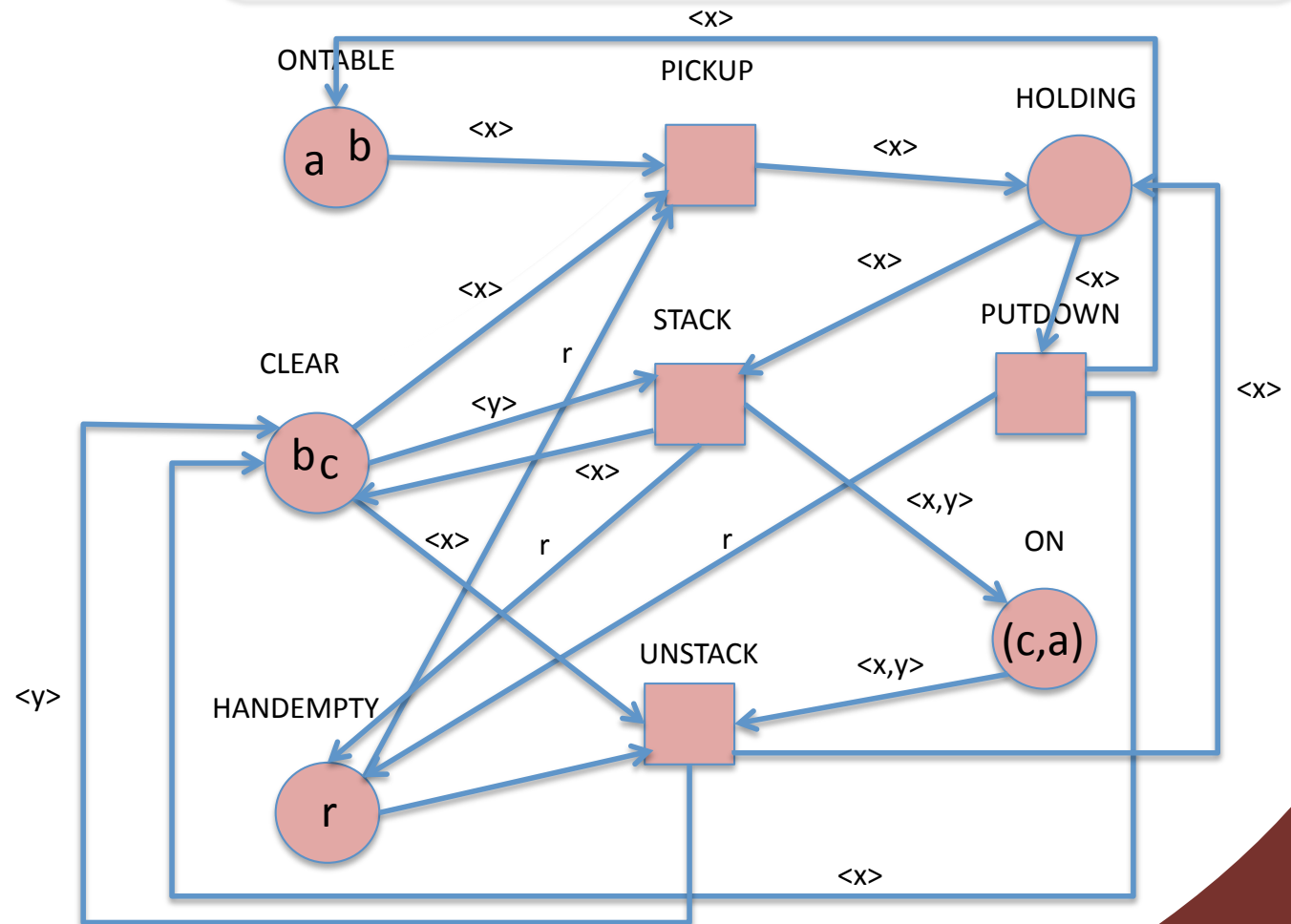
Problema já está modelado e com uma representação já definida para especificação de estados, e uma série de ações que modificam estes estados. Estes estados atuam sobre um conjunto de blocos individualizados (distinguíveis). Portanto a modelagem deste tipo de problema é direta e deve ser feita com vantagem usando a HLPN (comparado a fazer a rede clássica primeiro).

A modelagem do problema fica assim bem simples.

O Modelo HLPN

Blk:={a,b,c}
 R:= {r}
 OnB: Blk X Blk

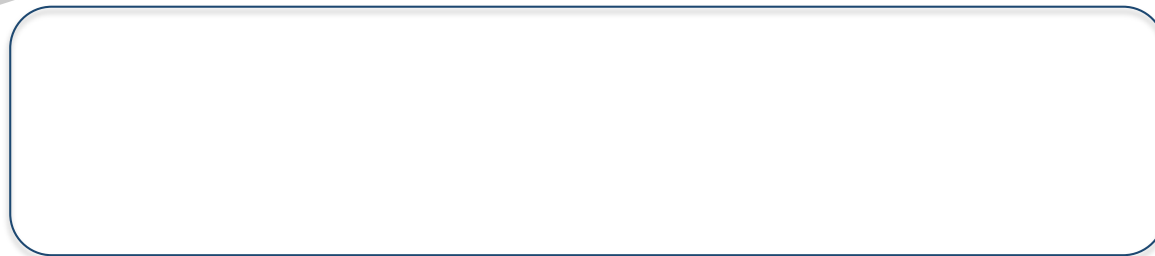
x,y: Blk



Exercício para casa

1. Modelar ou fazer o unfolding da rede da transparencia anterior de modo a ter uma rede P/T para o problema.
2. Use uma ferramenta clássica e insira a rede P/T sintetizada
3. Pesquise o significado da “anomalia de Sussman”
4. É possível reproduzir a anomalia de Sussman da rede clássica?
5. Ache uma ferramenta de suporte para Redes HLPN e insira o problema anterior. Reproduz a anomalia de Sussman?
6. Dê a sua opinião: o que é mais fácil neste caso? Gerar a rede P/T ou sintetizar a rede HLPN? Justifique a resposta.





Fim

